

**Universidad de las Ciencias Informáticas**

**FACULTAD 6**



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

**Título: Subsistema de Solicitud y Préstamo de  
Agorav para Android.**

**Autor: Yusniel Almarales Avila**

**Tutor: Ing. Isleivys Rodríguez Pupo**

**La Habana, Julio de 2015**

**“Año 57 de la Revolución”**

## *Declaración de Autoría*

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

---

Firma del autor

Yusniel Almarales Avila

---

Firma Tutor

Ing. Isleivys Rodríguez Pupo

**Datos de contacto**

**Autor:**

Yusniel Almarales Avila  
Universidad de las Ciencias Informáticas,  
La Habana, Cuba

**E-mail:** yalmarales@estudiantes.uci.cu

**Tutor:**

Ing. Isleivys Rodríguez Pupo  
Universidad de las Ciencias Informáticas,  
La Habana, Cuba

**E-mail:** irodriguez@uci.cu



*No se mide el valor de un hombre por sus ropas o por los bienes que posee, el verdadero valor del hombre es su carácter, sus ideas y la nobleza de sus ideales.*

*Charles Chaplin*

## *Dedicatoria*

*Dedico este trabajo de diploma a mis padres, mis abuelas y a mi hermanita, por confiar en mi todo el tiempo y apoyarme en todo; a toda mi familia que de una forma u otra siempre me brindaron su apoyo incondicional.*

## *Agradecimientos*

A mi mamá Lili por ser la principal causa de mi existencia, apoyarme constantemente en todos los planes que me propongo. Por la educación brindada en el transcurso de mi vida y por enseñarme a ser quien soy.

A mi papá Julio porque desde pequeño me has apoyado en todo, la voz fuerte que me enseñó que tengo que seguir hacia adelante, no importan los obstáculos y que todo está en el empeño que le ponga a las tareas.

A mis abuelas Chabyn y Alva por ser mis segundas madres y consentirme en todo, por ser tan importantes en mi vida, por su apoyo incondicional en todo lo que necesité, por todo gracias.

A mi hermanita Yisel por llegar a mi vida cuando más necesitaba esa chispita que me animara con sus travesuras, sus besos, sus abrazos y su amor.

A toda mi familia por apoyarme y confiar en mí, por su preocupación durante mis años de estudio y su constante trabajo de darme ánimos para continuar.

A mi primo Jorge Luis, por compartir su vida desde la secundaria hasta ahora conmigo, en los estudios, en las diversiones y por estar siempre presente en los momentos difíciles.

A mis amigos Mauricio, Rolando, Yosvany, Alexander (El Cama) por compartir conmigo durante estos años, su confianza en todo momento y su apoyo incondicional.

A mis demás amigos que compartieron y se relacionaron conmigo a lo largo de estos años y a todos los que aportaron su granito de arena para que hoy yo sea la persona que soy.

A Yoendry por haberme apoyado y ayudado en todo lo que necesité con la tesis, sin ti no hubiese sido posible lograr esto.

A mi tutora Isleivys que aunque pasó a ser mi tutora hace poco, siempre se preocupó por mi tesis, me aconsejó y me ayudó en todo lo que estuvo a su alcance y a todos los profesores que participaron en mi formación como ingeniero.

Al tribunal y oponente por su valioso aporte con el cual contribuyeron a mejorar este trabajo.

Y para finalizar al Comandante Fidel Castro y a la Revolución por permitirme estudiar en esta gran Universidad.

## Resumen

Android es un sistema operativo (SO) orientado a dispositivos móviles además de ser un sistema abierto, multitarea, que permite a los desarrolladores acceder a las funcionalidades principales del dispositivo mediante aplicaciones, en la actualidad es uno de los SO de mayor distribución en el mundo. La presente investigación tiene como objetivo el desarrollo del subsistema de Solicitud y Préstamo del sistema Agorav para el SO Android. Para guiar el desarrollo del subsistema se utilizó como metodología el Proceso Unificado de Desarrollo (RUP, por sus siglas en inglés), Java y XML como lenguajes de programación y Eclipse 4.2.2 como Entorno de Desarrollo Integrado (IDE, por sus siglas en inglés). El subsistema de Solicitud y Préstamo en Android tiene buena accesibilidad y presenta una correcta visualización, erradica el problema de incompatibilidad entre los navegadores de los dispositivos móviles y los ordenadores, además permite la búsqueda de materiales audiovisuales, así como solicitar y descargar la media deseada. Esta solución elimina los problemas existentes en el Subsistema de Solicitud y Préstamo del sistema Agorav al ser accedido desde dispositivos móviles con SO Android.

**Palabras claves:** Agorav, Android, préstamo, solicitud, subsistema, visualización

## **Abstract:**

Android is an operating system (OS) mobile devices-oriented and an open and multitask system as well, that allows developers accessing the principal functionalities of the device through applications, today it is one of the most distributed OS in the world. The present work aims at developing a Loan and Request Subsystem from Agorav platform for Android OS. To conduct the development of the subsystem it was used the Rational Unified Process (RUP) as methodology, Java and XML as programming languages and Eclipse 4.2.2 as Integrated Development Environment (IDE). The Loan and Request Subsystem from Agorav platform in Android is well accessible and with a correct visualization, it avoids the compatibility problem among browsers in mobile devices and computers, it also permits searching audiovisual materials, as well as requesting and downloading the media desired. This solution eliminates the existing problems in the Loan and Request Subsystem from Agorav system by being accessed from mobile devices with Android OS.

**Keywords:** Agorav, Android, application, loan, subsystem, visualization

Introducción.....	1
Capítulo 1: Fundamentos teóricos del Subsistema de Solicitud y Préstamo del sistema Agorav para Android.....	6
1.1.    Conceptos asociados al dominio del problema.....	6
1.1.1. <i>Android</i> .....	6
<b>1.1.2.    Materiales Audiovisuales</b> .....	6
1.1.3. <i>Visualización</i> .....	7
1.1.4. <i>Recuperación de materiales audiovisuales</i> .....	8
1.1.5. <i>Solicitud de materiales audiovisuales</i> .....	8
1.1.6. <i>Préstamo de materiales audiovisuales</i> .....	9
1.1.7. <i>Servicio Web</i> .....	9
1.2.    Análisis de soluciones existentes.....	9
1.3.    Propuesta de solución.....	10
1.4.    ¿Por qué una aplicación móvil y no una web móvil?.....	11
1.5.    Tecnologías y herramientas a utilizar en el desarrollo del subsistema.....	12
1.5.1. <i>Proceso Unificado de Desarrollo</i> .....	12
1.5.2. <i>Lenguaje Unificado de Modelado (UML)</i> .....	13
1.5.3. <i>Herramienta CASE Visual Paradigm for UML 8.0</i> .....	14
1.5.4. <i>Lenguajes de Programación</i> .....	14
1.5.5. <i>Entorno de desarrollo integrado</i> .....	15
1.5.6. <i>Android Software Development Kit (SDK) 4.4</i> .....	16
1.5.7. <i>Android Development Tools (ADT) 23.0.2</i> .....	16
Conclusiones del Capítulo 1.....	17
Capítulo 2: Análisis y diseño del Subsistema de Solicitud y Préstamo del sistema Agorav para Android...	18
2.1.    Diseño del modelo de dominio.....	18
2.1.1. <i>Descripción del dominio del problema</i> .....	19
2.1.2. <i>Descripción de los conceptos del dominio</i> .....	19
2.2.    Especificación de los requisitos para el Subsistema de Solicitud y Préstamo del sistema Agorav para Android.....	20
2.2.1. <i>Requisitos Funcionales del Subsistema de Solicitud y Préstamo del sistema Agorav para Android</i> .....	20

2.2.2. <i>Requisitos no funcionales del Subsistema de Solicitud y Préstamo del sistema Agorav para Android</i> .....	23
2.3.    Modelo del sistema.....	23
2.3.1. <i>Descripción de los actores del sistema</i> .....	24
2.4.    Especificación de CU.....	25
2.5.    Patrones de diseño de software utilizados para el Subsistema de Solicitud y Préstamo del sistema Agorav para Android .....	30
2.5.1. <i>Patrones Generales de Software para Asignar Responsabilidades (GRASP)</i> .....	31
2.5.2. <i>Patrón arquitectónico Modelo Vista Controlador</i> .....	32
2.6.    Diagrama de clase del diseño .....	33
Conclusiones del Capítulo 2 .....	35
Capítulo 3: Implementación y pruebas del Subsistema de Solicitud y Préstamo del sistema Agorav para Android.....	36
3.1. Diagrama de componentes .....	36
3.2. Modelo de Despliegue .....	38
3.2.1. Descripción de componentes .....	39
3.3. Estándar y estilos de codificación .....	41
3.4. Proceso de pruebas.....	44
3.4.1. <i>Estrategia de pruebas</i> .....	44
3.4.2. <i>Tipo de prueba</i> .....	45
3.4.3. <i>Método de prueba</i> .....	45
3.4.4. <i>Técnica de prueba</i> .....	45
3.5.    Diseño de casos de pruebas .....	46
3.5.1. <i>Caso de prueba para el CU “Gestionar Solicitud”</i> .....	48
3.6.    Resultados de las Pruebas.....	51
Conclusiones del Capítulo 3 .....	52
Conclusiones Generales.....	53
Recomendaciones .....	54
Trabajos citados.....	<b>¡Error! Marcador no definido.</b>
Anexos .....	60
1.    Descripción de los CU .....	60

2.	Casos de prueba.....	68
2.1.	<i>Caso de prueba Buscar Material.....</i>	68
2.2.	<i>Caso de prueba Reproducir Material.....</i>	73

## **Índice de Tablas.**

Tabla 1. Descripción de los actores del sistema.....	24
Tabla 2. Descripción del CU Buscar Material. ....	25
Tabla 3. Secciones de prueba para el caso de uso Gestionar Solicitud. ....	46
Tabla 4. Descripción de las variables correspondientes al caso de prueba para el caso de uso Gestionar Solicitud.....	46
Tabla 5. Caso de prueba para el caso de uso Gestionar Solicitud: Sección 1: “Adicionar solicitud”. ....	48
Tabla 6. Caso de prueba para el caso de uso Gestionar Solicitud: Sección 1: “Modificar solicitud”. ....	49
Tabla 7. Caso de prueba para el caso de uso Gestionar Solicitud: Sección 1: “Eliminar solicitud”. ....	50
Tabla 8. Caso de prueba para el caso de uso Gestionar Solicitud: Sección 1: “Listar solicitudes”.....	51

## **Índice de Imágenes.**

Imagen 1. Diagrama modelo de dominio. ....	19
Imagen 2. Diagrama de CU del Sistema.....	24
Imagen 3. Modelo Vista Controlador. ....	33
Imagen 4. DCD del CU Buscar Material.....	34
Imagen 5. Diagrama de componentes del CU Buscar Material. ....	37
Imagen 6. Diagrama de Despliegue. ....	38
Imagen 7. Nodo Dispositivo Inteligente. ....	39
Imagen 8. Nodo Servidor Web. ....	39
Imagen 9. Nodo Servidor Streaming. ....	39
Imagen 10. Nodo Servidor de Almacenamiento. ....	40
Imagen 11. Nodo Servidor de Base de Datos.....	40
Imagen 12. Gráfico de resultado de las no conformidades. ....	52

## Introducción

Con el paso de los años el desarrollo tecnológico ha provocado grandes y significativos cambios en todos los sectores de la sociedad. Uno de los grandes avances tecnológicos que cada día juega un papel más importante para el hombre es Internet, tecnología decisiva de la era de la información en la que se está viviendo. Internet permite producir, distribuir y utilizar información digitalizada en cualquier formato. Según el estudio publicado por Martin Hilbert en Science en 2010, el 95% de toda la información existente en el planeta está digitalizada y en su mayor parte accesible en Internet y otras redes informáticas (Castells, 2013).

