

**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS  
Facultad 9**



**TÍTULO:** Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

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

**AUTOR:** José Augusto López Amado

**TUTOR:** Ing. María de Dolores Guardia Macías

**Ciudad de La Habana, junio de 2010.  
Año 52 del triunfo de la Revolución**

**A mis tíos que han sido mis padres SIEMPRE**

**A mi familia de la UCI y de la FEU**

**Y a ti por creer en mí...**

Toda la gloria del mundo cabe en un grano de maíz.

José Martí



## AGRADECIMIENTOS

Agradezco a nuestro invencible Fidel y a esta inmensa Revolución porque gracias a ellos este guajirito de Las Tunas tuvo la posibilidad de estudiar en la mejor Universidad del país y aprender muchísimas cosas para el FUTURO, como profesional y más como persona.

Doy infinitas gracias a mis viejos, a mi papá Marrero y a mi tía Enma, ellos me lo han inculcado todo, ellos son mi ejemplo y lo serán siempre. El cariño de tantos años y su preocupación día a día no me han dejado estar solo ni un momento, a pesar de los kilómetros que nos separan.

A Suleika y a José Angel, otros padres que tengo lejos pero que me han apoyado en todo momento y a veces siento que los tengo muy cerca en mi corazón.

A mis tíos Eny y Prieto, así como a mis primos Eribertico, Yuleiny y Ailyn. Unas personas increíbles que tengo cerca y a las que le debo muchísimo.

A mi otra parte de la FAMILIA... la que he encontrado aquí en la UCI...

A mis hermanos: LARA, Yuri Antonio, Carlo Arce, YIDIAN, Isnán y Richard. A todos gracias por estar en los momentos malos y buenos, por enseñarme tanto y por aguantar mis pesadeces.

GRACIAS a mis compañeros de Brigada, a los del 9106, 9205, 9305, 9406 y 9506, mil gracias por seguirme en las locuras diarias y por apoyarme cuando nadie más confiaba en mi.

A mi GENTE de la **FEU**, a todos esos que no se cansan y siguen echando pa' lante sin descansar ni un segundo, sin dormir, TODO por la organización, TODO por los muchachos que representan, TODO por la Revolución.

A Cesar Lage, Yosbel, Yadainy, Reinaldo Rosado, Alfonsito, Javier Arza, Iyugnis, Yunia, René Lazo, Humberto, Emilio, Yanio y Aylin porque aun estamos aprendiendo de ellos y del ejemplo que han sido siempre para todos los que tuvimos el placer de trabajar junto a ellos.

La gracias a Jorge Luis, Dayana, Eneris, Jenny, Tan, Adi, Chirino, Michel Legrá, Ernesto, Edgar, Darián, Mara, Maurice, la Jabá, Rosalina, Felipe, Ulises, Arcadio, Jeinson, Frank Emilio, Lisandra Cala,

David Gerardo, Bonet, Aldo, Eiger, Yunier, Yenier, Yanela, Elizabeta, Vismar, Alelí, Freddy, Dainer, Ricardito, Carlos Rodríguez, David Ballester, Julitín, Yuniel, Haniel, Aballe, Abel Vitier, Yarismay, Carlos Miguel, Idanis, Bencomo, Mario, Yoelito, Edison, YANET, Norcky, Arleen, Aylin Curbelo, Yamilé, Héctor, Abel, Mavis, José Joel, Alicia, Odiel, Raimé, Adrián, Aksana, Evelio, Yorlen, Yosbany Tejas, Osvaldo Yusdelky, Olía, Amanda, Victor Gabriel, Manuel, Adrián Román, Livan, Lester, Daniel Angel, Alexeis, Jose Angel, Juan Carlos Hernández. A mi equipo de Residencia, a Richard Rivero, Dariel Osorio, Carlo Carlo, Eliober, Daylenis, Ani Luz, Daineris, Jorge Arce, Annia Susel, Yanisleidis, Beatriz, Caridad, Marlon, Krysna, Raydel y Goar. También a Yosbany, CARLOS CALZADO, Grisell, Omar, Thais, Yandris Mata, Humberto y a Iván. A mi gente de la Divulgación, a Carlos Rojas, Yaisi, Tito, Raico, Dayán, Daniel, Reynier, Claudia, MARCIA, Ernesto Castro, Maylien, Kenier, Rosemary, Arletys. A mi otro equipo de Recreación, Randy, Dennier, José Luis, Cire, Adrián, Chardi, Yanita, Analaisy, Lisandra, Rafael y Caballero, Licet, Uffo y los invencibles Rosita y Mariño... a todos por hacer tanta **FEU** y por compartirla conmigo.

Las gracias también a personas tan especiales como Yoilen, Raciél (un ejemplo en todo momento), Jose Enrique, Juan Carlos, Daniel Meriño y Yakelin.

A mis compañeros y compañeras de aquí y de allá, Barbarita, Gretel, Carlos Javier, Alexeis, Annaliet, Chela, Dariel Cabello, Igniris, Isabel, Isael, Odelkis, Alberto Román, El Dunier, Enrique Almeida, Douglas, Yoandri, Abelito, Lissy, Yaimy, Susell, Jean Michael, Yordanis, Angel, Frank, Karen, Argel, Arturo, Themis, Omar Antonio, Daymara, Lotti, Yandris Cajigal, Luisa, Angélica, Denis, Mercedes, Tatiana, Adrián Machín, Manolito, Yadelis, Odileisy, Evelyn, Adriana, Hainet y los Mellizos Frank y Fredy. Mis gracias al súper piquete de la merienda Aliandro, Franklin, Alexander, Yunier, Ever, Manuel Alejandro, Hogguit, Pupo, Julio, Osvaldo, Dacal y Edgar. Y gracias a mi gente del Café.

Las gracias también por el ejemplo a mis profesores Ariadna, Osmel, La Negra, Burgos, Yuneiry, Nápoles, Peñalver, Arturo, Amador, Yoelito, Alleyne, Gregorio, Pacheco, Reynaldo, Jorge Emilio, Alexander, Yoenis, Lázaro, Adrián Gracia, Lago, Yareisis, Lola, Aliosmi, Yanisley, Héctor y Raúl de la Cruz.

A todos mis compañeros de CCM en especial a Nilo, Luis Angel, Leandro y Mauricio. Y a todos mis líderes de proyecto por ayudarme siempre para que pudiera trabajar, hacer FEU y cumplir al mismo tiempo.

## DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor del trabajo Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM) y autorizo al Departamento de Señales Digitales de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

José Augusto López Amado

**Autor**

---

María de Dolores Guardia Macías

**Tutor**

## **RESUMEN**

Este trabajo aborda la investigación y desarrollo de dos subsistemas, el primero a partir de una búsqueda inicial de medias, tanto de video como de audio, permite su reproducción, edición y catalogación para luego ser exportada ya con datos específicos que la identificará de forma factible en una nueva búsqueda. En el caso del otro subsistema que se aborda se encarga de administrar y gestionar todos los permisos a los usuarios que accederán al sistema Captura y Catalogación de Medias además permite gestionar los locales, las máquinas y los recursos que se utilizarán, así como también a partir de este módulo se puede realizar reportes de las actividades que realizan en el sistema. En este documento se recogen las principales funcionalidades de los subsistemas en cuestión y se muestran las dependencias de los componentes que los integran a través del modelo de implementación. Se describen las herramientas utilizadas y se expone la validación de la solución propuesta.

## **PALABRAS CLAVES**

Administrar, Catalogación, Medias, Video, Audio

## ÍNDICE DE FIGURAS

Figura 1 - Esquema de relación entre objetos, durante la ejecución.....	17
Figura 2 - Mapeo Objeto-Relacional.....	22
Figura 3 - Autenticación de Usuario del Subsistema de Administración .....	26
Figura 4 - Insertar Rol.....	26
Figura 5 - Modificar Usuario .....	27
Figura 6 – Crear una nueva Planificación.....	28
Figura 7 – Administrar Procesos.....	28
Figura 8 – Eliminar Servidores de Media.....	29
Figura 9 - Generar y Visualizar Reportes.....	29
Figura 10 – Modificar las configuraciones de los Reportes.....	30
Figura 11 – Insertar Recurso .....	31
Figura 12 – Modificar Local.....	31
Figura 13 - Reproductor de medias.....	32
Figura 14 - Reproductor del material editado.....	33
Figura 15 - Opción de Catalogar.....	33
Figura 16 – Opción de Visualizar Reporte .....	34
Figura 17 – Búsqueda Avanzada .....	34
Figura 18 – Opción de Exportar.....	35
Figura 19 - Modelo de Implementación.....	36
Figura 20 - Diagrama de componentes del subsistema de Administración .....	37
Figura 21 - Diagrama de componentes del subsistema de Catalogación de Medias .....	38
Figura 22 – Prueba de Caja Blanca .....	44
Figura 23 – Prueba de Caja Negra .....	46

## ÍNDICE DE TABLAS

<b>Tabla 1 - Criterios de calidad durante la codificación .....</b>	<b>39</b>
<b>Tabla 2 - Nombramiento de elementos.....</b>	<b>39</b>
<b>Tabla 3 - Pruebas al Subsistema de Administración. Defectos y dificultades encontradas .....</b>	<b>50</b>
<b>Tabla 4 - Pruebas al Subsistema de Catalogación de Medias. Defectos y dificultades encontradas .....</b>	<b>53</b>

## ÍNDICE

<b>INTRODUCCIÓN</b> .....	<b>10</b>
<b>CAPÍTULO 1: “TENDENCIAS ACTUALES”</b> .....	<b>13</b>
<b>1.1 Introducción</b> .....	<b>13</b>
<b>1.2 Conceptos fundamentales sobre los subsistemas de administración y catalogación</b> .....	<b>13</b>
1.2.1 Sistema.....	13
1.2.2 Administración .....	14
1.2.3 Catalogación .....	14
1.2.4 Medias .....	15
<b>1.3 Técnicas de Programación</b> .....	<b>15</b>
<b>1.4 Paradigmas de Programación</b> .....	<b>16</b>
1.4.1 Paradigma Orientado a Objetos .....	17
<b>1.5 Lenguajes de Programación</b> .....	<b>18</b>
1.5.1 Java .....	19
<b>1.6 Entorno Integrado de Desarrollo</b> .....	<b>20</b>
1.6.1 Netbeans .....	20
<b>1.7 Hibernate</b> .....	<b>21</b>
<b>1.8 Gestor de Base de datos</b> .....	<b>22</b>
1.8.1 PostgreSQL .....	23
<b>1.9 Conclusiones</b> .....	<b>23</b>
<b>CAPÍTULO 2: “DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA”</b> .....	<b>25</b>
<b>2.1 Introducción</b> .....	<b>25</b>
<b>2.2 Descripción de los Subsistemas</b> .....	<b>25</b>
2.2.1 Subsistema de Administración .....	25
2.2.2 Subsistema de catalogación de medias .....	32
<b>2.3 Modelo de Implementación</b> .....	<b>35</b>
<b>2.4 Estándar de Codificación</b> .....	<b>38</b>
<b>2.5 Conclusiones</b> .....	<b>41</b>
<b>CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”</b> .....	<b>42</b>
<b>3.1 Introducción</b> .....	<b>42</b>
<b>3.2 Pruebas de Software</b> .....	<b>42</b>
3.2.1 Pruebas Unitarias .....	42
3.2.2 Pruebas de Caja Blanca .....	43

3.2.3 Prueba de Caja Negra .....	46
<b>3.3 Pruebas de Software a los Subsistemas.....</b>	<b>49</b>
3.3.1 Pruebas del subsistema de Administración.....	49
<b>3.3.2 Pruebas al subsistema de Catalogación de Medias.....</b>	<b>50</b>
<b>3.4 Conclusiones.....</b>	<b>53</b>
<b>CONCLUSIONES.....</b>	<b>54</b>
<b>RECOMENDACIONES.....</b>	<b>55</b>
<b>BIBLIOGRAFÍA CONSULTADA .....</b>	<b>56</b>
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>57</b>
<b>GLOSARIO.....</b>	<b>59</b>
<b>ANEXOS.....</b>	<b>61</b>



## INTRODUCCIÓN

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

## INTRODUCCIÓN

La Informática tiene su punto inicial de desarrollo a partir de la Segunda Guerra Mundial e inicios de la Guerra Fría como consecuencia de las imperiosas necesidades de información que se suscitaron en este período, así como la necesidad de desarrollar y perfeccionar métodos y medios para almacenar, distribuir, intercambiar y publicar la información, lo que favorece el surgimiento de diversas áreas de investigación y la transformación de distintas ramas científicas. Hecho que a su vez acelera la globalización de esta ciencia que en la actualidad está presente en disímiles espacios, jugando un papel primordial de apoyo a los mecanismos de trabajo y subsistencia de la raza humana en general.

El siglo XX probablemente pase a la historia como la época en la que la investigación científico-técnica estableció sus objetivos en mayor medida por presiones externas. El descubrimiento de las ondas electromagnéticas, los circuitos eléctricos y los electrónicos sirvieron como punto de partida a mediados del siglo pasado para comenzar un avance indetenible en la construcción de herramientas y medios de comunicación preferentemente audiovisuales.

Desde los inicios del siglo XXI la creciente evolución de las Tecnologías de la Información y las Comunicaciones (TIC) ha provocado una revolución tecnológica a nivel mundial. Cada día se hace más indispensable el uso de las ciencias informáticas para darle solución a diversos problemas persistentes en la sociedad. Dentro del desarrollo de las TIC está el de los medios de comunicación, esto trae como consecuencia el incremento de los materiales audiovisuales y el problema de no administrar y catalogar los mismos.

En el presente los sistemas de captura y catalogación de productos televisivos, además de tipo propietario son aplicaciones de alto costo en el mercado mundial a causa de los avances que se han logrado en el campo de los audiovisuales. En Cuba todos estos mecanismos de trabajo con medias se hacen de forma manual. Existe un personal encargado de grabar los diferentes programas que entran dentro de la planificación de la entidad y estos archivos son almacenados sin descripción alguna. Por consiguiente la dirección del Departamento de Señales Digitales perteneciente al Centro de Desarrollo de Geoinformática y Señales Digitales apoyados en los motivos antes explicados ha identificado la necesidad de informatizar los procesos de captura y catalogación de medias, a través de la implementación de varios subsistemas que conforman una solución integral.

Estudios realizados han permitido identificar la necesidad de realizar un sistema de administración y configuración que permita la integración y seguridad de la solución general de Captura y Catalogación de Medias y además posibilite a los usuarios hacer una planificación previa de los programas que serán grabados. **(1)**



## INTRODUCCIÓN

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

Este trabajo consiste en la implementación de dos subsistemas automatizados para el proceso de administración y catalogación de medias, como causa de la inexistencia de una solución igual como producto en el Departamento de Señales Digitales perteneciente al Centro de Desarrollo de Geoinformática y Señales Digitales de la Universidad de las Ciencias Informáticas, dándole respuesta así al **problema** que genera esta investigación: “necesidad de un sistema que permita realizar los procesos de administración del sistema y la catalogación de medias”.

A pesar de la poca documentación científica existente sobre el trabajo con archivos multimedia en plataformas libres, de ser factible la investigación en cuestión se lograrían importantes ganancias para el centro y para el país, pues los subsistemas de administración y catalogación de medias no solo se podrían distribuir como un software único sino que también se pueden integrar a otros sistemas que tratan archivos de televisión y video. En la dirección de televisión de la Universidad la implantación de los subsistemas propuestos traería como consecuencia la mejora y optimización de muchos de los procesos que allí se llevan a cabo.

La aplicación aportaría, desde un punto de vista económico, ganancias significativas, pues podría ser distribuida como software independiente e integrarse de manera sencilla a múltiples sistemas de gestión integral de televisión y video; de la misma manera que constituiría un ahorro para la UCI por los altos volúmenes de multimedia que genera y gestiona en su entorno. **(2)**

Entonces, a partir de lo antes expuesto se determinó como **objeto de estudio** de este trabajo es: los procesos de catalogación de medias, y administración del sistema y como **campo de acción** es la informatización de los procesos de catalogación de medias, y administración del sistema en general para la obtención del producto para la captura y catalogación de medias.

El **objetivo general** a tener en cuenta es: implementar los módulos de catalogación de medias y administración del sistema de Captura y Catalogación de Medias. Para su cumplimiento se han planteado las siguientes tareas:

- Identificación de las distintas técnicas de programación que existen a nivel internacional, nacional y de la Universidad.
- Valoración de las tendencias, técnicas, tecnologías y metodologías relacionadas con las técnicas de programación actuales, así como de las plataformas de desarrollo que la soportan.
- Construcción del Modelo de Implementación.
- Definición de un estándar de codificación.
- Implementación del sistema diseñado.
- Desarrollo del prototipo de interfaz de usuario.
- Integración con otros componentes o partes del sistema.



## INTRODUCCIÓN

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

- Desarrollo de pruebas de unidad a los componentes que certifiquen la veracidad de los algoritmos empleados.

La solución del problema antes planteado automatiza y optimiza el proceso de catalogación de medias y la administración del sistema de captura y catalogación de medias disminuyendo así el trabajo manual y el margen de error en la productividad, por lo que se **defiende la siguiente idea**:

Si se logra una implementación correcta del prototipo funcional de los procesos que intervienen en la catalogación de medias y la administración del sistema, se puede lograr un correcto funcionamiento de dos de los módulos del sistema de Captura y Catalogación de Medias.

### Métodos de Investigación Científica

Entre los **métodos teóricos** que se utilizarán para la investigación científica está:

- **Histórico-Lógico:** Se utiliza en la investigación de otros sistemas implementados para la administración y catalogación tales como: Agora, SHIVA, Videoma, Videoteca y MEDIABOX. Se analiza el funcionamiento y como tratan las medias, además de examinar las ventajas y desventajas de los mismos.
- **Modelación:** Es utilizado para la construcción de los modelos y diagramas como el Modelo de Implementación y el de Componentes. Todo esto apoyado en los artefactos y modelos que brinda la metodología RUP para crear abstracciones y explicar la realidad.

Y el **método Empírico** que se manejará es:

- **Observación:** Este método se utiliza para obtener datos importantes de otros sistemas o prototipos funcionales. En este caso se emplea para evaluar los resultados de la aplicación Tarsys de la empresa Tedral.



## CAPÍTULO 1: “TENDENCIAS ACTUALES”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

## CAPÍTULO 1: “TENDENCIAS ACTUALES”

### 1.1 Introducción

En este capítulo se hace referencia a las tendencias y técnicas de programación actuales que influyen sobre el desarrollo del trabajo, así como las plataformas que las soportan. Se tratan los principales conceptos asociados a los subsistemas a implementar, cuestión teórica fundamental a la hora de comprender el trabajo. De igual forma se describen las principales herramientas que son utilizadas para implementar los subsistemas de Administración y Catalogación de la solución Captura y Catalogación de Medias.

### 1.2 Conceptos fundamentales sobre los subsistemas de administración y catalogación

En la actualidad la programación y el desarrollo del software se hacen cada vez más importantes para el desarrollo nacional de cualquier país. Dada esta situación los países subdesarrollados y con menos recursos naturales potencian la producción en renglones donde el conocimiento y los recursos humanos sean los factores predominantes. Para el país se impone la máxima de desarrollar sobre plataformas libres y tratar de hacerlo con la tecnología más avanzada en este campo pues la competencia que ofrecen las grandes potencias presiona a producir con mayor calidad, al mismo tiempo que con herramientas menos costosas.

En un campo tan costoso y tan desarrollado como el de los medios de comunicación, realizar cualquier aporte para el país es significativo. Construir un sistema con todos sus módulos funcionando a la perfección debe ser la meta a perseguir. De aquí que la implementación de todos los subsistemas de la solución Captura y Catalogación de Medias en una plataforma libre y con herramientas del mismo tipo se convierte en un reto muy importante.

#### 1.2.1 Sistema

Para lograr la creación de un sistema en general, debemos asegurar la correcta realización de los módulos que lo conforman, de ahí que se debe conocer muy bien el concepto de este término que tanto es utilizado a diario en el mundo de la informática y la electrónica.

Según el Diccionario Estándar de Términos Eléctrico y Electrónicos (Standard Dictionary of Electrical and Electronic Terms) del Instituto de Ingenieros Electricistas y Electrónicos o IEEE por sus siglas en inglés (Institute of Electrical and Electronics Engineers) un sistema es un todo integrado, aunque compuesto de estructuras diversas, interactuantes y especializadas. Cualquier sistema tiene un



## CAPÍTULO 1: “TENDENCIAS ACTUALES”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

número de objetivos, y los pesos asignados a cada uno de ellos pueden variar ampliamente de un sistema a otro. **(3)**

Un sistema informático emplea dispositivos programables por medios de equipos computarizados, con la finalidad de procesar datos que han sido almacenados en algún momento. Sus componentes son el hardware y el software, elementos que ordenadamente relacionados entre sí contribuyen a lograr un determinado objetivo.

Los sistemas están compuestos por otros sistemas que se denominan subsistemas, ejemplo de estos son los que se desarrollaran en este trabajo. Los subsistemas de Administración y de Catalogación son parte de la solución general de Captura y Catalogación de Medias que se implementa en el Departamento de Señales Digitales, perteneciente al Centro de Desarrollo de Geoinformática y Señales Digitales de la Facultad 9 en la Universidad de las Ciencias Informáticas (UCI).

### 1.2.2 Administración

Todo sistema informático debe llevar programada al menos una variante de administración, que sirva para controlar de forma automática todo lo que ocurre o lo que sucederá en el software.

La administración es un proceso muy particular que lleva a cabo actividades de planeación, organización, ejecución y control, desempeñadas para lograr las metas implantadas desde un inicio con el uso de varios recursos y de seres humanos.

La real academia de la lengua española conceptualiza administrar como graduar o dosificar el uso de algo, para obtener mayor rendimiento de ello o para que produzca mejor efecto. **(4)**

La administración es un módulo principal en la concepción de cualquier sistema informático, así queda demostrado en lo antes expuesto. La determinante funcionalidad de dicho módulo lo convierte en un componente imprescindible para el adecuado trabajo de cualquier aplicación profesional, pues aminora la labor humana sobre la máquina proporcionando mayor poder y calidad al producto final.

### 1.2.3 Catalogación

Cuando existen varios recursos informáticos bien catalogados es más fácil trabajar con ellos, tanto a la hora de administrarlos como de usarlos.

Según la real academia de la lengua española catalogación es: Apuntar, registrar ordenadamente libros, documentos, etc., formando catálogo de ellos. **(5)**

La catalogación es un subconjunto de un campo mucho mayor que, en ocasiones, es llamado control bibliográfico; por lo que resulta de ayuda verla a través de este contexto. **(6)**



## CAPÍTULO 1: “TENDENCIAS ACTUALES”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

El proceso de catalogación es de suma importancia cuando tenemos un gran cúmulo de información, en este caso de archivos de audio y video. La catalogación ayuda de forma directa a colocar donde se determine los ficheros con una descripción previa, para luego acceder a los mismos de forma fácil y organizada.

### 1.2.4 Medias

La catalogación es un proceso que es aplicado sobre las medias. Estas son capturadas previamente para luego ser procesadas, provienen de diferentes eventos planificados y de diferentes medios.

Los medios de comunicación constituyen el origen de la denominada cultura de masas, de donde se nutre cultural e intelectualmente un alto porcentaje de la población, lo que hace que la estructura social sea más homogénea. Son generadores de nuevas tendencias sociales, desde las actitudes políticas hasta las normas o los valores, pasando por las modas o las necesidades de consumo. (7)

Entre los principales medios de comunicación que más influyen actualmente en nuestra sociedad podemos encontrar la radio y la televisión. Son grandes esferas que generan un excesivo contenido audiovisual, que es recogido principalmente en archivos de audio y de video. Podemos definir como media a todo aquel contenido de sonido y video que es utilizado, capturado, administrado, catalogado y reproducido por los medios de comunicación.

### 1.3 Técnicas de Programación

El trabajo tanto con medias como con otro tipo de archivo ha ido evolucionando aparejado con la informática, aplicándose nuevas prácticas y tecnologías de punta. Para esto también se ha desarrollado la comunicación con las máquinas y la forma de hacerlas ejecutar eventos y cálculos que el ser humano jamás podría hacer.

Escribir programas de computación es un arte que tiene muchísimos años y que cada día adquiere más adeptos. (8) Existen varias clases de programación, dependiendo de los métodos utilizados y las técnicas empleadas.

Uno de estos tipos es la **programación modular** que consta de varias secciones divididas de forma que interactúan a través de llamadas a procedimientos, que integran el programa en su totalidad.

En la programación modular, el programa principal coordina las llamadas a los módulos secundarios y pasa los datos necesarios en forma de parámetros. A su vez cada módulo puede contener sus propios datos y llamar a otros módulos o funciones.

Por otra parte la **programación funcional** se caracteriza principalmente por permitir declarar y llamar a funciones dentro de otras funciones.



## CAPÍTULO 1: "TENDENCIAS ACTUALES"

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

La **programación lógica** se suele utilizar en la inteligencia artificial y pequeños programas infantiles. Se trata de una programación basada en el cálculo de predicados (una teoría matemática que permite lograr que un ordenador basándose en hecho y reglas lógicas, pueda dar soluciones inteligentes).

En la **programación concurrente** que es otro tipo de programación se utiliza cuando tenemos que realizar varias acciones a la vez. Se suele utilizar para controlar los accesos de usuarios y programas a un recurso de forma simultánea. Se trata de una programación más lenta y laboriosa, obteniendo unos resultados lentos en las acciones.

La **programación estructurada** está compuesta por un conjunto de técnicas que han ido evolucionando aumentando considerablemente la productividad del programa reduciendo el tiempo de depuración y mantenimiento del mismo. Esta programación estructurada utiliza un número limitado de estructuras de control, reduciendo así considerablemente los errores.