Hoy en día el hombre cuenta con diversos dispositivos electrónicos que tienen la capacidad de conectarse a Internet, ya no solo se conectan las computadoras u ordenadores portátiles, ahora y de manera creciente los dispositivos comunes (teléfonos, reproductores multimedia, entre otros) se convierten en otra forma de acceso a la red de redes. El hecho de que todos estos dispositivos se puedan conectar da origen al término "Internet de las cosas", que básicamente plantea que todas las cosas del mundo pueden estar conectadas a Internet.

Con esta nueva tendencia, aparecen los llamados Dispositivos Inteligentes (DI), que son aquellos dispositivos informáticos, ya sean ordenadores, portátiles, agendas personales, teléfonos inteligentes, consolas de juego y otros, que presentan un sistema que le permite relacionarse e intercambiar información con otros dispositivos (Tugurium, 2006).

Entre los DI más utilizados se encuentran los teléfonos móviles (*smartphones*) y las tabletas electrónicas (*tablet*), los cuales han adquirido otras funcionalidades que van más allá de realizar una llamada, traducir un texto o enviar mensajes. Estos dispositivos han incorporado las funcionalidades de otros, tales como cámara de fotos, agendas electrónicas, reloj despertador, calculadoras, GPS, reproductor de multimedia, entre muchos otros. Las funcionalidades que han permitido que estos dispositivos sean los que gocen de mayor popularidad son precisamente las opciones de conectarse a Internet fácilmente y visualizar contenido multimedia y audiovisual desde la red o del propio almacenamiento del dispositivo. En estos DI la conexión a la red es tan fácil como hacerlo desde un ordenador, cosa que les otorga nuevas ventajas y una mayor aceptación por los usuarios, los cuales los seleccionan como la vía ideal de comunicación para relacionarse e informarse.

Los DI cuentan al igual que los ordenadores con un sistema operativo (SO) que permite la interacción entre el hombre y la máquina. De acuerdo con un estudio llevado a cabo por Gabriel Osmar Pedrozo en la Universidad Nacional del Nordeste (Argentina) publicado en el 2012, el 75% de los usuarios de DI usan como SO Android, quedándose por debajo otros como iOS, BlackBerry, Symbian y Windows Phone (Petrazzini, 2012).

Actualmente el mundo se encuentra en un momento en el que ha aumentado el nivel de utilidad y funcionalidad de la tecnología móvil y con esta la de los DI que también tengan la característica de ser móviles. Cuba no se encuentra exenta de estos avances tecnológicos, pues en la población se hacen cada vez más visibles estos dispositivos, entre los que se destacan las tabletas electrónicas y los teléfonos inteligentes.

En la Universidad de las Ciencias Informáticas (UCI), existen varios centros dedicados al desarrollo y la producción de software para clientes nacionales e internacionales. En el Centro de Geoinformática y Señales Digitales (GEYSED) perteneciente a la Facultad 6 se encuentra el proyecto productivo Agorav, el cual es el encargado del desarrollo del sistema Agorav. Este sistema se desarrolla sobre tecnologías web, con el objetivo de lograr una mayor distribución y facilitar el acceso al mismo desde cualquier estación de trabajo vinculada a la red de la institución donde sea desplegado. Entre los subsistemas comprendidos por el sistema se encuentra el de recuperación, solicitud y préstamo de los materiales previamente catalogados, el cual agiliza estas actividades en instituciones donde sea desplegado el sistema. Entre los clientes potenciales se encuentran instituciones tales como televisoras, videotecas, productoras de cine, entre otras.

La solicitud y préstamo de Agorav es una forma de recuperar, visualizar o descargar videos de interés. El subsistema cuenta con dos tipos de búsquedas: "Avanzada y Básica". Cada una de estas presenta varios criterios para realizar la búsqueda con eficiencia y rapidez. Por ejemplo si se busca un video por el nombre del autor, saldrían todos los videos asociados a este nombre. Luego de encontrar el video deseado, se pueden realizar acciones tales como: visualizarlo en un reproductor del mismo sistema, escoger una parte del video que sea de interés o solicitarlo de inmediato. Para la realización de la solicitud, se llena un formulario con los datos del usuario y la media seleccionada, esta se ubica en una lista de solicitudes. Existe una persona en el sistema con el rol de Atención al público que es el encargado de revisar las solicitudes realizadas por los usuarios y aceptarlas o no, después de aceptada la misma, el

usuario puede descargarla en ese momento o conectarse luego para realizar la descarga, ya que esta se guarda en la lista de solicitudes.

La recuperación, solicitud y préstamo del sistema Agorav, como cualquier sistema web, puede ser accedida desde muchos DI siempre que cuenten con una interfaz de red. Su implementación no ofrece los mejores resultados para su uso en estos dispositivos, principalmente desde el punto de vista visual, ya que este no se redimensiona al tamaño de la pantalla del dispositivo desde el que fue accedido, el reproductor implementado en el subsistema no se carga en estos dispositivos por lo que no permite la visualización de los materiales audiovisuales presentes en el subsistema. Además presenta incompatibilidad en los navegadores, haciendo al subsistema inoperable en la mayoría de los casos.

A partir de la situación planteada surge como **problema de la investigación**: ¿Cómo lograr una correcta visualización del Subsistema de Solicitud y Préstamo del sistema Agorav desde dispositivos inteligentes con sistema operativo Android?

El **objeto de estudio** procesos de visualización en dispositivos inteligentes, estableciendo como **campo de acción** procesos de visualización del Subsistema de Solicitud y Préstamo del sistema Agorav en los dispositivos inteligentes con sistema operativo Android.

Para darle solución al problema planteado se define como **objetivo general** desarrollar el Subsistema de Solicitud y Préstamo del sistema Agorav para dispositivos inteligentes con sistema operativo Android.

Para guiar la investigación se definen las siguientes **preguntas científicas**:

- ¿Cuáles son los fundamentos teóricos de los procesos de visualización en dispositivos inteligentes?
- ¿Cuáles son las características que debe cumplir el Subsistema de Solicitud y Préstamo presente en el sistema Agorav para erradicar los problemas de visualización desde dispositivos inteligentes con sistema operativo Android?
- ¿Cómo llevar a cabo el proceso de desarrollo del Subsistema de Solicitud y Préstamo del sistema Agorav que permita la visualización desde dispositivos inteligentes con sistema operativo Android?

- ¿Permite el Subsistema de Solicitud y Préstamo del sistema Agorav, la visualización desde dispositivos inteligentes con sistema operativo Android?

Para el cumplimiento del objetivo trazado se definieron las **tareas de la investigación** que se proponen a continuación:

- Caracterización del sistema operativo Android y su estructura para el desarrollo de nuevas aplicaciones para el mismo.
- Caracterización de los procesos de visualización del Subsistema de Solicitud y Préstamo del sistema Agorav en dispositivos inteligentes con sistema operativo Android.
- Investigación sobre las tecnologías utilizadas en el desarrollo de aplicaciones para dispositivos inteligentes con sistema operativo Android.
- Identificación de los requisitos para la implementación el Subsistema de Solicitud y Préstamo en dispositivos inteligentes con sistema operativo Android.
- Diseño del Subsistema de Solicitud y Préstamo para dispositivos inteligentes con sistema operativo Android y los artefactos necesarios para su posterior implementación.
- Implementación del Subsistema de Solicitud y Préstamo para dispositivos inteligentes con sistema operativo Android.
- Realización de las pruebas para comprobar el correcto funcionamiento del Subsistema de Solicitud y Préstamo del sistema Agorav en dispositivos inteligentes con sistema operativo Android.

Con la misión de obtener conocimientos necesarios que hagan posible la materialización del objetivo general, se han utilizado diferentes tipos de métodos de investigación Teóricos y Empíricos, los cuales se describen a continuación:

## **Métodos Teóricos**

**Analítico–sintético:** Utilizado al estudiar la documentación especializada en el proceso de visualización desde dispositivos inteligentes con SO Android, que posibilitó la extracción de los elementos más importantes, permitiendo arribar a conclusiones.

## **Estructura del trabajo de diploma**

El siguiente trabajo de diploma se ha estructurado en un resumen, introducción, tres capítulos, conclusiones generales, recomendaciones, referencias bibliográficas y anexos. En el primer capítulo titulado “Fundamentos teóricos del Subsistema de Solicitud y Préstamo del sistema Agorav para Android” se realiza la elaboración del marco teórico donde se exponen los conceptos asociados a la solución de la problemática y las diferentes soluciones existentes a nivel mundial. También se describen y caracterizan la metodología, herramientas y tecnologías a utilizar para el desarrollo del subsistema. En el segundo capítulo titulado “Análisis y diseño del Subsistema de Solicitud y Préstamo del sistema Agorav para Android” se describen detalladamente las características del sistema a desarrollar. Se especifica el Modelo de Dominio para describir los principales conceptos del negocio, los requisitos funcionales y no funcionales, así como el diagrama de casos de uso del sistema y la descripción de uno de ellos. Se modelan y detallan los diagramas que representan las funcionalidades del sistema, a partir de los patrones de diseño identificados. En el tercer capítulo titulado “Implementación y pruebas del Subsistema de Solicitud y Préstamo del sistema Agorav para Android” se muestra la distribución física de los distintos componentes lógicos desarrollados, a través del modelo de despliegue y la organización del sistema mediante el modelo de componentes. Se explican los estilos de programación y estándares de codificación empleados y por último se valida el sistema desarrollado aplicándole las pruebas necesarias para demostrar que la solución es correcta.