Un programa está estructurado si posee un único punto de entrada y sólo uno de salida, existen de "1 a n" caminos desde el principio hasta el fin del programa y por último, que todas las instrucciones son ejecutables sin que aparezcan bucles infinitos. **(9)**

Para la implementación de los subsistemas de Administración y Catalogación de medias del sistema de Captura y Catalogación de Medias (CCM) se ha decidido utilizar la **programación orientada a objetos**, por las ventajas que conlleva aplicar esta técnica:

- Promueve la reutilización y extensión del código.
- Permite crear sistemas más complejos.
- Relaciona el sistema implementado con el mundo real.
- Facilita la creación de programas visuales.
- Permite la construcción de un prototipo funcional.
- Agiliza el desarrollo y construcción del software.
- Facilita el trabajo en equipo.
- Facilita el mantenimiento del software.

Este tipo de programación se ve reflejada en lenguajes tan popularmente usados como el C#, C/C++ y Java.

### 1.4 Paradigmas de Programación

No sólo existen varios tipos de lenguajes de programación, sino que también podemos encontrar distintas formas de programar una aplicación. Hay diversos paradigmas que permiten encontrar una solución más adecuada a nuestros problemas. La idea es que el programador los conozca y sepa seleccionar el adecuado para cada situación particular. **(8)**



La Programación Orientada a Objetos constituye un paradigma de programación ya que es una perspectiva exclusiva para la construcción de software. Los paradigmas no son mejores unos que otros, solo que uno es más apropiado según la situación dada. A veces a la hora de programar suelen mezclarse varios paradigmas y la solución sigue siendo factible.

### 1.4.1 Paradigma Orientado a Objetos

La programación orientada a objetos (POO) es una forma de estructurar un programa sobre la base de objetos. Cada elemento o componente en un programa que se base en esta técnica es concebido como un objeto que tiene propiedades y métodos. La ejecución de un programa depende pura y exclusivamente de una interacción de los objetos que lo componen. **(8)**

La programación orientada a objetos trata de amoldarse al modo de pensar del hombre y no al de la máquina. Esto es posible gracias a la forma racional con la que se manejan las abstracciones que representan las entidades del dominio del problema, y a propiedades como la jerarquía o el encapsulamiento.

El elemento básico de este paradigma no es la función (elemento básico de la programación estructurada), sino un ente denominado objeto. **(10)**

Un **objeto** es una entidad que tiene en los datos sus atributos particulares, y en los métodos o procedimientos la forma de operar sobre ellos. Durante la ejecución, los objetos reciben y envían mensajes a otros objetos para realizar las acciones requeridas (figura 1).



Figura 1 - Esquema de relación entre objetos, durante la ejecución.



## CAPÍTULO 1: “TENDENCIAS ACTUALES”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

Un objeto consta de tiempo de vida, estado, comportamiento y se consideran como instancias de clases.

Las **clases** son abstracciones que representan a un conjunto de objetos con un comportamiento e interfaz común. Una clase es una abstracción que define la forma del objeto, se podría hablar de la clase como el molde para los objetos. **(10)**

La programación orientada a objetos introduce también otros conceptos, que amplían y superan los que se conocían hasta el momento.

**Abstracción:** se define como la separación de propiedades de la implementación. Los objetos son agentes abstractos con ciertas aptitudes que comuniquen entre ellos. No hay porque conocer todos los detalles.

**Polimorfismo:** uso de la misma definición con diferentes tipos de datos.

**Herencia:** el propósito de la herencia es extender la funcionalidad de una clase por rehúso automático de las definiciones y la funcionalidad de otra clase: los objetos contienen las propiedades y el comportamiento de todas sus clases padres, además de los de su propia clase. **(11)**

Entre todos los lenguajes orientados a objetos que se destacan se pueden mencionar:

- ABAP
- Ada
- C++
- C#
- Clarion
- Delphi
- Eiffel
- Java
- Léxico
- Perl
- PHP (a partir de su versión 5)
- Python
- Ruby
- Smalltalk
- VB.NET

Lo que es verdad es que para cada paradigma conviene utilizar ciertos lenguajes de programación, y no cualquiera. Esto se debe a que la mayoría de los lenguajes fueron creados para ser utilizados en determinados ambientes de programación. **(8)**

### 1.5 Lenguajes de Programación.

Los lenguajes de programación son un conjunto de reglas, herramientas y condiciones que permiten crear programas o aplicaciones dentro de una computadora. Estos programas son los que permitirán ordenar distintas acciones a la computadora en un “idioma” comprensible por ella. **(8)**

La computadora codifica todas sus acciones en ceros y unos. Los lenguajes más próximos a la máquina y su forma única (código binario) de interpretar las acciones se denominan lenguajes de bajo



## CAPÍTULO 1: “TENDENCIAS ACTUALES”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

nivel, y por el contrario los que son más cercanos a los programadores se llaman lenguaje de alto nivel.

### 1.5.1 Java

Java es un lenguaje de alto nivel que surge en 1991 cuando un grupo de ingenieros de Sun Microsystems trataron de diseñar un nuevo lenguaje de programación destinado a electrodomésticos. La reducida potencia de cálculo y memoria de los electrodomésticos llevó a desarrollar un lenguaje sencillo capaz de generar código de tamaño muy reducido. **(12)**

Entre sus primeros creadores están James Gosling, Patrick Naughton, Chris Warth, Ed Frank y Mike Sheridan.

Java fue diseñado como un lenguaje orientado a objetos desde el principio. Los objetos agrupan en estructuras encapsuladas tanto sus datos como los métodos (o funciones) que manipulan esos datos. La tendencia del futuro, a la que Java se suma, apunta hacia la programación orientada a objetos, especialmente en entornos cada vez más complejos y basados en red. **(13)**

Existen distintos programas comerciales que permiten desarrollar código Java. La compañía Sun, creadora de Java, distribuye gratuitamente la máquina de desarrollo de Java (Java Development Kit o JDK). Se trata de un conjunto de programas y librerías que permiten desarrollar, compilar y ejecutar programas en Java. Incorpora además la posibilidad de ejecutar parcialmente el programa, deteniendo la ejecución en el punto deseado y estudiando en cada momento el valor de cada una de las variables (con el denominado Debugger). Cualquier programador con un mínimo de experiencia sabe que una parte muy importante (muchas veces la mayor parte) del tiempo destinado a la elaboración de un programa se destina a la detección y corrección de errores. Existe también una versión reducida del JDK, denominada JRE (Java Runtime Environment) destinada únicamente a ejecutar código Java (no permite compilar).

La compañía Sun describe el lenguaje Java como “simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutra, portable, de altas prestaciones, multitarea y dinámico”. Además de una serie de halagos por parte de Sun hacia su propia criatura, el hecho es que todo ello describe bastante bien el lenguaje Java, aunque en algunas de esas características el lenguaje sea todavía bastante mejorable. **(12)**

Java es un lenguaje multiplataforma. Existen compiladores de Java para la mayoría de las plataformas e intérpretes de Java para todas. **(14)**

Además de lo que se necesita para la implementación de los dos subsistemas en cuestión Java también permite a los programadores:



## CAPÍTULO 1: “TENDENCIAS ACTUALES”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

- Crear programas para que funcionen en un navegador web y en servicios web.
- Desarrollar aplicaciones para servidores como foros en línea, tiendas, encuestas, procesamiento de formularios HTML, etc.
- Desarrollar potentes y eficientes aplicaciones para teléfonos móviles, procesadores remotos, productos de consumo de bajo coste y prácticamente cualquier tipo de dispositivo digital. **(15)**

Java es un potente lenguaje para programar sobre plataformas libres que es una de las finalidades de este trabajo. Para la implementación de los módulos se combinan librerías y servicios que usan el lenguaje Java para crear otros servicios totalmente personalizados, algo que aporta un alto nivel al software. Este lenguaje se ha seleccionado para implementar los subsistemas de Administración y Catalogación del sistema Captura y Catalogación de Medias (CCM) ya que ha sido probado, mejorado y ampliado por una comunidad especializada de desarrolladores, la mayor y más activa del mundo, lo que brinda un excelente respaldo a la hora de programar. Gracias a su versatilidad, eficiencia y portabilidad Java es un recurso inestimable para el desarrollo de software, pues permite implementar en una plataforma y ejecutar la aplicación en prácticamente cualquier otra.

### 1.6 Entorno Integrado de Desarrollo

El lenguaje de programación Java tiene varios entornos integrados de desarrollo o IDE por sus siglas en inglés (Integrated Development Environment). Estos son programas compuestos por un cúmulo de herramientas encaminadas a la construcción de aplicaciones. Un IDE puede ser un software por sí mismo o pueden ser parte de otro software que así lo requiera. Los IDE se encargan de gestionar una estación de trabajo sencilla y amigable para los lenguajes de programación y a la vez brindan un constructor de interfaz gráfica, un editor de código, así como un compilador en su mayoría.

#### 1.6.1 Netbeans

En el caso de Java que es el lenguaje seleccionado para realizar los subsistemas propuestos a implementar en este trabajo, tiene entre sus entornos integrados de desarrollo el Eclipse y el NetBeans. Se opta por el NetBeans por una serie de características que lo hace superior a la hora de trabajar con los elementos que se necesitan para administrar el sistema y para la catalogación de las medias.

El NetBeans nació como un proyecto estudiantil en la República Checa en 1996. Su nombre original era Xelfi y fue el primer IDE para Java, escrito en Java. El emprendedor Roman Stanek, invierte en Xelfi y nace un negocio, mientras que Jarda Tulach, miembro del equipo original propone el nombre NetBeans. **(16)**



## CAPÍTULO 1: “TENDENCIAS ACTUALES”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

Este IDE tan popular es un producto de código abierto, con todos los beneficios del software disponible en forma gratuita, el cual ha sido examinado por una comunidad de desarrolladores. Este enfoque de bienes comunes creativos ha permitido una mayor capacidad de uso, con cada nueva versión, y ha proporcionado a los desarrolladores mayor flexibilidad, al modificar el IDE, si así lo desean. **(17)**

NetBeans ofrece a los desarrolladores múltiples ventajas, en la creación de nuevas aplicaciones multiplataforma. En estos momentos en que la arquitectura SOA (Arquitectura Orientada a Servicios) requiere servicios relacionados que operen procesos concretos del negocio, este IDE satisface los requisitos con un amplio conjunto de herramientas independientes de la plataforma, modulares y orientadas al objeto.

El NetBeans permite desarrollar aplicaciones web, Mobile, Enterprise y de escritorio, además que sirve para trabajar sobre lenguajes como Java, C/C++, PHP, Python, entre otros. Es un IDE multilenguaje completo y modular que le brinda facilidades al programador para el desarrollo intuitivo, a la vez que posee una gran cantidad de módulos de terceros o plugins que sirven de ayuda al implementador para hacer su trabajo más sencillo y eficiente. Es gratis y de código abierto desde junio del 2000. A través de él se pueden crear ventanas, barras de herramientas y menús, que complementan aplicaciones completas para el cliente. Otra de sus importantes ventajas por la cual se utiliza para la implementación de los dos módulos de Administración y Catalogación de Medias es que al igual que el lenguaje Java tiene una gran comunidad de programadores y usuarios.

### 1.7 Hibernate

Como se desarrollan los subsistemas sobre la plataforma Java, se utiliza de apoyo el Hibernate que es una herramienta que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) que permiten establecer estas relaciones. **(18)**

Hibernate es una herramienta ORM (Object-Relational Mapping) como se muestra en la figura 2, es un producto de código abierto con disímiles prestaciones, muy buena documentación y excelente estabilidad.



Figura 2 - Mapeo Objeto-Relacional.

El Hibernate presenta un modelo de programación natural, con una gran escalabilidad, es no intrusivo, estilo POJO (Plain Old Java Object), tiene muy buena documentación (fóruns para ayuda, libros) y comunidad activa con muchos usuarios alrededor del mundo. Hibernate es muy eficiente, tiene una arquitectura de caché de doble capa, admite los contextos de persistencia de larga vida denominados detach/reattach objetos. Está bajo licencia LGPL (Lesser GNU Public License). Brinda facilidades a la hora de realizar consultas, debido en parte a que se realizan en un potente lenguaje de consultas orientado a objetos.

### 1.8 Gestor de Base de datos

Los sistemas de gestión de bases de datos o SGBD (Database Management System, abreviado en inglés DBMS) son sistemas que permiten el manejo, actualización y utilización de los datos almacenados en diferentes bases de datos. Estas aplicaciones tiene como objetivo principal el de proporcionar al usuario diversos medios que le permitan administrar la información abstrayéndolo de cómo se guardan los datos en los dispositivos de almacenamientos y los métodos que se utilizan para esto. También una de las ventajas de los SGBD es que organizan los datos sin que esto tenga un gran impacto dentro del código de los programas, además al utilizar las interfaces y el lenguaje de consulta que brindan podemos simplificar el acceso a los datos desde la aplicación.

Existen varios gestores de bases de datos, dentro de los más utilizados se encuentran el PostgreSQL, el MySQL y Oracle, todos con grandes comunidades alrededor del mundo entero.



## CAPÍTULO 1: “TENDENCIAS ACTUALES”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

### 1.8.1 PostgreSQL

El PostgreSQL es un proyecto con más de 15 años de vida. Se inicia en la Universidad de Berkeley en 1977 bajo el nombre Ingres como un ejercicio de aplicación de las teorías de las RDBMS (Sistema Administrador de Bases de Datos Relacionales). En 1986 cambia su nombre a Postgres con el objetivo de aplicar los conceptos de Objetos Relacionales y en 1995 vuelve a cambiar su nombre a Postgres95 que luego derivaría a PostgreSQL.

Este gestor trabaja en casi todos los principales sistemas operativos como son el Linux, Unix, BSDs, Mac OS, Beos, Windows, etc. Su documentación está muy bien organizada, es pública y libre. **(19)**

PostgreSQL ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que otros productos, conservando todas las características, estabilidad y rendimiento. Es extremadamente común que compañías reporten que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad. Ni una sola vez. Simplemente funciona.

Una de las ventajas del PostgreSQL es que el código fuente está disponible para todos sin costo. Si su equipo necesita extender o personalizar PostgreSQL de alguna manera, pueden hacerlo con un mínimo esfuerzo, sin costos adicionales. Esto es complementado por la comunidad de profesionales y entusiastas de PostgreSQL alrededor del mundo que también extienden PostgreSQL todos los días. **(20)**

Para el almacenamiento de los datos generados por los módulos de Administración y Catalogación se utiliza este gestor, teniendo en cuenta lo antes planteado, además en la Universidad se tienen muy buenas experiencias de trabajo con este software. PostgreSQL cumple con las necesidades que presenta la base de datos del sistema, además está liberado bajo la licencia BSD (Berkeley Software Distribution) lo que hace que esté disponible su código fuente y se pueda modificar según sea necesario, para adaptarlo a las prestaciones atípicas que presenta una aplicación que trabaja con datos audiovisuales.

### 1.9 Conclusiones

En este primer capítulo se han definido una serie de conceptos necesarios para comprender el entorno del trabajo. Se ha realizado un análisis de las diferentes técnicas de programación que se utilizan en el mundo de la informática, para poder determinar la más correcta y así darle cumplimiento a las tareas de implementación que depara el desarrollo del sistema. Se explican los detalles que determinaron la selección del lenguaje a utilizar, así como sus ventajas. También se estudió las diversas herramientas



## **CAPÍTULO 1: “TENDENCIAS ACTUALES”**

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

que son aplicadas en el desarrollo de los subsistemas de Administración y Catalogación de la solución Captura y Catalogación de Medias. A partir de todo lo investigado y analizados los trabajos de diplomas que anteceden a este se ha creado una sólida base para la implementación de los dos subsistemas en cuestión.



## CAPÍTULO 2: “DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA”

### 2.1 Introducción

En este capítulo se realiza una breve descripción de las funcionalidades de los subsistemas a implementar, así como de los distintos módulos que los conforman. Se hace referencia al modelo de implementación constituido a partir de los diagramas de componentes correspondientes a los subsistemas de Administración y Catalogación. Este está basado en el diseño realizado del Sistema de Captura y Catalogación de Medias. Se define también el estándar de codificación utilizado en la implementación.

### 2.2 Descripción de los Subsistemas

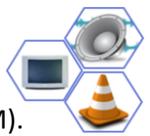
Actualmente cuando se trabaja en un proyecto con una gran cantidad de casos de uso estos son divididos según las funcionalidades en aplicaciones independientes que trabajan por sí mismas, a las que denominamos subsistemas. Estos subsistemas agrupan los casos de usos por áreas funcionales. El Sistema de Captura y Catalogación de Medias está dividido en cuatro subsistemas, en este trabajo se da implementación a dos de ellos: el subsistema de Administración y el de Catalogación de Medias.

#### 2.2.1 Subsistema de Administración

El subsistema de Administración provee al sistema en general de la seguridad y la configuración que este necesita para lograr un buen desempeño y una buena integración entre todos los módulos. Dentro de este subsistema es donde se gestiona todo lo referente a usuarios y sus roles dentro del software, lo concerniente al servidor de media y a las estaciones de trabajo. También desde este módulo se puede controlar las excepciones, errores y monitorear las acciones realizadas dentro del sistema así como llevar un control de las trazas que dejan los usuarios cuando interactúan con el software.

##### ➤ Autenticar Usuario

Es la parte inicial del subsistema, donde se verifica que el usuario que accede tiene todos los permisos asignados para entrar y hacer cambios que repercutirán de forma directa en las funcionalidades de todo el sistema en general. Como se puede apreciar en la figura 3 el que desee interactuar en el subsistema debe introducir usuario, contraseña y a partir de ahí se verificará en la base de datos para permitirle la entrada, de tener autorizado el acceso.



**Subsistema de Administración**

Autenticación

Usuario :

Contraseña :

Figura 3 - Autenticación de Usuario del Subsistema de Administración

➤ **Funcionalidad gestionar roles y permisos de usuario.**

Esta funcionalidad permite la gestión de roles y permisos de los usuarios del sistema. A la hora de crear los roles, se asignan los permisos correspondientes a cada uno de ellos, facilitando la asignación de privilegios como se muestra en la figura 4. Según las necesidades del sistema los roles podrán ser modificados o eliminados.

Insertar Rol

Nombre:  Descripción:

Permisos:

**Administración**

Permiso de Administración

Planificar

Administrar

Figura 4 - Insertar Rol

➤ **Funcionalidad gestionar usuario.**

Haciendo uso de esta funcionalidad se podrá llevar a cabo el almacenamiento de todos los datos que se necesiten del usuario mediante inserción, modificación y eliminación de estos; en dependencia de



las necesidades y especificaciones de la aplicación en cada momento. Al insertar un usuario se asigna a cada uno su rol correspondiente, el cual engloba todos los permisos de acceso del mismo. De esta manera se controla el acceso a la aplicación basándose en los roles y permisos de cada usuario según su desempeño. Para modificar se selecciona un usuario del listado como se puede apreciar en la figura 5 y se pasa a cambiar los datos del mismo, para luego guardarlo en la base de datos con los campos actualizados.

Usuario	Rol	Nombre	1er Apellido	2do Apellido
isael	administrador	isaelo	a	s
a	a	Leandro	Rosales	Rodes
yan	administrador	yanio	Hdez	Heredia
asd	administrador	rafael	martin	sanchez
as	administrador	roberto	frometa	lescay
adnan	Administrad...	adnan	Fuentes	Diaz
afuentesd	rolAdnan	Adnan	Fuentes	Diaz
Operacion	administrador	adnan	fuentes	diaz
tmate	administrador	adnan	fuentes	diaz

Figura 5 - Modificar Usuario

➤ **Funcionalidad planificar procesos.**

Esta funcionalidad permite realizar la planificación de los eventos que se realizarán en la aplicación, o sea, planificar grabaciones, transcripciones y capturas. Las planificaciones contribuyen a racionalizar los recursos disponibles. Para crear una planificación nueva como se ilustra en la figura 6 se debe especificar el de grabación, con su fecha de inicio y de fin, además de poder definir si será periódica o no. Será posible reorganizar las planificaciones existentes en caso que todos los recursos se encuentren ocupados en un momento dado y surja una programación con mayor prioridad que las existentes. Para esto será necesario modificar o eliminar una planificación existente y establecer los nuevos parámetros.



Crear Planificación:

Tipo de Planificación:

**Captura de Video** Grabar Audio

Capturar Video:

Capturar Video

Nombre del Canal: -Seleccione-

Nombre de la Máquina: -Seleccione-

Recursos: -Seleccione-

Fecha de Inicio:

Dia / Mes / Año: 30/05/10

Hora: 00 Minuto: 00

Fecha de Fin:

Dia / Mes / Año: 30/05/10

Hora: 00 Minuto: 00

Periodica:

Periodica

Lunes Martes Miércoles Jueves Viernes Sábado Domingo

Insertar

Figura 6 – Crear una nueva Planificación

➤ **Funcionalidad administrar procesos.**

La administración de procesos posibilita operar con los procesos que se encuentran en ejecución en las distintas máquinas que administra el sistema, permitiendo: eliminar algún proceso que esté bloqueado y reiniciarlo, así como priorizar algunos que sea necesario por encima de otros.

Proceso:

Dirección IP: -Seleccione-

Usuario:  Conectar

Lista de procesos:

Actualizar Priorizar Eliminar

Figura 7 – Administrar Procesos

➤ **Funcionalidad gestionar servidores de medias.**

Esta funcionalidad es la encargada de permitir a los administradores (insertar, modificar o eliminar) servidores de medias con que trabajará el sistema directamente. En el caso de querer prescindir de



algún servidor se selecciona de la lista que esta almacenada en la base de datos como se muestra en la figura 8 y se procede a eliminarlo.

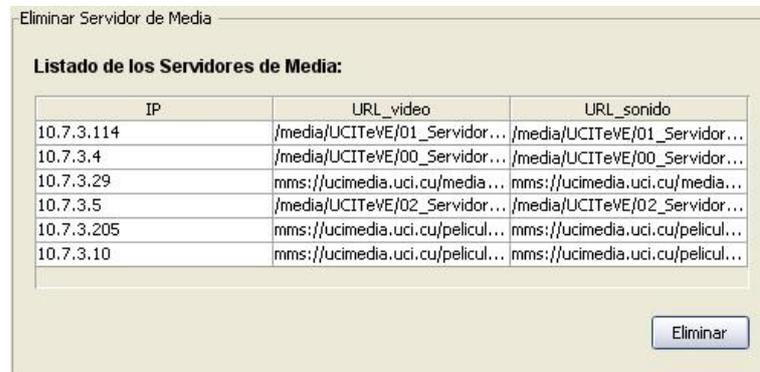


Figura 8 – Eliminar Servidores de Media

➤ **Funcionalidad generar y visualizar reportes de operaciones y trazabilidad de usuarios en la aplicación.**

Permite al sistema obtener los reportes a partir de las configuraciones previamente definidas, brindando a la aplicación el mecanismo para el seguimiento de los procesos, controlar las excepciones y errores, darle seguimiento a las acciones realizadas en la misma y llevar las trazas de las operaciones principales efectuadas por los usuarios. Al seleccionar un reporte del listado cargado de la base de datos como se ejemplifica en la figura 9 se puede visualizar y exportar para obtenerlo en forma de documento.



Figura 9 - Generar y Visualizar Reportes



➤ **Funcionalidad gestionar reportes de operaciones y trazabilidad de usuarios.**

Con el uso de esta funcionalidad se podrá dar seguimiento a los procesos de la aplicación para controlar las excepciones y errores de manera general, permite controlar las acciones realizadas en la aplicación y llevar las trazas de las operaciones principales realizadas por los usuarios. Si se desea modificar alguna configuración se debe seleccionar de un listado que se genera, mostrado en la figura 10, para proceder a cambiar el día, la hora y seleccionar el reporte.

Nombre	Hora	Minuto	Segundo
Usuarios Registrados	16	54	40
Usuarios Registrados	1	1	1
Usuarios Registrados	0	0	0
Usuarios Registrados	0	1	1
Operaciones Realiz...	14	50	25
Operaciones Realiz...	16	48	40
Asignacion a Usuarios	0	0	0
Usuarios Registrados	12	4	0
Usuarios Registrados	0	0	0
Usuarios Registrados	22	22	0

**Configuracion de Tiempo**

Lunes  Martes  Miercoles  Jueves  Viernes  Sabado  Domingo

**Configuracion de la Hora**

Hor Min Seg

**Seleccione el Reporte**

-Seleccione-

Modificar

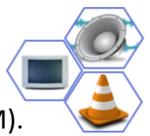
Figura 10 – Modificar las configuraciones de los Reportes

➤ **Funcionalidad gestionar máquina.**

Permite gestionar el hardware existente en el sistema, posibilitando insertar, modificar o eliminar las computadoras que procesarán la información de la aplicación.

➤ **Funcionalidad gestionar recursos.**

Permite gestionar los recursos que pueden poseer las máquinas en donde se procesa la información, tarjeta(s) de captura de video que pueden variar en número según las necesidades de la solución, discos duros, tarjeta(s) de audio y otros dispositivos necesarios para el buen funcionamiento del



sistema. Para insertar algún recurso nuevo se introduce el nombre, la máquina, el tipo de recurso y se procede a adicionarlo al listado de recursos como se ejemplifica en la figura 11.

Nombre	Tipo
6	recurso1
s	recurso1
aaa	recurso1
hh	disco duro
tarjeta1	recurso1
tarjeta2	recurso1

Figura 11 – Insertar Recurso

➤ **Funcionalidad gestionar local.**

Esta funcionalidad posibilita especificar (insertar, modificar o eliminar) los locales donde se crearan las condiciones de las grabaciones de audio. En el caso de quererse modificar un local se selecciona por el nombre y este puede ser cambiado y guardado como se muestra en la figura 12.

Nombre
Lab 202
Lab 206
Lab 205
Lab 203
Lab 201

Figura 12 – Modificar Local



### **2.2.2 Subsistema de catalogación de medias**

Este subsistema está integrado por varias funcionalidades que trabajan directamente con las medias. Se encarga de la catalogación de los archivos audiovisuales con los que se trabaja, permitiendo además reproducirlos, editarlos y exportarlos. Esto permite que los ficheros queden organizados y en carpetas según los criterios ajustados, atendiendo principalmente al tipo de contenido del fichero sin tomar en cuenta otros aspectos que podrían facilitar la búsqueda de un material para cualquier usuario que no tenga claro lo que busca específicamente.