# **Capítulo 1: Fundamentos teóricos del Subsistema de Solicitud y Préstamo del sistema Agorav para Android**

En este capítulo se abordan conceptos asociados al dominio del problema, los cuales ayudan a un mejor entendimiento del mismo. También se abordan aspectos importantes de soluciones existentes en el mundo que se pudieran tener en cuenta para un mejor desarrollo de la solución. Y por último se describen y caracterizan la metodología, las herramientas y lenguajes de programación que se emplean en el desarrollo e implementación del sistema informático que se propone como solución a la problemática existente.

## **1.1. Conceptos asociados al dominio del problema**

A continuación se relacionan los principales conceptos o temáticas que están asociados al desarrollo de la investigación.

### **1.1.1. Android**

Android es un SO móvil de código abierto, el cual está basado en una versión modificada de Linux (Lee, 2011). Al ser este un sistema abierto, utilizando aplicaciones que son implementadas usando el lenguaje de programación Java, los desarrolladores tienen acceso a las principales funcionalidades de los dispositivos donde se instale este SO (Gonzalez, 2011). La libertad del código de Android en poco tiempo ha logrado que se implante en multitud de dispositivos electrónicos, convirtiendo a Android en un SO multifuncional, lo que provoca una mayor expansión y crecimiento en su fabricación tecnológica y en el mercado y (Orozco, 2011). Ante indicadores de aceptación a nivel mundial que sobrepasan el 75% en su uso en los DI, la presente investigación se centrará en este SO, que además es soporte de un gran número de aplicaciones que trabajan y utilizan videos y materiales audiovisuales.

### **1.1.2. Materiales Audiovisuales**

Se hace necesario explicar el significado de materiales audiovisuales ya que a estos términos se les atribuyen definiciones muy diversas. Autores como (Kofler, 1991) plantean que las imágenes en movimiento componen la forma tradicional de material audiovisual, en la que se incluyen necesariamente

también grabaciones sonoras. Para la presente investigación la definición a utilizar de material audiovisual es la planteada por (Edmondson, 2002) en la que se refiere a los medios audiovisuales como obras que comprenden imágenes y/o sonidos reproducibles integrados en un soporte, y que se caracterizan porque requieren habitualmente un dispositivo tecnológico para su grabación, transmisión, percepción y comprensión. También plantea que el contenido visual y/o sonoro de estos medios consta con una duración lineal y que tiene como objetivo la comunicación de ese contenido.

Los materiales audiovisuales poseen un formato contenedor aunque suelen ser llamados indistintamente formatos de vídeo. Los formatos contenedores son llamados así, ya que unifican y guardan los formatos de audio, vídeo, subtítulos y otros posibles elementos, haciendo un objeto más compacto y completo (Edukanda, 2010). Actualmente existe una amplia variedad de estos formatos, disponibles para almacenar contenido multimedia en un archivo, entre los que se encuentran AVI, MPEG, MOV, entre otros. Las diferencias entre los formatos contenedores están dadas en que cada uno les garantiza a los materiales audiovisuales una calidad y un tamaño de archivo diferente, propiedades que influyen también en la calidad de visualización de estos materiales para las personas.

### **1.1.3. Visualización**

En una tesis doctoral presente en el sitio oficial de la Universidad Interamericana de Puerto Rico se exponen varios conceptos sobre visualización, los cuales se muestran a continuación. El término visualización se utiliza en referencia a figuras o representaciones pictóricas ya sean internas o externas. En ese sentido Presmeg (1986) se refirió a las imágenes visuales en el estudio “Visualización y matemáticas” (1986), ella suscribió la definición de imágenes visuales como un esquema mental que representa información visual o espacial. Por otra parte Zimmermann y Cunningham (1991) plantean que: Visualizar un diagrama significa simplemente formar una figura mental del diagrama, pero visualizar un problema es comprender el problema en términos de un diagrama o de una figura visual. Entre los roles que tiene la visualización se puede mencionar el razonamiento sobre las figuras. Ben-Chaim (1991) señaló que: la visualización cobra importancia en el proceso de razonamiento inductivo y deductivo (Márquez, 2002).

Refiriéndose a la visualización de información (Corona, 2013) plantea que la misma es: *“El uso de representaciones visuales e interactivas de datos abstractos, soportadas por un computador para amplificar el conocimiento”*.

Después de analizados los diferentes conceptos relacionados con la visualización y sus diferentes variantes es preciso definir lo que significará este término para la investigación. En el contexto en que se desarrolla el presente trabajo, muy relacionado con el desarrollo tecnológico y la era digital se define entonces como visualización a la acción de visualizar u observar las imágenes estáticas o en movimiento que reproducen o representan los dispositivos electrónicos inteligentes. Teniendo el vidente la opción de interpretar y asimilar el contenido visualizado, e incluso interactuar muchas veces con él.

#### **1.1.4. Recuperación de materiales audiovisuales**

Existen varios factores comprendidos en el proceso de búsqueda y recuperación de información, el cual es un proceso de resolución de problemas. Uno de estos factores es un usuario que busca información para solucionar una dificultad o necesidad informativa. Otro es el sistema de búsqueda, donde se encierran las técnicas para la recuperación de la información. Además la necesidad de información del usuario se debe satisfacer mediante el conjunto de resultados que se alcanzan del sistema. Estos elementos influyen en el resultado final de recuperación o búsqueda e interactúan entre ellos, logrando después que se pueda visualizar y solicitar dicha información (Pérez, 2014).

La recuperación de los materiales audiovisuales implica el proceso inicialmente descrito, todo se basa en la necesidad del usuario de encontrar el material que desea. Este proceso, en el sistema Agorav se evidencia a la hora de realizar una búsqueda, en caso de que el material no esté publicado, ni disponible visualmente pero se encuentre almacenado en la base de datos de este, se mostrará en el resultado de la búsqueda permitiendo recuperar dicho material, posibilitando la visualización y solicitud del mismo.

#### **1.1.5. Solicitud de materiales audiovisuales**

Cuando alguien se refiere al término “solicitud” se refiere generalmente a un documento escrito que va destinado a un organismo público o a una autoridad, a los que se pide algo o ante los que se plantea una reclamación con la manifestación de los motivos en los que se apoyan (Toledo, 2010). La solicitud de materiales audiovisuales es cuando un usuario luego de realizar una búsqueda entre las medias

existentes en el sistema Agorav encuentra la deseada y en ese momento realiza la solicitud de la misma, donde quedan recogidos los datos del usuario en cuestión (nombre, apellidos, carnet de identidad e institución) y la media seleccionada (título, tiempo de inicio, tiempo de fin y formato en que se desea solicitar).

### **1.1.6. Préstamo de materiales audiovisuales**

Se define préstamo en el sistema Agorav, cuando un usuario realiza la descarga de una media luego de realizada la solicitud por este, ser aceptada la misma y encontrarse en el estado de lista para descargarse.

### **1.1.7. Servicio Web**

Es una tecnología que usa un grupo de protocolos y estándares que utilizan para intercambiar datos entre aplicaciones. Diferentes aplicaciones de software implementadas en distintos lenguajes de programación, ejecutadas sobre cualquier plataforma pueden utilizar los servicios web con el objetivo de intercambiar datos en redes de ordenadores como Internet. A través de la adopción de estándares abiertos se logra la interoperabilidad. Los servicios web presentan características entre las que se encuentran que permiten ser consumidos desde cualquier tipo de aplicación sin tener en cuenta el lenguaje, facilitan el acceso a su contenido y funcionamiento, estos están orientados a la web y puede ser mezclados para proveer servicios integrados (Aperador, 2012). El sistema Agorav brinda varios servicios *Simple Object Access Protocol* (SOAP) que no son más que un estándar del *World Wide Web Consortium* (W3C) que especifica cómo objetos remotos pueden establecer comunicación mediante el intercambio de XML. Se basa en la comunicación de dos partes (cliente y servidor), una de las cuales (el servidor) proporciona una serie de servicios que son consumidos por la otra (cliente) (Calahorro, 2009). Estos servicios SOAP pueden ser consumidos desde otros sistemas o aplicaciones.

## **1.2. Análisis de soluciones existentes**

En el mundo existen varias soluciones informáticas que garantizan la visualización, la solicitud y el préstamo (descarga) de materiales audiovisuales por parte de los clientes. A continuación se describen las características de algunas de ellas.

**YouTube para Android:** esta aplicación consta con una interfaz bien diseñada, la cual ofrece todas las funciones de YouTube evitando desperdiciar el espacio en la pantalla. Posibilita visualizar vídeos en

vertical y leer la información y videos relacionados al mismo tiempo. En cuanto es girado el dispositivo desde donde se esté visualizando en modo apaisado, YouTube se redimensiona automáticamente y muestra el vídeo en pantalla completa, eliminando el resto de los elementos. Posee botones de reproducción sencillos y para pausar solo hay que tocar la pantalla o deslizar el dedo para ir adelante o atrás en la reproducción (Beech, 2015). Solo permite la descarga de los materiales audiovisuales de manera completa, sin tener opciones para seleccionar solo una parte del audiovisual. No hay forma de hacer una solicitud de un material que no esté visible o no, pues si está visible le permite la descarga a cualquier usuario y si no lo está, no hay manera de comunicarles a los administradores que lo publiquen nuevamente.

**Vimeo para Android:** es la aplicación oficial del portal “Vimeo” de videos online. Con esta aplicación se pueden realizar acciones tales como: visualizar los videos y descargarlos. La aplicación está diseñada con una interfaz clara y organizada para encontrar fácilmente todas las opciones que ofrece, entre las que se encuentran búsquedas rápidas y visualización de materiales audiovisuales haciendo uso de la tecnología *streaming* (Cruz, 2012). Al igual que la aplicación de YouTube para Android no tiene opciones para la selección de una parte de un material audiovisual, ni para realizar solicitudes.

Las soluciones antes descritas solo permiten la visualización de las páginas oficiales de determinados sitios, en estos casos YouTube y Vimeo, por lo que no pueden ser adaptadas para que visualicen otros contenidos web. Tampoco presentan funcionalidades necesarias como la selección de fragmentos de un audiovisual, o la opción de realizar una solicitud de estos. Al no constituir una solución para el problema planteado en la investigación se propone realizar el Subsistema de Solicitud y Préstamo del sistema Agorav para Android. Cabe destacar que las aplicaciones estudiadas presentan particularidades que sirven de modelo para la realización del subsistema a implementar, tales como una interfaz intuitiva y comprensible, además de brindar la posibilidad de visualizar la media y descargarla. También aportaron varias ideas de cómo realizar algunas funcionalidades del subsistema en desarrollo.

### **1.3. Propuesta de solución**

El sistema Agorav es una solución web que permite la gestión de contenido audiovisual con el objetivo de transmitirlo posteriormente utilizando tecnología *streaming*. En este sistema se maneja la información asociada a contenidos audiovisuales, permitiendo acrecentar sus publicaciones y resultados. El sistema

puede ser empleado con fines docentes, informativos o de entretenimiento. Entre los subsistemas comprendidos por el sistema se encuentra el de recuperación, solicitud y préstamo de los materiales previamente catalogados.

Como cualquier sistema web, a la recuperación, solicitud y préstamo del sistema Agorav, se puede acceder desde varios DI, siempre que exista alguna red a la cual puedan conectarse. Estos pueden acceder pero se les hace difícil la navegación ya que su implementación no ofrece los mejores resultados para su uso, principalmente desde el punto de vista visual, ya que este no se redimensiona al tamaño de la pantalla del dispositivo desde el que fue accedido, además de existir incompatibilidad entre los navegadores de los ordenadores y de dichos dispositivos, haciendo al sistema inoperable en la mayoría de los casos.

Como propuesta de solución a la problemática del presente trabajo se define la realización del Subsistema de Solicitud y Préstamo del sistema Agorav para Android. Con el desarrollo de dicho subsistema se le permitirá a cualquier usuario con un DI que presente Android como SO consumir un servicio que le permitirá realizar una búsqueda básica de archivos multimedia y otro servicio para la búsqueda avanzada. Como resultado se obtendrá uno o varios videos según los filtros introducidos en dichas búsquedas, los cuales podrán ser visualizados en un reproductor. Este presenta varias funcionalidades como son: reproducir, pausar o detener el video en reproducción, además de permitir crear marcas en este e insertar los subclips que deseen solicitar. Al presionar el botón de aceptar la aplicación mostrará una lista con los subclips insertados además de mostrar el video completo permitiendo realizar la solicitud de cada uno de estos. Al presionar el botón solicitar se mostrará al usuario un formulario donde se ingresarán los datos de este y del subclip o la media a solicitar, donde al presionar el botón de aceptar, la aplicación consumirá el servicio de solicitud y la creará en la base de datos del sistema Agorav. Una vez aceptada la solicitud realizada, se le permite al usuario descargar al dispositivo, la media o subclip solicitado.

#### **1.4. *¿Por qué una aplicación móvil y no una web móvil?***

Un sitio web móvil se diseña específicamente para las pantallas más pequeñas y las capacidades de la pantalla táctil de los teléfonos inteligentes y tabletas. El acceso al mismo es mediante el navegador Web de cualquiera de estos dispositivos. Los usuarios simplemente deben escribir la URL del sitio web deseado y este detecta automáticamente el DI y redirige al usuario a la versión móvil del sitio web. Entre las ventajas de un sitio web para móviles se encuentra que hace que los sitios web regulares sean más

accesibles para los usuarios móviles. Posee los mismos elementos que la versión regular de la página web, como su apariencia, las páginas, imágenes entre otros contenidos. Este cuenta con un diseño para usar en móviles lo cual ofrece una mejor legibilidad y funcionalidad cuando se ve en un *smartphone* o *tablet*. Al existir un sitio web para móviles, los usuarios pueden acceder a este en cualquier momento y en cualquier lugar utilizando cualquier dispositivo (Angeles, 2014).

Una aplicación móvil es una aplicación para DI que a diferencia de un sitio web para móviles, esta debe ser descargada e instalada. Una aplicación móvil funciona muy parecido a un sitio web para móviles. Es mucho más fácil acceder a una aplicación móvil que a un sitio web para móviles, todo lo que se necesita es la aplicación, en lugar de tener que abrir un navegador Web y escribir en él una URL. La aplicación puede extraer contenido y datos a través de Internet, de manera similar a un sitio web, o puede descargar el contenido de modo que se puede acceder sin conexión a Internet. Una aplicación móvil ofrece a los usuarios la funcionalidad y velocidad que no se puede lograr con un sitio web (Angeles, 2014).

Se eligió la realización de una aplicación móvil ya que esta utiliza menos recursos al consumir los servicios brindados por el sistema Agorav, que la implementación de un sitio web móvil. Además la aplicación móvil al estar instalada en un dispositivo permite guardar datos en cache y trabajar en segundo plano con estos datos sin conexión, mientras que un sitio web móvil debe contar con conexión a Internet todo el tiempo. El reproductor que está implementado actualmente en el sistema Agorav no se carga en los DI, lo que no permite la visualización de los materiales audiovisuales presentes en el sistema, por lo que se implementará una aplicación que contenga un reproductor que permita visualizar las medias contenidas en el sistema Agorav.

## **1.5. Tecnologías y herramientas a utilizar en el desarrollo del subsistema**

En este epígrafe se exponen las herramientas, metodologías y lenguajes de programación utilizados para el desarrollo del subsistema, además de la arquitectura que se utilizará para la creación de los servicios que se brindarán.

### **1.5.1. Proceso Unificado de Desarrollo**

Para guiar el proceso de desarrollo de software de la presente investigación se utilizó como metodología el Proceso Unificado de Desarrollo (RUP por sus siglas en inglés) ya que es muy utilizada para la

documentación y realización de software orientados a objetos. Propone varias iteraciones en el proceso de desarrollo del sistema lo que posibilita la corrección de errores y mejoras del software a medida que avanza el proyecto, permitiendo a los desarrolladores poder realizar un producto con mayor calidad. Con esta metodología el sistema se va desarrollando y documentando al mismo tiempo. RUP ayuda a construir un software de alta calidad, desarrollado en el tiempo planificado, permite que se trabaje de forma organizada donde se controle y documente todo lo relacionado con el proyecto.

RUP es una forma disciplinada de asignar trabajos y responsabilidades en una empresa de desarrollo (quién hace qué, cuándo y cómo). Es un marco del proyecto que describe una clase de los procesos que son iterativos e incrementales. De los procesos existentes actualmente este es el proceso de desarrollo más general. Los procesos de RUP estiman tareas y horario del plan midiendo la rapidez de iteraciones referente a sus estimaciones originales. Las iteraciones prematuras de proyectos llevados a cabo por RUP se orientan fuertemente sobre arquitectura del software; la puesta en práctica rápida de particularidades se atrasa hasta que se ha identificado y se ha analizado una arquitectura firme (Flores, 2012). Se escoge esta metodología pues es la utilizada en el desarrollo del sistema Agorav. Al mantener RUP como metodología se garantiza que los documentos ingenieriles (artefactos) desarrollados estén en correspondencia con los confeccionados anteriormente para el mismo proyecto.

### **1.5.2. Lenguaje Unificado de Modelado (UML)**

Según (Larman, 1999) *“UML se define como un lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de software. Es un sistema notacional (que, entre otras cosas, incluye el significado de sus notaciones) destinado a los sistemas de modelado que utilizan conceptos orientados a objetos.”*

Se selecciona UML pues es un lenguaje gráfico para representar, especificar y documentar un sistema que consta con un conjunto de notaciones y diagramas estándares para modelar sistemas orientados a objetos, detallando la semántica esencial de lo que estos significan (Schmuller, 2001). RUP propone usar UML para llevar la documentación del sistema, facilitar la etapa del diseño y posterior construcción o desarrollo, transmitir ideas, y ayudar al equipo a comunicarl

### **1.5.3. Herramienta CASE Visual Paradigm for UML 8.0**

CASE es un acrónimo para *Computer Aided Software Engineering*, que en español significa Ingeniería de Software Asistida por Computadora. Se considera una herramienta CASE a un grupo de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de software, en su totalidad o en alguna de sus fases (UCOL, 2006). Estas herramientas facilitan la construcción de prototipos, el desarrollo conjunto de aplicaciones, perfeccionan y estandarizan la documentación. Además aumentan la portabilidad de las aplicaciones, facilitan la reutilización de componentes y viabilizan un desarrollo y refinamiento visual de las aplicaciones, mediante el manejo de gráficos. Al utilizar una herramienta CASE en el desarrollo de un proyecto se garantiza un aumento en la calidad y por consiguiente, un aumento de la productividad (INEI, 1999). Para el desarrollo de la solución se utilizó *Visual Paradigm for UML 8.0*.

Visual Paradigm es una herramienta multiplataforma, característica que la favorece y que viene muy acorde a la migración al software libre que lleva a cabo Cuba. Es una herramienta UML profesional que soporta el ciclo de vida del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Se integra con varios IDEs (Entorno de Desarrollo Integrado) y soporta múltiples usuarios trabajando sobre un mismo proyecto. Ofrece interoperabilidad entre diagramas, ya que permite a partir de un diagrama obtener otro que guarde relación con el mismo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación (Visual Paradigm, 2014).

### **1.5.4. Lenguajes de Programación**

Un lenguaje de programación no es más que un lenguaje artificial que se puede usar para controlar el comportamiento de una máquina, principalmente una computadora. Estos se componen de un grupo de reglas sintácticas y semánticas que permiten enunciar instrucciones que luego serán interpretadas (Alegsa, 2009). Para el desarrollo de la aplicación propuesta se hacen uso de los siguientes lenguajes:

**Java:** es el lenguaje de programación oficial y el soportado de forma nativa por Android, tiene gran compatibilidad con dicho SO por lo que se utilizará para el desarrollo del subsistema propuesto. Java es un lenguaje de programación con el que se puede realizar cualquier tipo de programa. Actualmente es un lenguaje muy extendido, por lo que cobra cada vez más importancia, tanto en el ámbito de Internet como

en la informática en general. Está desarrollado por la compañía Sun Microsystems. Este se sustenta de 5 pilares los cuales son: la programación orientada a objetos, la posibilidad de ejecutar un mismo programa en diversos sistemas operativos, la inclusión por defecto de soporte para trabajo en red, la opción de ejecutar el código en sistemas remotos de manera segura y la facilidad de uso (Álvarez, 2001).