#### **➤ Autenticar Usuario**

Es la parte inicial del subsistema, donde se verifica que el usuario que accede tiene todos los permisos asignados para la búsqueda y trabajo con las medias que están almacenadas en los servidores.

#### **➤ Funcionalidad reproducir medias.**

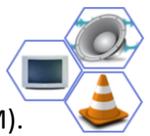
El sistema a través de esta funcionalidad será capaz de reproducir medias, ejemplificado en la figura 13, para lo cual se gestiona el material desde un servidor de streaming y se visualiza sin necesidad de descargar el fichero en la PC donde se esté trabajando.



Figura 13 - Reproductor de medias

#### **➤ Funcionalidad editar archivos de media.**

Esta funcionalidad brinda una interfaz para una edición en caliente de la media, es decir, en este proceso, físicamente la media no será afectada, solo se crearán nuevas referencias para su posible



futura exportación. Esta nueva media se podrá visualizar en un reproductor aparte como se muestra en la figura 14.



Figura 14 - Reproductor del material editado

➤ **Funcionalidad catalogar media.**

A través de esta funcionalidad la aplicación será capaz de extraer de la base de datos todos los campos necesarios para catalogar un archivo determinado y visualizarlo desde el servidor de streaming actualizando los metadatos referentes al mismo en la base de datos.



Figura 15 - Opción de Catalogar



➤ **Funcionalidad hacer reportes a petición de los usuarios.**

La responsabilidad de esta funcionalidad es la de permitir hacer reportes de los archivos almacenados en el servidor de medias de acuerdo a los criterios del usuario y salvarlos en un formato estándar para su portabilidad. Se selecciona el tipo de reporte que se desea generar como se muestra en la figura 16 y se puede pasar a visualizarlo aceptando la opción seleccionada.



Figura 16 – Opción de Visualizar Reporte

➤ **Funcionalidad buscar media.**

Con la utilización de esta funcionalidad se podrá hacer búsquedas de medias de acuerdo a criterios específicos y se obtendrán los resultados en un listado de medias como se visualiza en la figura 17.



Figura 17 – Búsqueda Avanzada



➤ **Funcionalidad exportar media.**

Esta funcionalidad permite al sistema de acuerdo a un pedido, exportar la media desde el servidor de medias hasta una dirección especificada por el usuario como se puede apreciar en la figura 18, pasando luego por un proceso de transcodificación.

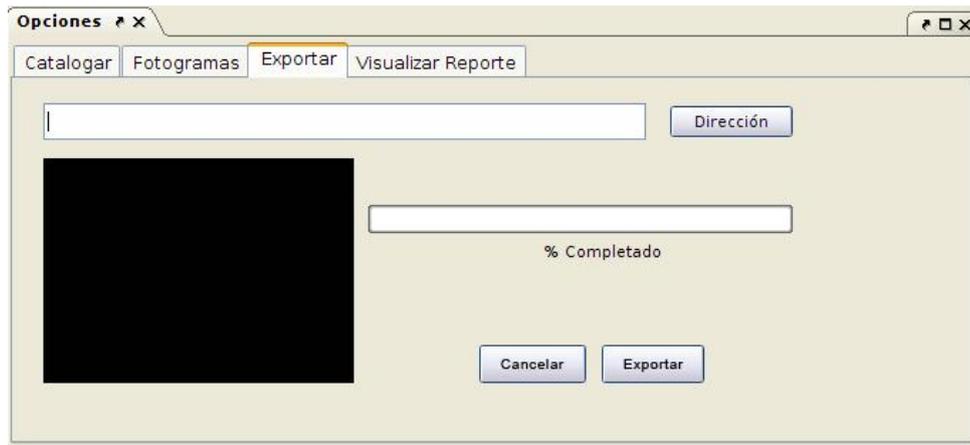


Figura 18 – Opción de Exportar

**2.3 Modelo de Implementación**

El modelo de implementación describe cómo los elementos del modelo de diseño, como las clases, se implementan en términos de componentes, como ficheros de código fuente, ejecutables, etc. El modelo de implementación describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen los componentes unos de otros.

**(21)**

En el modelo de implementación del sistema Captura y Catalogación de Medias (CCM) representado en la figura 3 se observan los cuatro subsistemas y los componentes externos, en su mayoría librerías, que se relacionan con la base de datos para concebir el buen funcionamiento del software integrado. Dentro de los componentes podemos encontrar el **jasper\_lib** que es una librería para generar reportes, el **jcalendar** que ahorra la tarea de desarrollar un control que cumpla las funcionalidades de un calendario, el **jVLC** usado para reproducir medias dentro de la aplicación, el **Hibernate** para el mapeo de la base de datos, el **LAME** para la conversión de audio y el **MPlayer** para reproducir, capturar y codificar los videos apoyado en el codificador Mencoder.

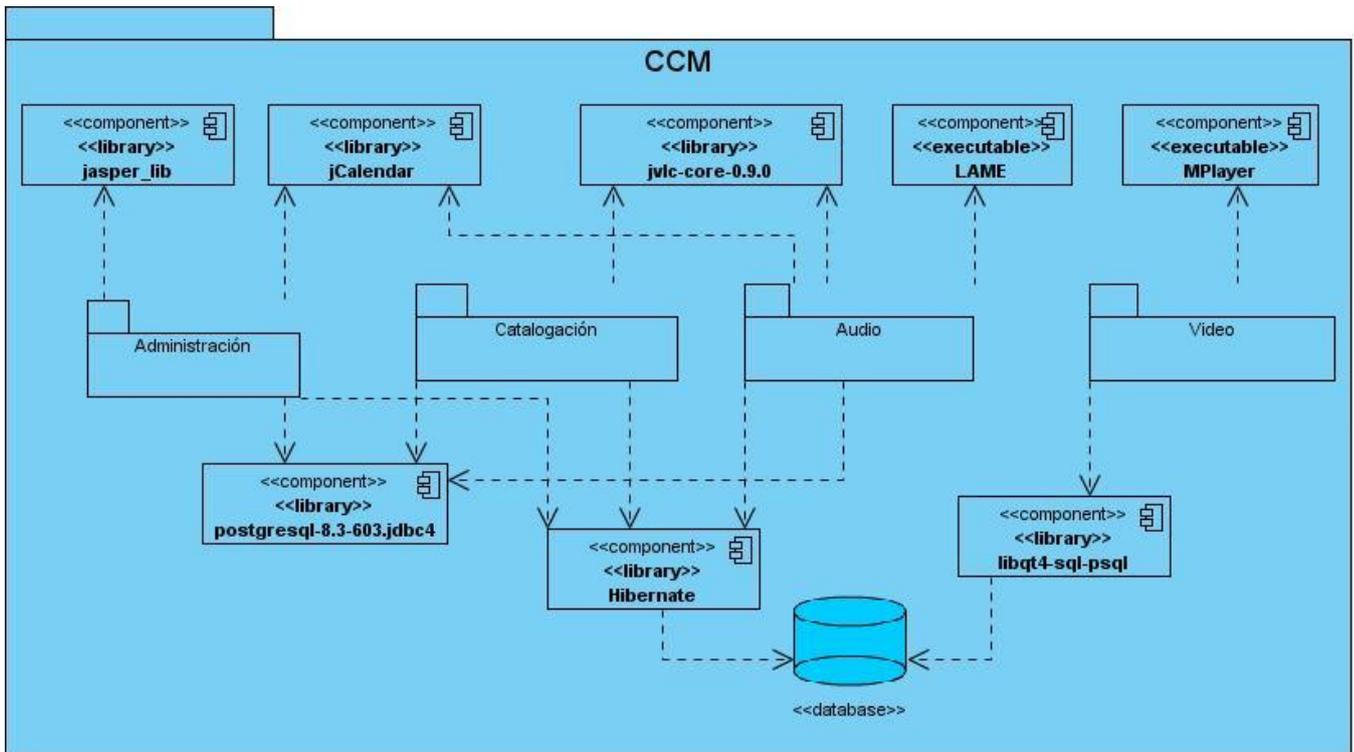


Figura 19 - Modelo de Implementación

➤ Diagramas de componentes

Un diagrama de componentes representa las dependencias entre componentes software, incluyendo componentes de código fuente, componentes del código binario, y componentes ejecutables. Un módulo software se puede representar como componente. Algunos componentes existen en tiempo de compilación, algunos existen en tiempo de enlace, y algunos existen en tiempo de ejecución; otros componentes existen en varias de estas ocasiones.

El diagrama de componentes es un diagrama que muestra las organizaciones y las dependencias entre tipos de componentes. **(22)**

Los principales estereotipos estándares que presentan los componentes dentro de un modelo son:

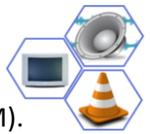
`<<executable>>` Modela programas que se ejecutan en un nodo.

`<<file>>` Simboliza ficheros de datos o código fuente.

`<<library>>` Constituye librerías estáticas o dinámicas.

`<<table>>` Simboliza tablas de bases de datos.

`<<document>>` Representa documentos.



A continuación el diagrama de componentes del subsistema de Administración representado en la figura 4 y el del subsistema de Catalogación de Medias en la figura 5.

- Subsistema de Administración

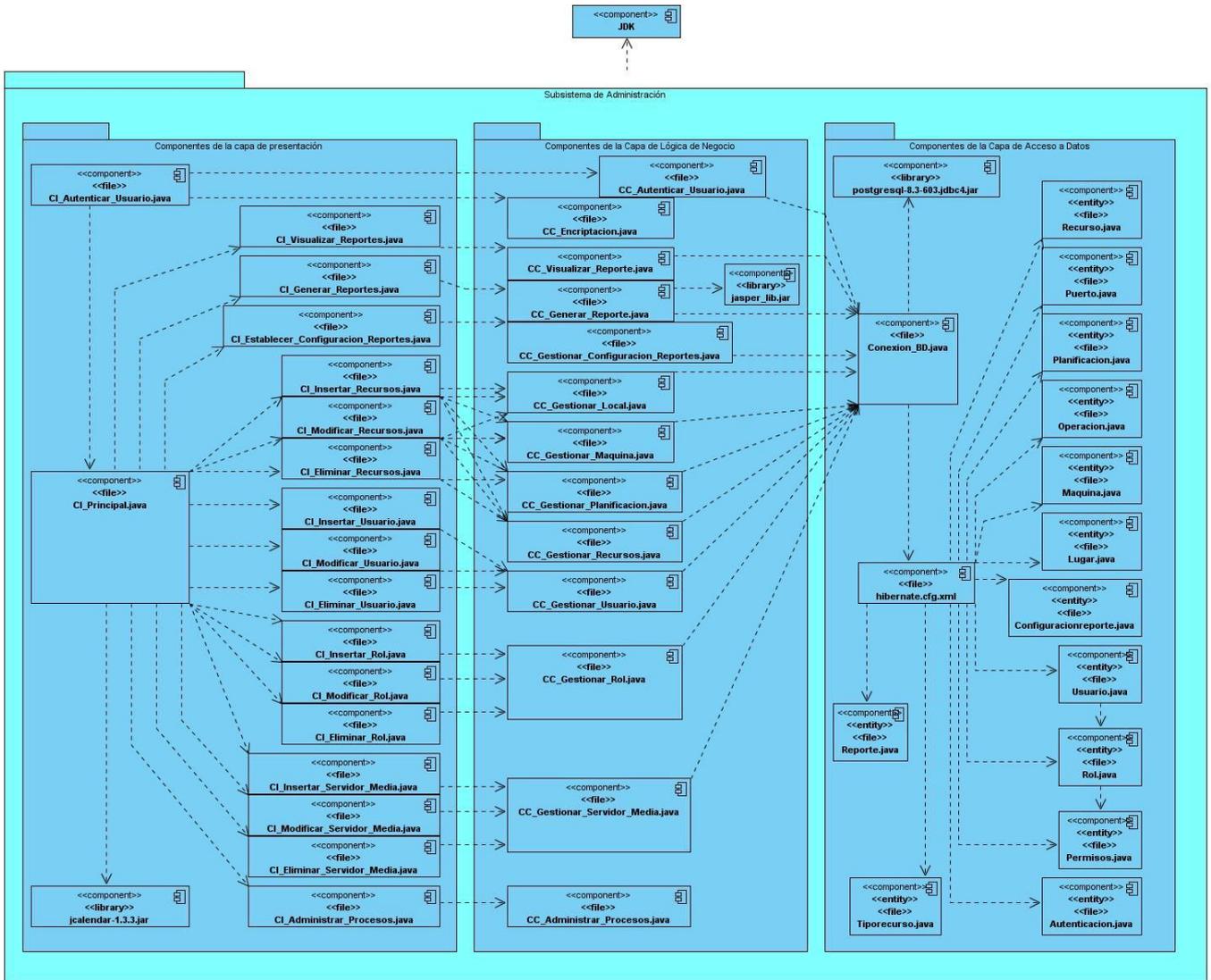


Figura 20 - Diagrama de componentes del subsistema de Administración





En esta parte del trabajo se presenta el estándar de codificación utilizado a la hora de implementar los subsistemas de Administración y Catalogación de Medias del producto Captura y Catalogación de Medias.