**XML:** Lenguaje de Marcado Extensible (XML) es una especificación de propósito general para la creación de lenguajes de marcado personalizado. Se clasifica como un lenguaje extensible porque permite a los desarrolladores definir sus propios elementos. Tiene como principal objetivo ayudar a los sistemas de información a compartir datos estructurados, sobre todo a través de Internet, es utilizado tanto para codificar documentos como para serializar datos (Lana, 2013). Se utiliza para el desarrollo de las vistas que se le mostrarán al usuario al interactuar con el subsistema.

### **1.5.5. Entorno de desarrollo integrado**

Un entorno de desarrollo integrado (IDE por sus siglas en inglés) es un programa informático compuesto por un conjunto de herramientas de programación entre las que se encuentran: un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Puede dedicarse a uno o varios lenguajes de programación, además de proveer un entorno de trabajo amigable para la mayoría de estos (ALEGSA, 2010).

Para el desarrollo de aplicaciones para el SO Android los entornos más utilizados son Android Studio y Eclipse, a continuación se detallan sus características principales y se realiza la selección de uno de ellos para el desarrollo de la solución propuesta.

**Android Studio:** es un IDE de desarrollo multiplataforma creado por Google e implementado en Java. Es una especie de escritorio de trabajo para un desarrollador. Este cuenta con algunas herramientas que facilitan el desarrollo de las aplicaciones, como por ejemplo el poder previsualizar las aplicaciones en diferentes teléfonos inteligentes y tabletas electrónicas para saber cómo está quedando según como se está implementando y cómo se ve en los diferentes tipos de pantalla que existen. Presenta una interfaz específica para el desarrollo en Android, permite la importación de proyectos realizados en el entorno Eclipse y posee un editor de diseño que muestra una vista previa de los cambios realizados directamente en el archivo xml. Luego de implementada una aplicación, es compilada para crear un archivo *.apk* el cual es el que se instala en los dispositivos (Jiménez, 2013 ).

**Eclipse:** es un IDE de código abierto y multiplataforma. Cuenta con herramientas para desarrollar aplicaciones de consola, Web y Web Services. Proporciona soporte a todo tipo de proyectos abarcando el ciclo de vida completo en el desarrollo de aplicaciones. Posee un desarrollo continuo y una amplia comunidad de desarrolladores. Permite la instalación de *plugins* destinados a mejorar las funcionalidades del propio IDE y a extenderse en más tecnologías. Es capaz de automatizar, documentar, generar código y una profunda prueba de errores a la hora de compilación y tratamiento de ficheros (Martínez, 2010).

Para el desarrollo del Subsistema de Solicitud y Préstamo para Android se utilizará Eclipse ADT Bundle en su versión 23.0.2. Este entorno posee las herramientas del SDK de Android y una versión del IDE de Eclipse con una función de ADT (Herramientas para el desarrollo de Android, ADT por sus siglas en inglés) (Ugarte, 2014). Para su selección también se tubo en cuenta que el equipo de desarrollo está más familiarizado con este IDE, lo que ayudará a aprovechar mejor el tiempo ya que no se tendrá que estudiar el IDE Android Studio para comprender su funcionamiento, por otra parte las versiones del Eclipse ADT Bundle son estables y propician un buen funcionamiento, además de poseer un mayor número de *plugins* para el desarrollo de aplicaciones.

#### **1.5.6. Android Software Development Kit (SDK) 4.4**

Android SDK es el paquete de desarrollo imprescindible para programar e implementar todo tipo de aplicaciones para Android, el SO para DI de Google. Este paquete de desarrollo incluye las aplicaciones y herramientas vitales para desarrollar las aplicaciones utilizando Java como lenguaje de programación y probar el código, respectivamente. Las librerías adjuntas en Android SDK son compatibles con los siguientes entornos de desarrollo: Eclipse, JDK5 o JDK6, *Android Development Tool Plugin* y *Apache Ant* (Developer, 2014).

#### **1.5.7. Android Development Tools (ADT) 23.0.2**

*Android Development Tools (ADT)* es un *plugin* para el IDE Eclipse. ADT aumenta las capacidades de Eclipse para que pueda configurar rápidamente nuevos proyectos para Android, crear una interfaz de usuario de la aplicación, agregar paquetes basados en la API de Android Framework, depurar sus aplicaciones utilizando las herramientas del SDK de Android, e incluso exportar firmados (o no firmados) archivos *.apk* con el fin de distribuir las aplicaciones (Cloninger, 2012).

## Conclusiones del Capítulo 1

- El estudio de los principales conceptos asociados a la visualización en DI permitió sentar las bases para el desarrollo de la propuesta de solución, además del por qué y la necesidad de la realización del presente trabajo.
- El estudio de las soluciones existentes para la visualización de sistemas web que trabajan con audiovisuales permitió comprobar que las mismas no constituyen solución para la problemática de la investigación, de ahí el hecho de desarrollar el Subsistema de Solicitud y Préstamo para Android.
- La utilización del IDE de desarrollo Eclipse, los lenguajes de programación Java y XML, la metodología RUP, además de Visual Paradigm como herramienta CASE facilitan y fortalecen el trabajo, aportan rapidez y eficacia a lo largo del ciclo de vida del desarrollo y garantizan una mayor usabilidad de la aplicación.

## **Capítulo 2: Análisis y diseño del Subsistema de Solicitud y Préstamo del sistema Agorav para Android**

En el presente capítulo se exponen las principales características y cualidades del sistema a implementar, mediante la identificación de los requisitos funcionales y no funcionales con los que debe de cumplir la aplicación que dará solución a la problemática de la presente investigación. También se describen los conceptos relacionados con las definiciones de patrones de diseño de software empleados. Además se detallan los pasos de la metodología que se propone en el capítulo anterior (RUP). Se incluye la elaboración del modelo de dominio, la definición de actores y relaciones entre ellos, además de los diagramas de casos de uso del sistema y las descripciones textuales de los casos de uso. Y por último se muestran los diagramas de clases del diseño propuestos.

### **2.1. Diseño del modelo de dominio**

El modelo del dominio es una representación de las clases conceptuales del mundo real que se utiliza como guía para el diseño de los objetos. En él se muestran las clases conceptuales más significativas del dominio del problema. Este se realiza cuando no se logra determinar el proceso del negocio con fronteras bien establecidas. Este modelo representa clases conceptuales del dominio del problema que son las ideas u objetos físicos y el enlace de unos objetos con otros. Para la realización del mismo se utiliza un lenguaje común facilitando la comunicación entre los desarrolladores y usuarios de la aplicación y un mayor entendimiento del contexto en que se desarrolla el sistema. El modelo de dominio se describe mediante diagramas UML (diagrama de clases). Estos diagramas muestran las clases del dominio y cómo se relacionan unas con otras mediante asociaciones (Larman, 2003).

A continuación se presenta el Modelo de Dominio (ver Imagen.1) de la aplicación para la visualización de los contenidos audiovisuales brindados desde el sistema Agorav en DI y la descripción de las clases del dominio que intervienen en dicho modelo.

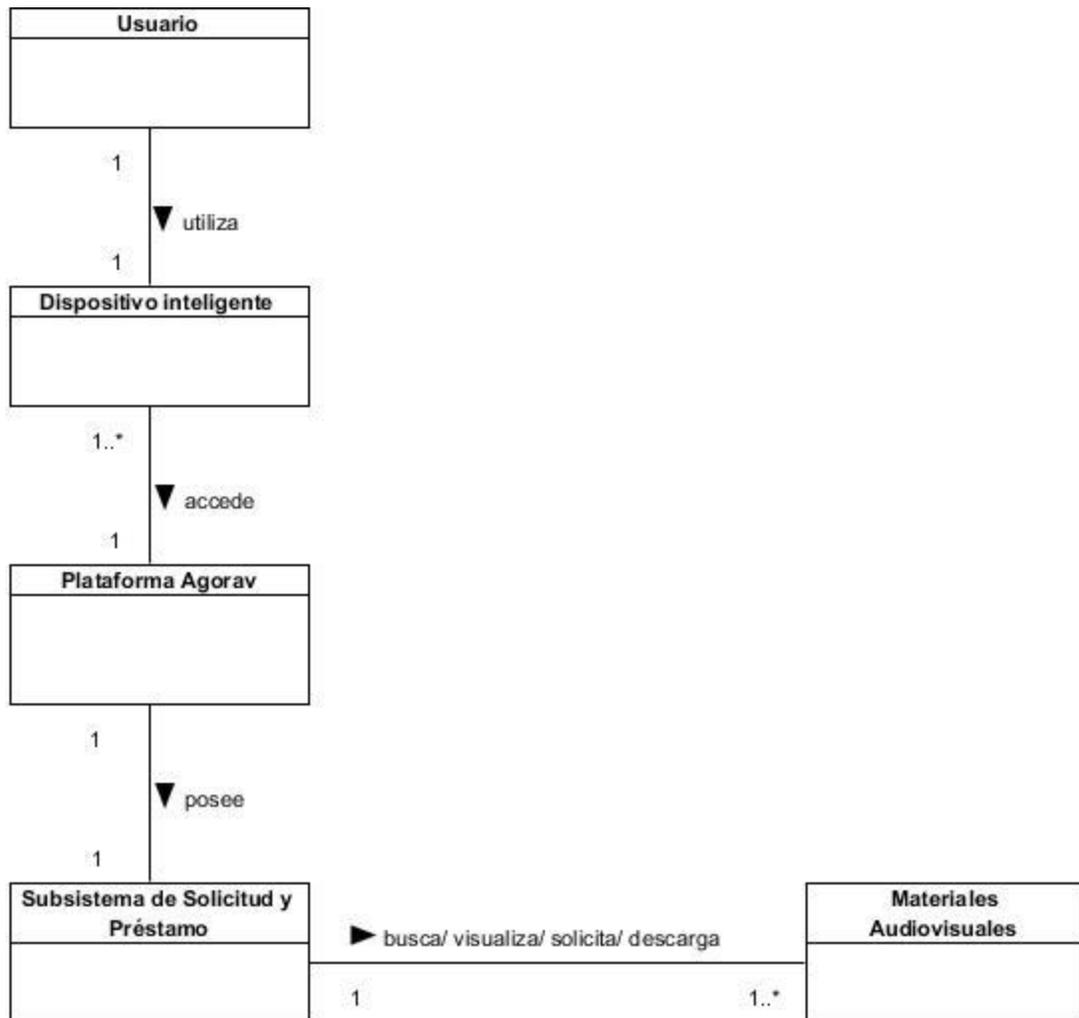


Imagen 1. Diagrama modelo de dominio.

### 2.1.1. Descripción del dominio del problema

El usuario utiliza un DI desde el cual accede al sistema Agorav. Dicho sistema posee el Subsistema de Solicitud y Préstamo, el cual permite la búsqueda, visualización, solicitud y descarga de materiales audiovisuales presentes en el sistema.

### 2.1.2. Descripción de los conceptos del dominio

**Usuario:** Persona que interactúa con el Subsistema de Solicitud y Préstamo del Sistema Agorav.

**Sistema Agorav:** Sitio web destinado a la gestión y transmisión de materiales audiovisuales a través de la red de datos.

**Subsistema de Solicitud y Préstamo:** Subsistema del sistema Agorav, desde el cual se pueden realizar búsqueda de materiales audiovisuales, visualizarlos, solicitarlos y descargarlos.

**Archivos Multimedia:** Contenido audiovisual presente en el sistema.

**Dispositivo inteligente:** Dispositivo electrónico desde el cual se accede al sistema Agorav.

## **2.2. Especificación de los requisitos para el Subsistema de Solicitud y Préstamo del sistema Agorav para Android**

Los requisitos son declaraciones que identifican atributos, capacidades, particularidades y/o cualidades que requiere cumplir un sistema para que posea valor y provecho para el usuario. En otras palabras, los requisitos expresan qué elementos y funciones son imprescindibles para un proyecto (Alegsa, 2009).

(Pressman, 2005) plantea que la tarea del análisis de requisitos es un proceso de descubrimiento, refinamiento, modelado y especificación. Se refina en detalle el ámbito del software, y se crean modelos de los requisitos de datos, flujo de información y control, y del comportamiento operativo. Se analizan soluciones alternativas y se asignan a diferentes elementos del software. El análisis de requisitos permite al desarrollador o desarrolladores especificar la función y el rendimiento del software, indica la interfaz del software con otros elementos del sistema y establece las restricciones que debe cumplir el software. Definiéndose de esta manera los límites del sistema.

### **2.2.1. Requisitos Funcionales del Subsistema de Solicitud y Préstamo del sistema Agorav para Android**

#### **RF1. Realizar búsqueda avanzada.**

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

#### **RF2. Realizar búsqueda básica.**

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

### **RF3. Limpiar filtros de búsqueda.**

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

### **RF4. Reproducir media.**

**Descripción:** El sistema permitirá reproducir el *streaming* de video en baja resolución asociado a los materiales almacenados en el servidor.

### **RF5. Pausar Reproducción.**

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

### **RF6. Parar Reproducción.**

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

### **RF7. Ir a un Instante de Tiempo.**

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

### **RF8. Controlar Volumen.**

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

### **RF9. Crear Marcas.**

**Descripción:** El sistema debe permitir establecer marcas de inicio y fin sobre la línea de reproducción del material en reproducción.

**RF10. Ir a la Marca Inicio.**

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

**RF11. Ir a la Marca Fin.**

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

**RF12. Reproducir Fragmento entre Marcas.**

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

**RF13. Insertar Solicitud**

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

**RF14. Modificar Solicitud**

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

**RF15. Eliminar Solicitud**

**Descripción:** El sistema debe permitir a los usuarios eliminar una solicitud haya realizado anteriormente.

**RF16. Listar Solicitudes**

**Descripción:** El sistema debe permitir a los usuarios la visualización de la lista de solicitudes realizadas por este.

**RF17. Responder Solicitud**

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

### **RF18. Descargar Material**

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

### **2.2.2. Requisitos no funcionales del Subsistema de Solicitud y Préstamo del sistema Agorav para Android**

#### **RnF1. Requisitos de software**

- Se debe disponer de una versión superior a la 2.2, del SO Android en el dispositivo.

#### **RnF2. Requisitos de hardware**

- El DI debe poseer alguna tecnología que le permita conectarse a la red.
- Se requiere un mínimo de 255 Mb de memoria RAM.

#### **RnF3. Usabilidad**

- La aplicación debe contar con una interfaz de fácil uso, que describa con claridad los procesos que realiza el Subsistema de Solicitud y Préstamo.
- La tipografía debe ser legible, con color en contraste con el fondo, para facilitar la visualización de la aplicación.

## **2.3. Modelo del sistema**

La aplicación constará de cuatro casos de usos (CU) los cuales son inicializados por cualquier usuario. El usuario y el CU sombreado no se aplican en el desarrollo de la aplicación, simplemente se muestran en el diagrama para representar que un usuario no puede realizar la descarga de una solicitud, sin antes haber sido atendida por un usuario con el rol de Atención al público.

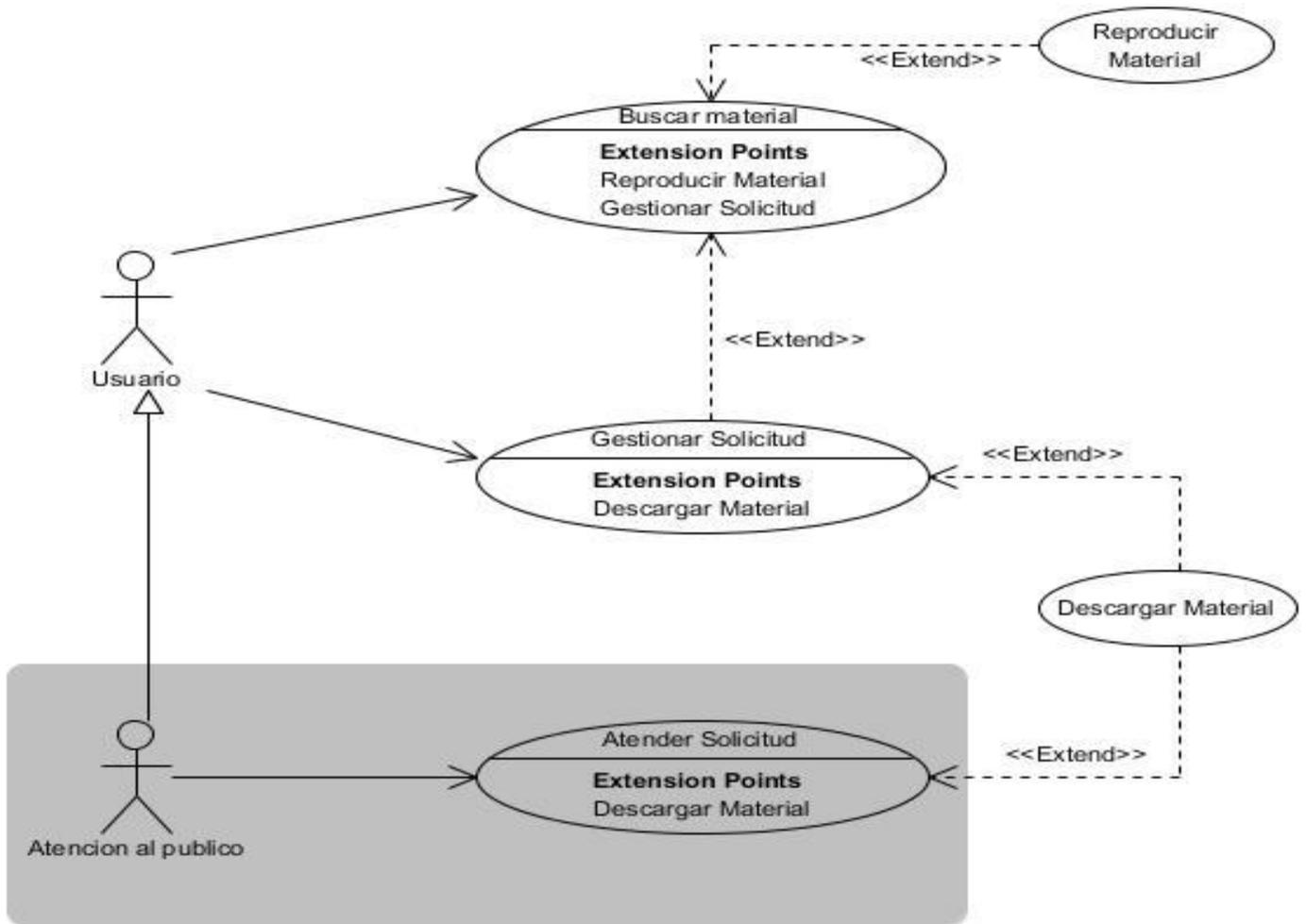


Imagen 2. Diagrama de CU del Sistema.

### 2.3.1. Descripción de los actores del sistema

Tabla 1. Descripción de los actores del sistema.

Actor	Descripción
Usuario	Persona que interactúa con la aplicación
Atención al Público	Persona que se encarga de atender las solicitudes hechas por los usuarios

## 2.4. Especificación de CU

Tabla 2. Descripción del CU Buscar Material.