### ➤ **Criterios de Calidad**

Es importante que durante la codificación se consideren permanentemente los siguientes criterios de calidad:

Criterio	Objetivo
Facilidad de Comunicación	Proporcionar al usuario entradas y salidas fácilmente asimilables.
Auto descripción	Proporcionar en el código, explicaciones sobre la implantación realizada.
Simplicidad	La implantación realizada debe hacerse de la forma más comprensible posible.

Tabla 1 - Criterios de calidad durante la codificación

### ➤ **Nombramiento de Elementos**

Elemento	Reglas de nombramiento
Clases, interfaces y archivos fuente	Nombre sustantivo singular, con la primera letra en mayúscula y las demás en minúsculas.
Variables	Nombre sustantivo en mayúsculas. Para separar palabras se usará el guión bajo: _
Paquetes y directorios	Nombre sustantivo singular en minúsculas

Tabla 2 - Nombramiento de elementos

### ➤ **Principios Generales**

- Los nombres de cada uno de los elementos del programa deben ser significativos; su nombre debe explicar en lo posible el uso del elemento.
- No manejar en los programas más de una instrucción por línea.
- Declarar las variables en líneas separadas.
- Añadir comentarios descriptivos junto a cada declaración de variables, si es necesario.



## CAPÍTULO 2: “DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

- La mayoría de los elementos se deben nombrar usando sustantivos.
- Los atributos deben comenzar con letra minúsculas y los métodos deben comenzar con letra mayúsculas.
- Los parámetros deben comenzar con la letra p y con nombre lo más similar posible al atributo que se refiere.

### ➤ **Comentarios**

Cada programa deberá comenzar con un comentario que incluya:

- Autor
- Fecha
- Objetivo, o problema que resuelve el programa
- Algoritmo.

Cada función debe tener un encabezado que contenga:

- Objetivo de la función y no descripción del procedimiento.
- Comentarios de apoyo a variables, llamadas a función o inclusión de archivos que no sean obvios al proceso.
- Explicación de uso de argumentos (parámetros) no obvios.
- Explicación de uso de valores devueltos (de retorno).

Se utilizará dos tipos de comentarios.

*/\* Se puede utilizar en el caso de un atributo, de un método o algún segmento de código \*/*

Otros Comentarios Internos

Los otros comentarios necesarios deben figurar a partir de la columna 35.

// .....

//.....

### ➤ **Nombres de identificadores.**

Se considera como identificador a los nombres de variables (arreglos, matrices, apuntadores), funciones, así como cualquier tipo de dato definido por el usuario (estructura, clase). Dichos identificadores deberán seguir las siguientes normas, además de las definidas por el propio lenguaje.



## CAPÍTULO 2: “DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

- Deberán tener un nombre significativo para que por su simple lectura, pueda conocerse su función, sin tener que consultar manuales o hacer demasiados comentarios.
- Para nombres que se usen con frecuencia o para términos largos, se recomienda usar abreviaturas estándar para que éstos tengan una longitud razonable. Si usa abreviaturas deben manejar la misma lógica en todo el programa.
- Evitar identificadores que comiencen con uno o dos caracteres de subrayado para evitar que se confundan con los que el compilador selecciona.
- Cada identificador de función, variable o procedimiento deberá ser precedido por la abreviación del tipo de dato de que es la variable, o si se trata de una función o procedimiento del tipo de dato que regresa.

### ➤ **Identificadores de variables**

Comenzarán siempre con la primera letra minúscula correspondiente a su tipo de dato.

Para distinguir palabras dentro del nombre deberá emplearse un guión bajo (\_).

Ejemplo:

tipo\_de\_media

## **2.5 Conclusiones**

En el presente capítulo se presentó la descripción de los subsistemas de Administración, Catalogación de Medias y sus funcionalidades. Se representó el modelo de Implementación del sistema Captura y Catalogación de Medias así como el funcionamiento interno de dos de sus módulos a través de los diagramas de componentes. Se especificó y caracterizó el estándar de codificación utilizado en la implementación de los subsistemas. Estando el software en funcionamiento, se concluye enunciando que debería pasar a la etapa de pruebas y realizar los ensayos necesarios para asegurar el sistema libre de no conformidades y con la debida calidad.



## CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

### 3.1 Introducción

Un producto de software debe pasar por un intensivo proceso de refinamiento para poder alcanzar la eficacia pretendida. Uno de los mayores problemas que se afrontan en el desarrollo de software es la calidad de los mismos. Por lo que el proceso de prueba es sin duda uno de los aspectos fundamentales para medir el estado de una aplicación informática. La ausencia de errores no puede ser certificada a través de las pruebas, sino que a través de este procedimiento se puede demostrar la existencia de defectos en el software. En este capítulo se especifica todo lo relacionado a la validación de los subsistemas para garantizar el correcto funcionamiento de los mismos.

### 3.2 Pruebas de Software

Prueba del software es el proceso usado para determinar calidad de una aplicación informática. Es una investigación técnica orientada a proveer herramientas de obtención de información sobre la calidad del producto o del servicio bajo prueba. Las técnicas para encontrar problemas en un programa son extensamente variadas y van desde el uso del ingenio por parte del personal de prueba hasta herramientas automatizadas que ayudan a aliviar el peso y el costo de tiempo de esta actividad.

Con la realización de estas pruebas se pretende encontrar y documentar los defectos que puedan afectar la calidad del software y validar los requisitos que debe cumplir el software y a su vez que estos fueron implementados correctamente.

Todo producto puede ser probado de las siguientes formas:

- Minucioso examen de los detalles procedimentales del código a evaluar, por lo que es necesario conocer la lógica del programa.
- Pruebas sobre la interfaz del programa a probar, entendiendo por interfaz las entradas y salidas de dicho programa. No es necesario conocer la lógica del programa, únicamente la funcionalidad que debe realizar. **(24)**

#### 3.2.1 Pruebas Unitarias

En las pruebas unitarias se analiza una porción de código de manera aislada, como por ejemplo funciones y métodos. A los que después de pasarle unos parámetros de entrada se debe obtener otros parámetros de salida claramente definidos, es decir, validan la forma en la que las funciones y métodos trabajan en cada caso particular. Las pruebas unitarias se encargan de un único caso cada



### CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

vez, lo que significa que un único método puede necesitar varias pruebas unitarias si su funcionamiento varía en función del contexto. **(25)**

El objetivo de las pruebas unitarias es el aislamiento de partes del código y la demostración de que estas partes no contienen errores. Una vez creados el conjunto de pruebas unitarias sobre un fragmento de código los beneficios obtenidos, incluso antes de ejecutar ninguna prueba, son múltiples.

- **Simplificación de la integración:** Las pruebas unitarias eliminan las posibles incertidumbres y errores en lo que se espera de cada una de las unidades ayudando a entender la integración de cada una de las partes.
- **Refactorización de código:** Una vez refactorizado el código; las mismas pruebas unitarias nos pueden servir para probar el nuevo código asegurándonos de que este sigue siendo válido bajo la nueva implementación.
- **Documentación:** Las pruebas unitarias sirven como método de documentación mismo. Los desarrolladores pueden ver a través de las pruebas unitarias cual es el objetivo de las distintas partes del código de una manera básica.
- **Diseño:** Cuando se desarrolla el software las pruebas unitarias pueden tomar el lugar del diseño formal. Cada prueba unitaria puede ser vista como un elemento de diseños que especifica las clases, los métodos y el comportamiento observable de la aplicación.

#### 3.2.2 Pruebas de Caja Blanca

La prueba de caja blanca, denominada a veces prueba de caja de cristal es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para obtener los casos de prueba. Mediante los métodos de prueba de caja blanca, el ingeniero del software puede obtener casos de prueba que garanticen que se ejercita por lo menos una vez todos los caminos independientes de cada módulo, ejerciten todas las decisiones lógicas en sus vertientes verdadera y falsas; ejecuten todos los ciclos en sus límites y con sus límites operacionales, y ejerciten las estructuras internas de datos para asegurar su validez.

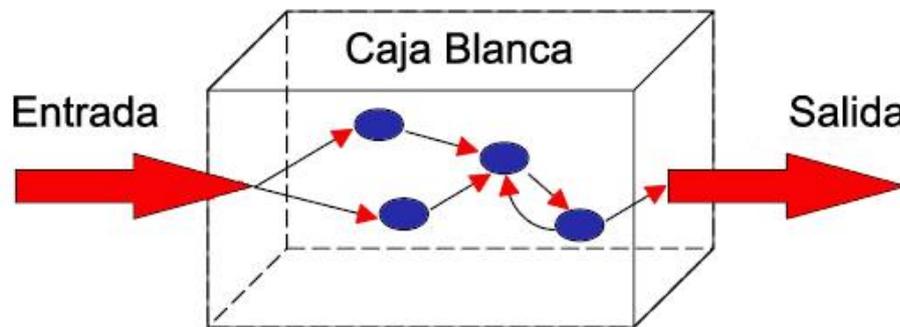


Figura 22 – Prueba de Caja Blanca

Con este método se determina cuáles son los casos de prueba a partir del código fuente del software y se utilizan las especificaciones para determinar el resultado esperado del caso. Los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa. La prueba de caja blanca del software se basa en el minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que ejerciten conjuntos específicos de condiciones y/o ciclos. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado o mencionado. **(26)**

### **Métodos de Prueba de Caja Blanca**

**Prueba del camino básico:** es una técnica propuesta inicialmente por Tom McCabe, la cual le permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. Los casos de prueba obtenidos del conjunto básico garantizarán que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa.

Algunos elementos y conceptos utilizados alrededor de éste método son los siguientes:

- *Grafo de flujo o grafo del programa:* representa el flujo de control lógico de un programa y se utiliza para trazar más fácilmente los caminos de éste. (Cada nodo representa una o más sentencias procedimentales y cada arista representa el flujo de control)
- *Complejidad ciclomática:* es una métrica de software que proporciona una medición cuantitativa de la complejidad lógica de un programa. Cuando se usa en el contexto de las pruebas, el cálculo de la complejidad ciclomática representa el número de caminos independientes del conjunto básico de un



### CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

programa. Esta medida ofrece al probador de software un límite superior para el número de pruebas que debe realizar para garantizar que se ejecutan por lo menos una vez cada sentencia.

- *Camino independiente*: cualquier camino del programa que introduce, por lo menos, un nuevo conjunto de sentencias de proceso o una nueva condición.

De forma general, los pasos que se debe seguir para la obtención de los casos de prueba en este método, son los siguientes:

1. Emplear el diseño o el código para elaborar el grafo de flujo.
2. Determinar la complejidad ciclomática del grafo de flujo.
3. Determinar un conjunto básico de caminos linealmente independientes.
4. Preparar los casos de prueba que forzarán la ejecución de cada camino del conjunto básico.

**Prueba de la estructura de control**: dentro de éste tipo de prueba se contempla el método del camino básico mencionado anteriormente pero además existen otras pruebas asociadas que permiten ampliar la cobertura de la prueba y mejorar su calidad. Estas son:

**Prueba de condición**: es un método de diseño de casos de prueba que ejercita las condiciones lógicas contenidas en el módulo de un programa. Algunos conceptos empleados alrededor de esta prueba son los siguientes:

- Condición simple: es una variable lógica o una expresión relacional ( $E1 < \text{operador} - \text{relacional} > E2$ ).
- Condición compuesta: está formada por dos o más condiciones simples, operadores lógicos y paréntesis.

En general los tipos de errores que se buscan en una prueba de condición, son los siguientes:

- Error en operador lógico (existencia de operadores lógicos incorrectos, desaparecidos, sobrantes).
- Error en variable lógica.
- Error en paréntesis lógico.
- Error en operador relacional.
- Error en expresión aritmética.

**Prueba del flujo de datos**: selecciona caminos de prueba de un programa de acuerdo con la ubicación de las definiciones y los usos de las variables del programa.

**Prueba de bucles**: es una técnica que se centra exclusivamente en la validez de las construcciones de bucles (bucles simples, anidados, concatenados y no estructurados). **(27)**



### 3.2.3 Prueba de Caja Negra

Las pruebas de caja negra se centran en lo que se espera de un módulo, es decir, intentan encontrar casos en que el módulo no se atiene a su especificación. Por ello se denominan pruebas funcionales, y el probador se limita a suministrarle datos como entrada y estudiar la salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro.



Figura 23 – Prueba de Caja Negra

Las pruebas de caja negra se apoyan en la especificación de requisitos del módulo. De hecho, se habla de "cobertura de especificación" para dar una medida del número de requisitos que se han probado. Es fácil obtener coberturas del ciento por ciento en módulos internos, aunque puede ser más laborioso en módulos con interfaz al exterior. En cualquier caso, es muy recomendable conseguir una alta cobertura en esta línea. **(28)**

#### Métodos de Prueba de Caja Negra

Para confeccionar los casos de prueba de caja negra existen diversos métodos, algunos de ellos son:  
**Métodos de prueba basados en grafos:** en este método se debe entender los objetos (objetos de datos, objetos de programa tales como módulos o colecciones de sentencias del lenguaje de programación) que se modelan en el software y las relaciones que conectan a estos objetos. Una vez que se ha llevado a cabo esto, el siguiente paso es definir una serie de pruebas que verifiquen que todos los objetos tienen entre ellos las relaciones esperadas. En este método:

- 1.- Se crea un grafo de objetos importantes y sus relaciones.
- 2.- Se diseña una serie de pruebas que cubran el grafo de manera que se ejerciten todos los objetos y sus relaciones para descubrir errores.



### CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

Se describe un número de modelados para pruebas de comportamiento que pueden hacer uso de los grafos:

- *Modelado del flujo de transacción:* Los nodos representan los pasos de alguna transacción (por ejemplo, los pasos necesarios para una reserva en una línea aérea usando un servicio en línea), y los enlaces representan las conexiones lógicas entre los pasos (por ejemplo, vuelo.información.entrada es seguida de validación /disponibilidad.procesamiento).

- *Modelado de estado finito:* Los nodos representan diferentes estados del software observables por el usuario (por ejemplo, cada una de las pantallas que aparecen cuando un telefonista coge una petición por teléfono), y los enlaces representan las transiciones que ocurren para moverse de estado a estado (por ejemplo, petición-información se verifica durante inventario-disponibilidad-búsqueda y es seguido por cliente-factura-información-entrada).

- *Modelado de flujo de datos:* Los nodos objetos de datos y los enlaces son las transformaciones que ocurren para convertir un objeto de datos en otro.

Modelado de planificación: Los nodos son objetos de programa y los enlaces son las conexiones secuenciales entre esos objetos. Los pesos de enlace se usan para especificar los tiempos de ejecución requeridos al ejecutarse el programa.

- *Grafo Causa-efecto:* El grafo Causa-efecto representa una ayuda gráfica en seleccionar, de una manera sistemática, un gran conjunto de casos de prueba. Tiene un efecto secundario beneficioso en precisar estados incompletos y ambigüedades en la especificación. **(27)**

**Partición equivalente:** Pressman presenta la partición equivalente como un método de prueba de caja negra que divide el campo de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba. Un caso de prueba ideal descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar.

Una clase de equivalencia representa un conjunto de estados válidos o no válidos para condiciones de entrada. Típicamente, una condición de entrada es un valor numérico específico, un rango de valores, un conjunto de valores relacionados o una condición lógica.

El objetivo de partición equivalente es reducir el posible conjunto de casos de prueba en uno más pequeño, un conjunto manejable que evalúe bien el software. Se toma un riesgo porque se escoge no probar todo. Así que se necesita tener mucho cuidado al escoger las clases.



### CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

La partición equivalente es subjetiva. Dos probadores quienes prueban un programa complejo pueden llegar a diferentes conjuntos de particiones.

En el diseño de casos de prueba para partición equivalente se procede en dos pasos:

1.- Se identifican las clases de equivalencia. Las clases de equivalencia son identificadas tomando cada condición de entrada (generalmente una oración o una frase en la especificación) y repartiéndola en dos o más grupos.

Es de notar que dos tipos de clases de equivalencia están identificados: las clases de equivalencia válidas representan entradas válidas al programa, y las clases de equivalencia inválidas que representan el resto de los estados posibles de la condición (es decir, valores erróneos de la entrada).

2.- Se define los casos de prueba. El segundo paso es el uso de las clases de equivalencia para identificar los casos de prueba. El proceso es como sigue: se asigna un número único a cada clase de equivalencia. Hasta que todas las clases de equivalencia válidas han sido cubiertas por los casos de prueba, se escribe un nuevo caso de prueba que cubra la clase de equivalencia válida. Y por último hasta que los casos de prueba hayan cubierto todas las clases de equivalencia inválidas, se escribe un caso de la prueba que cubra una, y solamente una, de las clases de equivalencia inválidas descubiertas. **(27)**

**Análisis de valores límite:** los errores tienden a darse más en los límites del campo de entrada que en el centro. Por ello, se ha desarrollado el análisis de valores límites (AVL) como técnica de prueba. El análisis de valores límite lleva a una elección de casos de prueba que ejerciten los valores límite.

El análisis de valores límite es una técnica de diseño de casos de prueba que completa a la partición equivalente. En lugar de seleccionar cualquier elemento de una clase de equivalencia, el AVL lleva a la elección de casos de prueba en los extremos de la clase. En lugar de centrarse solamente en las condiciones de entrada, el AVL obtiene casos de prueba también para el campo de salida.

Condiciones sublímite. Las condiciones límite normales son las más obvias de descubrir. Estas son definidas en la especificación o son evidentes al momento de utilizar el software. Algunos límites, sin embargo, son internos al software, no son necesariamente aparentes al usuario final pero aún así deben ser probadas por el probador. Estas son conocidas como condiciones sublímites o condiciones límite internas. Una condición sublímite común es la tabla de caracteres ASCII, por ejemplo, si se está evaluando una caja de texto que acepta solamente los caracteres AZ y az, se debe incluir los valores en la partición inválida justo «debajo de» y «encima de» esos caracteres de la tabla ASCII, [", y {. **(27)**



## CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

### 3.3 Pruebas de Software a los Subsistemas

A los subsistemas de Administración y Catalogación de Medias del producto Captura y Catalogación de Medias se le aplicarán específicamente las pruebas de caja negra con la finalidad de saber la salida que brindan luego de ser sometidos a varias entradas, su rendimiento y comportamiento. Luego de este proceso se podrá pasar a la rectificación de errores, para lograr una mayor calidad en el producto.

#### 3.3.1 Pruebas del subsistema de Administración

A partir de las pruebas de caja negra realizadas al subsistema de Administración del Producto Captura y Catalogación de Medias del Departamento de Señales Digitales, se han obtenido una serie de no conformidades, que se muestran a continuación en una tabla en donde se registran dichos defectos.

#### Elementos probados:

- Autenticar Usuario
- Gestionar roles y permisos de usuario.
- Gestionar usuario.
- Planificar procesos.
- Administrar procesos.
- Gestionar servidores de medias.
- Gestionar reportes.
- Visualizar reportes.
- Gestionar máquinas
- Gestionar recursos
- Gestionar local.

#### Registro de defectos y dificultades encontradas en el subsistema de Administración

Elemento	No.	No conformidad	Aspecto correspondiente	Clasificación	Estado NC
Gestionar Rol Completo Eliminar rol Modificar rol Eliminar rol	1	Tiene un mensaje estándar, para todo el caso de uso en caso de cualquier falla de escenario.	Se recomienda, especificar un mensaje transmitiendo el tipo de falla, esto para cada sección, en caso de que todos los campos fueran vacíos, en caso de tener descripción vacío, en caso de tener nombre vacío, en caso de no haber especificado algún	Recomendación	13\Abril\2010 PD: Resuelto



### CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

			permiso, y así sucesivamente, para modificar, y eliminar		
Modificar Rol	2	No Muestra información del rol a modificar	Al seleccionar el rol, que se desea modificar, no muestra ninguno de los datos, que se habían insertado anteriormente del este rol.	Correspondencia con la documentación	13\Abril\2010 PD: Resuelto
Modificar Rol	3	No muestra un mensaje, en caso de cuando se valide los datos, ya sean estos lo que estén anteriormente en la base datos	Se recomienda que cuando se selecciona, un rol a modificar, y le pongo los mismos permisos que tenía anteriormente en la BD, me muestre un mensaje donde me especifique que estos datos es lo que el rol, ya tiene en la BD	Recomendación	13\Abril\2010 PD: Resuelto
Mensaje de error al insertar una planificación	4	Mensaje de la aplicación cambiar fecha o recursos.	Se recomienda, cambiar el mensaje, ya que hay un campo recurso, en la sección de captura de video, y no es este el recurso que se necesita cambiar.	Recomendación	13\Abril\2010 PD: Resuelto
Campo de texto. Espacio cuando se encuentra en blanco	5	Muestra mensaje de no conexión	Al tratar de insertar un servidor, con el campo de texto Espacio, y este se encuentra en blanco, no me inserta, pero me muestra el mensaje de no conexión.	Validación	13\Abril\2010 PD: Resuelto

Tabla 3 - Pruebas al Subsistema de Administración. Defectos y dificultades encontradas

#### 3.3.2 Pruebas al subsistema de Catalogación de Medias

Como resultado de las pruebas de Caja Negra realizadas al subsistema de Video del Producto Captura y Catalogación de Medias del Departamento de Señales Digitales, se detectaron una serie de problemas los cuales son mostrados en este epígrafe.

Elementos probados:

- Autenticar Usuario
- Buscar Media
- Visualizar Reportes



### CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

- Exportar Media
- Reproducir Medias
- Editar Video
- Catalogar Media

#### Registro de defectos y dificultades encontradas en el subsistema de Catalogación de Medias

Elemento	No.	No conformidad	Aspecto correspondiente	Clasificación	Estado NC
En el caso de uso Autenticar usuario	1	Posición del campo autenticación	La ventana que permite el acceso al sistema debe estar situada en el medio de la pantalla.	R: Recomendación	6/abril/2010 PD: Resuelto
En el caso de uso Buscar media.	2	Criterios de búsqueda en búsqueda avanzada	El campo de fecha debería ser del formato especificado ejemplo Calendario.	R: Recomendación	6/abril/2010 PD: Resuelto
En el caso de uso Buscar media.	3	Direccionamiento en el menú principal	En el menú si se escoge la opción búsqueda avanzada mostrar interfaz correspondiente.	NS: No Significativa	6/abril/2010 PD: Resuelto
En el caso de uso Exportar media	4	Cancelar la exportación de la media	Si se desea cancelar el proceso cuando se está exportando se cierra la aplicación.	S: Significativa	6/abril/2010 PD: Pendiente
En el caso de uso Exportar media	5	Cuando se va a exportar y no hay conexión con la base de datos	Si se desea exportar una media y no hay conexión con la base de datos no se maneja el mensaje de error	S: Significativa	6/abril/2010 PD: Resuelto
En el caso de uso Reproducir Media	6	Cuando se reproduce una media	El sistema reproduce la media correctamente pero después no se puede ejecutar ninguna	S: Significativa	6/abril/2010 PD: Resuelto



### CAPÍTULO 3: “VALIDACIÓN DE LA SOLUCIÓN PROPUESTA”

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

			funcionalidad pues se cierra la aplicación.		
En el caso de uso Reproducir Media	7	Fallo en la conexión al servidor de base de datos	Si no existe conexión con la base de datos no muestra ningún mensaje advirtiendo al usuario del error ocurrido.	S: Significativa	6/abril/2010 PD: Pendiente
En el caso de uso Reproducir Media	8	Fallo en la conexión al servidor de streaming.	Si no existe conexión con el servidor de streaming no muestra ningún mensaje advirtiendo al usuario del error ocurrido.	S: Significativa	6/abril/2010 PD: Resuelto
En el caso de uso Editar Video escenario editar video correctamente	9	No se actualizan los tiempos inicio ni fin	En el reproductor de editar a la media se le deben actualizar los tiempos inicio y fin.	NS: No Significativa	6/abril/2010 PD: Resuelto
En el caso de uso Editar Video escenario editar video correctamente	10	El botón de salvar media editada no es el que lleva.	En el reproductor de editar media cuando se desea salvar la media editada el botón no se corresponde. (Establecer algo que identifique la funcionalidad)	NS: No Significativa	6/abril/2010 PD: Resuelto
En el caso de uso Editar Video escenario fallo de conexión con la base de datos	11	Fallo en la conexión al servidor de base de datos	Si no existe conexión con la base de datos no muestra ningún mensaje advirtiendo al usuario del error ocurrido.	S: Significativa	6/abril/2010 PD: Resuelto
En el caso de uso Catalogar media. escenario catalogación correctamente	12	Muestra un mensaje de un error.	Cuando se cataloga una media se emite el mensaje de media catalogada pero también muestra un mensaje de error.	NS: No Significativa	6/abril/2010 PD: Resuelto
En el caso de uso Visualizar Reporte escenario visualizar reporte incorrectamente.	13	No se maneja el error ocurrido de la búsqueda.	Cuando se va a mostrar un reporte y se selecciona un criterio de búsqueda que no encuentra resultados a mostrar no se maneja el error ocurrido.	S: Significativa	6/abril/2010 PD: Resuelto



En el caso de uso Visualizar Reporte escenario fallo en la conexión al servidor de base de datos.	14	Fallo en la conexión al servidor de base de datos	Si no existe conexión con la base de datos no muestra ningún mensaje advirtiendo al usuario del error ocurrido.	S: Significativa	6/abril/2010 PD: Resuelto

Tabla 4 - Pruebas al Subsistema de Catalogación de Medias. Defectos y dificultades encontradas

### 3.4 Conclusiones

La fase de pruebas absorbe una buena porción de los costes de desarrollo de software. Su ejecución se basa en metodologías (reglas que se les dan a los encargados de probar) que se van desarrollando con la experiencia. Luego de haber investigado varias técnicas para realizar las pruebas de software y haber aplicado una en específico a los subsistemas de Administración y Catalogación de Medias se lograron detectar un conjunto de defectos y no conformidades. Darle solución a dichas dificultades garantizará una mayor calidad del producto informático y le aportará robustez al mismo.



## CONCLUSIONES

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

## CONCLUSIONES

Durante el desarrollo de la investigación se estudiaron los procesos principales de los subsistemas en cuestión, la catalogación de las medias y la gestión administrativa del sistema. Se investigó sobre las principales técnicas de programación y sobre las herramientas a utilizar para implementar las funcionalidades previstas, lo cual aportó un gran conocimiento empleado a la hora de desarrollar. Se realizaron los diagramas de componentes de cada uno de los subsistemas reflejando la relación de dependencia entre cada una de las clases y otros componentes que intervienen en la composición de los mismos, además apoyado en dichos diagramas se generó el modelo de implementación en donde se manifiesta la interrelación dentro del sistema como un todo. Se definió un estándar de codificación con el cual se logra una mayor uniformidad y legibilidad del código. Se certificaron las funcionalidades al realizarle las pruebas para la validación de la solución propuesta, esto permitió que se detectaran fallas y errores que fueron corregidos. Se puede afirmar que se cumplió con el objetivo trazado desde el inicio de la investigación. Esto se sustenta en el desarrollo culminado de los subsistemas de Catalogación de Medias y de Administración del Sistema Captura y Catalogación de Medias. En el caso del subsistema de Catalogación podría reportar grandes beneficios para el país no solamente en la parte económica. Su utilización en entidades nacionales como el ICRT, la Escuela Latinoamericana de Cine y en la propia Televisión Universitaria de la UCI podría representar un avance significativo en cuanto a la optimización del personal y el tiempo de trabajo, garantizando también la automatización de varios procesos que hasta hoy se hacen de forma manual.



## RECOMENDACIONES

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

## RECOMENDACIONES

Se recomienda el estudio del envío de señales entre aplicaciones con tecnologías como Corba<sup>1</sup> o Ice<sup>2</sup>, pues actualmente todos los procesos se realizan leyendo constantemente de la Base de Datos, lo que puede llegar a saturar la red con el envío continuo de peticiones y respuestas. Es necesaria la continuación del trabajo en una nueva versión del subsistema de Catalogación de Medias en donde se enfatice en la utilización de componentes más ligeros y compatibles dentro de la aplicación implementada en Java.

---

<sup>1</sup> (Arquitectura Común de Intermediarios en Peticiones a Objetos), es un estándar que establece una plataforma de desarrollo de sistemas distribuidos facilitando la invocación de métodos remotos bajo un paradigma orientado a objetos.

<sup>2</sup> (Internet Communications Engine) es un middleware libre, con licencia GPL, desarrollado por la empresa ZeroC y cuya motivación es responder a la complejidad de CORBA ofreciendo una funcionalidad similar.



## BIBLIOGRAFÍA CONSULTADA

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

## BIBLIOGRAFÍA CONSULTADA

1. Bonanata, Maximiliano. ***Programación y algoritmos***. Buenos Aires : MP Ediciones S.A, 2003.
2. García de Jalón, Javier, et al. ***Aprenda Java como si estuviera en primero***. San Sebastián : s.n., 2000.
3. Jacobson, Ivar, Rumbaugh, James and Booch, Grady. ***El Proceso Unificado de Desarrollo de Software***. Madrid : Pearson Educación S.A, 2000. ISBN: 84-7829-036-2.
4. Rumbaugh, James, Jacobson, Ivar and Booch, Grady. ***El Lenguaje Unificado de Modelado. Manual de Referencia***. Madrid : Pearson Educación S.A, 2000.
5. Torossi, Gustavo. ***El Proceso Unificado de Desarrollo de Software***.
6. Alvarez de Zayas, Carlos. ***Metodología de la Investigación Científica***. Santiago de Cuba : s.n., 1995.
7. Pressman, Roger. ***Ingeniería de Software, Un enfoque práctico***.



## REFERENCIAS BIBLIOGRÁFICAS

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

## REFERENCIAS BIBLIOGRÁFICAS

1. Díaz Berenguer, Abel y Román Vieito, Alberto Ramón. **Sistema de Administración y Configuración de la solución de Captura y Catalogación de Medias**. Ciudad Habana : s.n., 2009.
2. Almeida Maldonado, Enrique y Baby González, Ronald. **Sistema de Catalogación de Medias**. Ciudad Habana : s.n., 2009.
3. DAEDALUS. **Universidades Politécnica y Autónoma de Madrid (UPM y UAM)**. [En línea] [Citado el: 5 de Noviembre de 2009.] [http://www.daedalus.es/inteligencia-de-negocio/sistemas-complejos/ciencia-de-sistemas/que-es-un-sistema/..](http://www.daedalus.es/inteligencia-de-negocio/sistemas-complejos/ciencia-de-sistemas/que-es-un-sistema/)
4. RAE. **Real Academia de la Lengua Española**. [En línea] [Citado el: 5 de Noviembre de 2009.] [http://buscon.rae.es/drae/SrvltConsulta?TIPO\\_BUS=3&LEMA=Administrar](http://buscon.rae.es/drae/SrvltConsulta?TIPO_BUS=3&LEMA=Administrar).
5. RAE. **Real Academia de la Lengua Española**. [En línea] [Citado el: 5 de Noviembre de 2009.] [http://buscon.rae.es/drae/SrvltConsulta?TIPO\\_BUS=3&LEMA=Catalogar](http://buscon.rae.es/drae/SrvltConsulta?TIPO_BUS=3&LEMA=Catalogar).
6. Orienta. **Instituto Mexicano de Orientación y Evaluación Educativa**. [En línea] [Citado el: 6 de Noviembre de 2009.] <http://www.orienta.org.mx/biblioteca/catalogacion.html>.
7. HIRU. **Universidades e Investigación del Gobierno Vasco (Departamento de Educación)**. [En línea] [Citado el: 9 de Noviembre de 2009.] [http://www.hiru.com/es/gizarte\\_zientziak/gz\\_00650.html](http://www.hiru.com/es/gizarte_zientziak/gz_00650.html).
8. Bonanata, Maximiliano. **Programación y algoritmos**. Buenos Aires : MP Ediciones S.A, 2003.
9. Álvarez, Sara. **Desarrollo Web**. [En línea] 18 de Mayo de 2006. [Citado el: 9 de Noviembre de 2009.] <http://www.desarrolloweb.com/articulos/2477.php>.
10. UBU. **Universidad de Burgos**. [En línea] [Citado el: 15 de Noviembre de 2009.] [http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/l\\_1.htm](http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/l_1.htm).
11. Schaeffer, Elisa. UANL. **Universidad Autónoma de Nuevo León**. [En línea] 2007. [Citado el: 15 de Noviembre de 2009.] [it.ciidit.uanl.mx/~elisa/teaching/prog/herencia.pdf](http://it.ciidit.uanl.mx/~elisa/teaching/prog/herencia.pdf).
12. García de Jalón, Javier, y otros. **Aprenda Java como si estuviera en primero**. San Sebastián : s.n., 2000.
13. Álvarez Marañón, Gonzalo. CSIC. **Instituto de Física Aplicada del CSIC**. [En línea] 1999. [Citado el: 26 de Noviembre de 2009.] <http://www.iec.csic.es/CRIPTONOMICON/java/quesjava.html>.
14. Universidad Ponpeu Fabra. **Departamento de las tecnologías de la información y las comunicaciones**. [En línea] [Citado el: 26 de Noviembre de 2009.] <http://www.dtic.upf.edu/~vlopez/docencia/castellano/todoshtml/Plt3cas/node16.html>.



## REFERENCIAS BIBLIOGRÁFICAS

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

15. Java. **SUN Microsystems**. [En línea] [Citado el: 3 de Diciembre de 2009.] <http://java.com/es/about/> .
16. Cerda, Felipe. **Techbloop. Tendencias Tecnológicas en Español**. [En línea] [Citado el: 3 de Diciembre de 2009.] [www.techbloop.com/talks/netbeans65es\\_cl.pdf](http://www.techbloop.com/talks/netbeans65es_cl.pdf).
17. SUN Microsystems. **SUN Microsystems**. [En línea] [Citado el: 3 de Diciembre de 2009.] [http://www.sun.com/emrkt/innercircle/newsletter/latam/0207latam\\_feature.html](http://www.sun.com/emrkt/innercircle/newsletter/latam/0207latam_feature.html).
18. UNIFÉ. **Universidad Femenina del Sagrado Corazón** . [En línea] [Citado el: 4 de Diciembre de 2009.] [www.unife.edu.pe/ing/desarrollo.doc](http://www.unife.edu.pe/ing/desarrollo.doc).
19. Quiñones, Ernesto. **EQSOFT. Consultoría y Soporte EIRL**. [En línea] [Citado el: 5 de Diciembre de 2009.] [www.eqsoft.net/presentas/introduccion\\_a\\_postgresql.pdf](http://www.eqsoft.net/presentas/introduccion_a_postgresql.pdf).
20. Espinoza, Humberto. **LGS. Links Global Service**. [En línea] [Citado el: 5 de Diciembre de 2009.] [www.lgs.com.ve/pres/PresentacionES\\_PSQL.pdf](http://www.lgs.com.ve/pres/PresentacionES_PSQL.pdf).
21. Jacobson, Ivar, Rumbaugh, James y Booch, Grady. **El Proceso Unificado de Desarrollo de Software**. Madrid : Pearson Educación S.A, 2000. ISBN: 84-7829-036-2.
22. Rumbaugh, James, Jacobson, Ivar y Booch, Grady. **El Lenguaje Unificado de Modelado. Manual de Referencia**. Madrid : Pearson Educación S.A, 2000.
23. Torres, Fer. **dosmilmastres**. [En línea] 21 de Septiembre de 2006. [Citado el: 12 de Enero de 2010.] <http://www.dosmilmastres.com/blog/revisiones-de-codigo-y-estndares-de-codificacin/>.
24. Cruz Guerrero, Mauris Yadira y Santana Pacheco, Yudisel. **Implementación de un Sistema de Control de Pago para el Departamento de Transportaciones Nacionales. Trabajo**. Ciudad Habana : s.n., 2009.
25. Villar Matos, Heikel. **Implementación de los subsistemas de Administración y Transmisión de la Plataforma de Televisión Informativa PRIMICIA**. Ciudad Habana : s.n., 2008.
26. Universidad Simón Bolívar, Edificio de Matemáticas y Sistemas. **LDC**. [En línea] 2008. [Citado el: 1 de Mayo de 2010.] <http://www ldc.usb.ve/~teruel/ci3711/test1/index.html>.
27. Rojas, Johanna y Barrios, Emilio. **Universidad Distrital Francisco Jose de Caldas**. [En línea] 2007. [Citado el: 3 de Mayo de 2010.] <http://www.udistrital.edu.co/comunidad/grupos/arquisoft/fileadmin/Estudiantes/Pruebas/HTML%20-%20Pruebas%20de%20software/node26.html>.
28. Mañas, José A. **Departamento de Ingeniería de Sistemas Telmáticos**. [En línea] 16 de Marzo de 1994 . [Citado el: 3 de Mayo de 2010.] <http://www.lab.dit.upm.es/~lprg/material/apuntes/pruebas/testing.htm#s22>.



## GLOSARIO

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

## GLOSARIO

**Aplicación:** Es una clase de programa informático creado para facilitar al usuario un determinado tipo de trabajo. Esto lo caracteriza frente a otros programas como los sistemas operativos, las utilidades y los lenguajes de programación.

**Audio:** Señal, técnica o sistema electrónico relacionado con la grabación, reproducción y transmisión del sonido que normalmente se encuentra acotado al rango de frecuencias audibles para los seres humanos.

**Bucle:** es un ciclo en términos de programación, es una sentencia que se realiza repetidas veces a un trozo aislado de código, hasta que la condición asignada a dicho bucle deje de cumplirse.

**Debugger:** es un programa que permite depurar o limpiar los errores de otro programa informático.

**Ficheros:** Directorios. Agrupación de archivos de datos, atendiendo a su contenido, a su propósito o a cualquier otro criterio.

**Metadatos:** literalmente “sobre datos”, son datos que describen otros datos.

**Multimedia:** El término multimedia se utiliza para referirse a cualquier objeto o sistema que utiliza múltiples medios de expresión (físicos o digitales) para presentar o comunicar información.

**Multiplataforma:** Se utiliza el término para denominar a los programas, lenguajes de programación u otra clase de software que pueden brindar sus prestaciones funcionando sobre diversas combinaciones de hardware y software.

**Servidor:** Software u ordenador que provee servicios a otros programas o equipos denominados clientes.

**Streaming:** consiste en la distribución de audio o video por Internet.



## GLOSARIO

Subsistemas de Administración y Catalogación del Sistema de Captura y Catalogación de Medias (CCM).

---

**Televisión:** es un sistema para la transmisión y recepción de imágenes en movimiento y sonido a distancia.

**Transcodificación:** conversión directa de digital a digital de un códec a otro, en general con pérdida de calidad.

**Video:** es la tecnología de la captación, grabación, procesamiento, almacenamiento, transmisión y reconstrucción por medios electrónicos digitales o analógicos de una secuencia de imágenes que representan escenas en movimiento.



## ANEXOS

### Anexo I: interfaz del Subsistema de Catalogación con todos los componentes