<b>Objetivo</b>	Mostrar materiales audiovisuales según los criterios de búsqueda que utilice el usuario de acuerdo al tipo de búsqueda que desear realizar.	
<b>Actores</b>	Usuario:(Inicia) búsqueda de materiales.	
<b>Resumen</b>	El CU inicia cuando se selecciona una de las dos opciones de búsqueda de materiales que permite realizar el sistema (avanzada y básica), el resultado de la búsqueda se muestra teniendo en cuenta los criterios de búsqueda introducidos. El CU termina cuando se muestran los resultados obtenidos a partir de los criterios de búsqueda introducidos por el usuario.	
<b>Complejidad</b>	Media	
<b>Prioridad</b>	Media	
<b>Precondiciones</b>		
<b>Postcondiciones</b>	Se muestra el listado de los materiales de acuerdo a los criterios de búsqueda establecidos por el usuario.	
<b>Flujo de eventos</b>		
<b>Flujo básico: Buscar material</b>		
	<b>Actor</b>	<b>Sistema</b>
<b>1</b>	Activa la pestaña correspondiente para introducir opcionalmente los criterios de búsqueda de acuerdo al tipo de búsqueda que desea realizar, las que pueden ser:	

	<ul style="list-style-type: none"> <li>- Búsqueda Básica.</li> <li>- Búsqueda Avanzada. Ver sección 1: "Búsqueda avanzada".</li> </ul>	
2		Muestra el formulario correspondiente para introducir los valores de los atributos correspondientes a la clasificación de la búsqueda seleccionada.
"Búsqueda básica".		
<b>Flujo básico: Búsqueda básica</b>		
	<b>Actor</b>	<b>Sistema</b>
1		Activa el área de Búsqueda básica, donde podrán ser introducidos los criterios de búsqueda de la búsqueda básica.
2	<p>Introduce los filtros correspondientes a la búsqueda básica:</p> <ul style="list-style-type: none"> <li>- Si está Catalogado.</li> <li>- Si está Disponible.</li> <li>- Duración.</li> <li>- Nombre Físico.</li> <li>- Título</li> </ul> <p>Presiona botón "Buscar".</p>	

	Presionar botón "Limpiar". Ver sección 2: "Limpiar"	
3		Valida que los datos introducidos estén correctos.
4		Verifica que exista conexión con la base de datos.
5		Combina todos los filtros de búsqueda introducidos y muestra la lista de los materiales que cumplen con los mismos. Termina el CU.
<b>Flujos alternos</b>		
<b>3a. Evento:</b> Inserción de datos erróneos.		
	<b>Actor</b>	<b>Sistema</b>
3a		Muestra un mensaje de error indicando que el dato insertado no es válido. Regresa al paso 2 del flujo básico Búsqueda básica.
<b>Flujos alternos</b>		
<b>4a. Evento:</b> La conexión con el sistema Agorav fue fallida.		
	<b>Actor</b>	<b>Sistema</b>
4a		Si no se puede establecer con el sistema Agorav, el sistema muestra el siguiente mensaje: "No se puede establecer conexión con el sistema". Termina el CU.
<b>Flujos alternos</b>		
<b>5a. Evento:</b> No existen materiales que cumplan con los filtros introducidos.		

	Actor	Sistema
5a		Si no existen materiales que cumplan con los filtros introducidos por el usuario la lista de resultados no mostrará ningún elemento indicando que no existen coincidencias con los criterios de búsqueda introducidos. Termina el CU.
<b>Sección 1: "Búsqueda avanzada".</b>		
<b>Flujo básico: Búsqueda avanzada</b>		
	Actor	Sistema
1		Activa el área de Búsqueda avanzada permitiendo introducir los filtros de búsqueda avanzada.
2	<p>Introduce los datos correspondientes a los filtros de búsqueda avanzada:</p> <ul style="list-style-type: none"> <li>- Fecha Inicio</li> <li>- Fecha Fin</li> <li>- Texto</li> <li>- Ámbito</li> <li>- Presiona botón "Buscar"</li> <li>- Presiona botón "Limpiar". Ver sección 2: "Limpiar"</li> </ul>	
3		Valida que los datos introducidos estén correctos.

4		Verifica que exista conexión con la base de datos.
5		Combina todos los filtros de búsqueda introducidos y muestra la lista de los materiales que cumplen con los mismos. Termina el CU.
<b>Flujos alternos</b>		
<b>3a. Evento:</b> Inserción de datos erróneos.		
	<b>Actor</b>	<b>Sistema</b>
3a		Muestra un mensaje de error indicando que el dato insertado no es válido. Regresa al paso 2 del flujo básico Búsqueda avanzada (ver sección 1: "Búsqueda avanzada").
<b>Flujos alternos</b>		
<b>4a. Evento:</b> La conexión a la base de dato fue fallida.		
	<b>Actor</b>	<b>Sistema</b>
4a		Si no se puede establecer conexión con la base de datos, el sistema muestra el siguiente mensaje: "No se puede establecer conexión con los datos". Termina el CU.
<b>Flujos alternos</b>		
<b>5a. Evento:</b> No existen materiales que cumplan con los filtros introducidos.		
	<b>Actor</b>	<b>Sistema</b>
5a		Si no existen materiales que cumplan con los

		filtros introducidos por el usuario la lista de resultados no mostrará ningún elemento indicando que no existen coincidencias con los filtros introducidos. Termina el CU.
<b>Sección 2: "Limpiar".</b>		
<b>Flujo básico: Limpiar</b>		
	<b>Actor</b>	<b>Sistema</b>
<b>1</b>		Se reinician todos los valores introducidos en los filtros de búsqueda de todos los tipos de búsqueda. El sistema da la posibilidad de volver a insertar valores en los filtros. Regresa al paso 2 del flujo básico Búsqueda avanzada.
<b>Relaciones</b>	CU Incluidos	No existe.
	CU Extendidos	Reproducir Material Gestionar Solicitud
<b>Requisitos no funcionales</b>	RnF1 RnF2 RnF3	
<b>Asuntos pendientes</b>	No existen.	

## 2.5. Patrones de diseño de software utilizados para el Subsistema de Solicitud y Préstamo del sistema Agorav para Android

La arquitectura de software es un conjunto de patrones que proporcionan un marco de referencia necesario para guiar la construcción de un software, permitiendo a los programadores, analistas y todo el conjunto de desarrolladores del software compartir una misma línea de trabajo y cubrir todos los objetivos

y restricciones de la aplicación. Es considerada el nivel más alto en el diseño de la arquitectura de un sistema, puesto que establecen la estructura, funcionamiento e interacción entre las partes del software (González, 2014).

### **2.5.1. Patrones Generales de Software para Asignar Responsabilidades (GRASP)**

Un patrón describe un problema que ocurre varias veces, así como el núcleo de la solución al inconveniente, de forma que puede utilizarse en ilimitadas ocasiones sin tener que hacer dos veces lo mismo. (Larman, 2003).

En la solución del sistema se aplicaron principalmente los siguientes patrones de asignación de responsabilidades:

**Controlador:** es utilizado por todas las clases implicadas en la capa de presentación de la lógica del negocio. Poseen la responsabilidad de controlar el flujo de eventos mediante las actividades correspondientes. En la arquitectura de la aplicación se definió una clase controladora (*SolicitudService*) que es la encargada de controlar la lógica del negocio en lo que respecta a sus CU correspondientes.

**Creador:** Este patrón ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. Este patrón se evidencia en la clase *ReproductorActivity.java* que crea una lista de objetos de la clase *SubClip.java*, para acceder a los elementos de esta clase.

**Experto:** Este patrón indica que la clase que cuenta con la información necesaria para cumplir la responsabilidad es la responsable de manejar la información. En el Subsistema de Solicitud y Préstamo, específicamente en la clase *SolicitudActivity.java* se evidencia este patrón ya que es la única que tiene la información necesaria para crear la vista *solicitud.xml*.

**Alta Cohesión:** Este patrón plantea que la información que almacena una clase debe de ser coherente y debe estar (en la medida de lo posible) relacionada con la clase. El uso de dicho patrón queda evidenciado en las clases modelos *SubClip.java*, *Video.java* las cuales poseen la estructura necesaria para guardar la información de forma coherente y de acuerdo a su responsabilidad.

**Bajo acoplamiento:** Este patrón es utilizado en la creación de clases independientes para la lógica de los diferentes tipos de clases, actividades y entidades. Esto trae como ventaja que solo se realicen acciones sobre el tipo de entidad que se solicite o formulario correspondiente y no sobre todo el conjunto. Estas clases se encargan de la representación de los elementos reales de cada uno respectivamente y las actividades, de manejar los componentes visuales de cada funcionalidad dentro de los diferentes CU.

### **2.5.2. Patrón arquitectónico Modelo Vista Controlador**

Según (IAVILAE, 2013), Modelo Vista Controlador (MVC), es un patrón que divide una aplicación en tres niveles distintos, uno que representa a la interfaz gráfica (Vista), otro que representa el tratamiento de datos (Modelo) y otro que se encarga de toda la lógica que se tiene que llevar a cabo por la aplicación (Controlador). Esto se hace para permitir una mayor portabilidad de una aplicación, e incluso facilitar su mantenimiento. Pues si lo que se quiere es modificar la apariencia de la aplicación sólo se modifica la capa de la Vista, si se quiere cambiar de sistema de almacenamiento de datos sólo se cambia la capa del Modelo, y si lo que se quiere es portar la aplicación a otra plataforma lo que se hará será modificar la capa del Controlador, al igual que si se quiere modificar el código para optimizar rendimiento.

El sistema a desarrollar para la plataforma Android utiliza el patrón arquitectónico Modelo Vista Controlador (MVC) donde el modelo representa la construcción basada en la información con la que operará la aplicación. En esta parte del modelo también está la decisión de comunicarse mediante servicios web. La vista es la interfaz con la que el usuario interactúa que son construidas en XML. El controlador son todas las clases que le proporcionan información a las interfaces. Estos controladores se programan en lenguaje Java y son el núcleo de la aplicación.

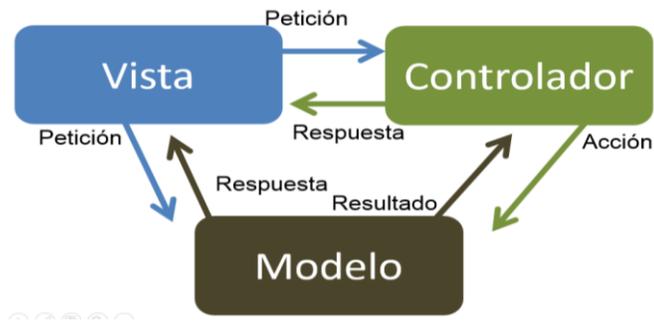


Imagen 3. Modelo Vista Controlador.

## 2.6. Diagrama de clase del diseño

Un diagrama de clases del diseño (DCD) describe gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación. Muestra las relaciones existentes entre las clases del sistema. Son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema (Pressman, 2010). A continuación se muestra el DCD del CU Buscar Material.

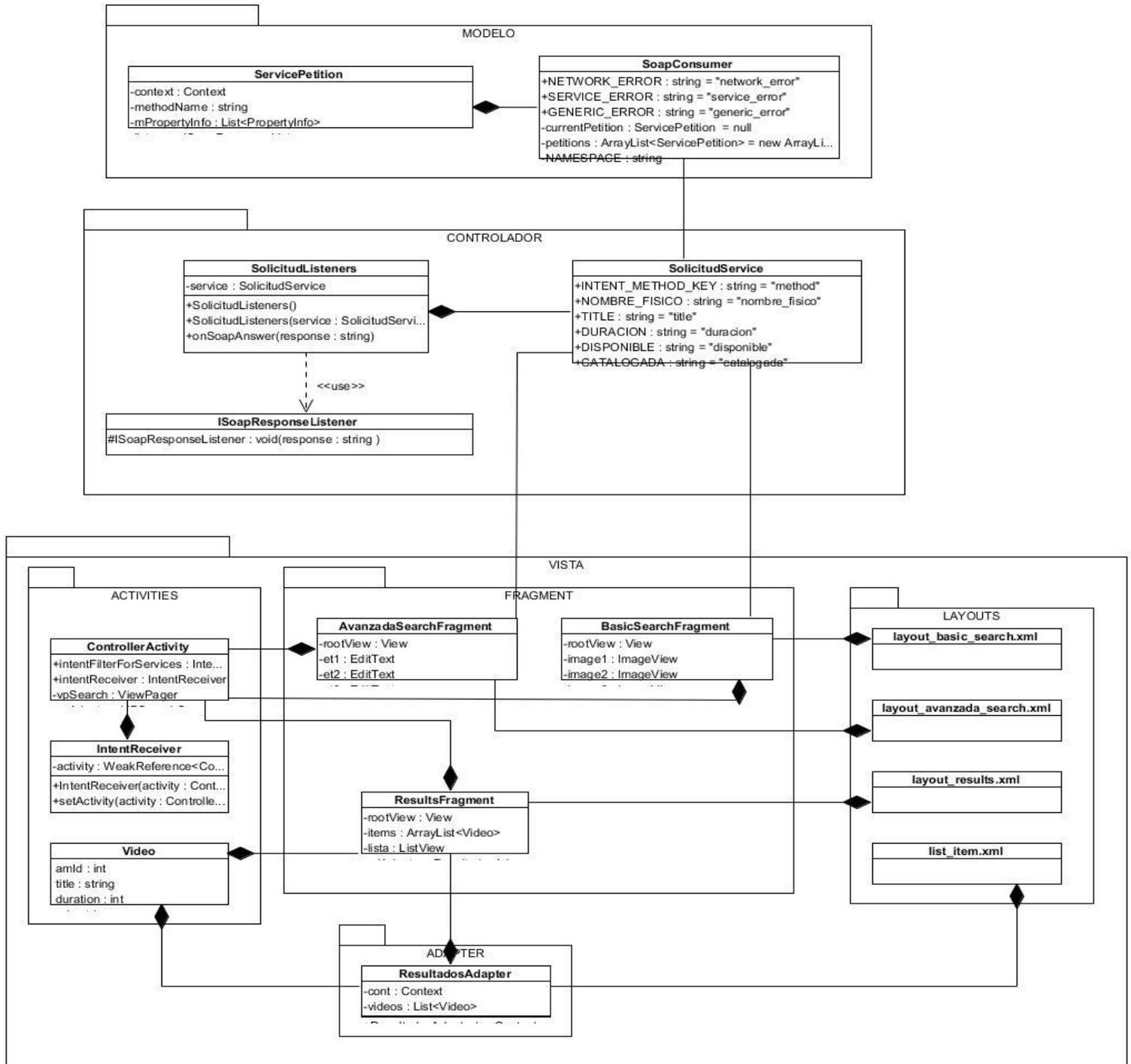


Imagen 4. DCD del CU Buscar Material.

## Conclusiones del Capítulo 2

- La confección del diagrama de clases del dominio permitió visualizar la relación existente entre los conceptos u objetos significativos en el dominio del problema.
- Con el modelo del dominio, la obtención de los requisitos funcionales y no funcionales, de conjunto con el modelado del sistema, propiciaron el entendimiento entre el equipo de desarrollo y el cliente de la aplicación en función de lo que el sistema debe realizar y las características que debe poseer.
- El diseño de los diagramas de CU del sistema brindó la posibilidad de mostrar gráficamente las funcionalidades a desarrollar y su interacción con los actores que las ejecutarán.
- La descripción de los CU permitió comprender el flujo de eventos que la aplicación debe cumplir para llevar a cabalidad los objetivos trazados.
- La utilización de los patrones arquitectónicos y de diseño seleccionados proporcionaron un correcto diseño del sistema.
- Se obtuvieron además los artefactos correspondientes al flujo de trabajo de Diseño los cuales sirven como guía para comenzar su implementación.

## **Capítulo 3: Implementación y pruebas del Subsistema de Solicitud y Préstamo del sistema Agorav para Android**

En el presente capítulo se tendrán en cuenta todos los aspectos del diseño del sistema con el fin de llevar a cabo el desarrollo de los flujos de trabajo de implementación y prueba. Se muestra la situación física de los distintos componentes lógicos desarrollados a través del modelo de despliegue y la organización del sistema mediante el modelo de componentes el cual representa la vista estática del sistema. Se explican los estilos de programación empleados. Y por último se define el modelo de prueba, describiendo como hacer las pruebas al sistema verificando que todos los requisitos hayan sido implementados y determinando los defectos de los mismos.

### **3.1. Diagrama de componentes**

Los diagramas de componentes ilustran las partes del software que conformarán un sistema. Un diagrama de componentes tiene un nivel más alto de abstracción que un diagrama de clase, usualmente un componente se implementa por una o más clases (u objetos) en tiempo de ejecución. Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre dichos elementos. Es un grafo de componentes unidos a través de relaciones que pueden ser de compilación o de ejecución, y además se pueden representar las interfaces de esos componentes (SparxSystems, 2007).

Es otra forma de representar una vista estática del sistema, que muestra la organización y dependencia que existe entre los componentes físicos que se necesitan para ejecutar la aplicación, sean estos componentes de código fuente, librerías, binarios o ejecutables. A continuación se muestra el diagrama de componentes del CU Buscar Material:

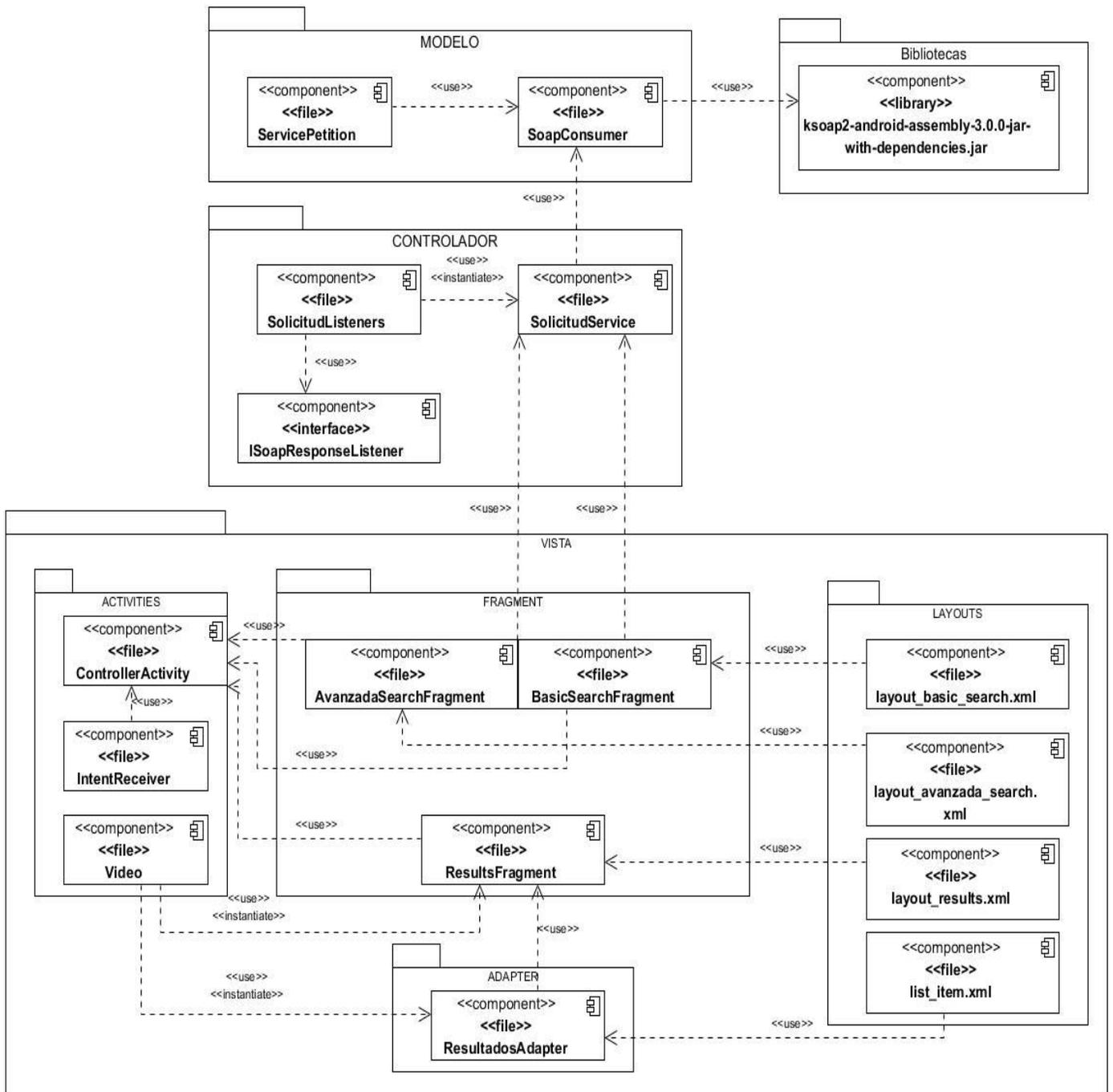


Imagen 5. Diagrama de componentes del CU Buscar Material.

### 3.2. Modelo de Despliegue

Los diagramas de despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y la distribución de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria (Marca, y otros, 2010).

El modelo de despliegue suministra la base para la comprensión de la distribución física de un sistema a través de nodos, indicando de qué forma se sitúa el software en el hardware que lo contiene.

A continuación se presenta el modelo despliegue del Subsistema de Solicitud y Préstamo del sistema Agorav para Android:

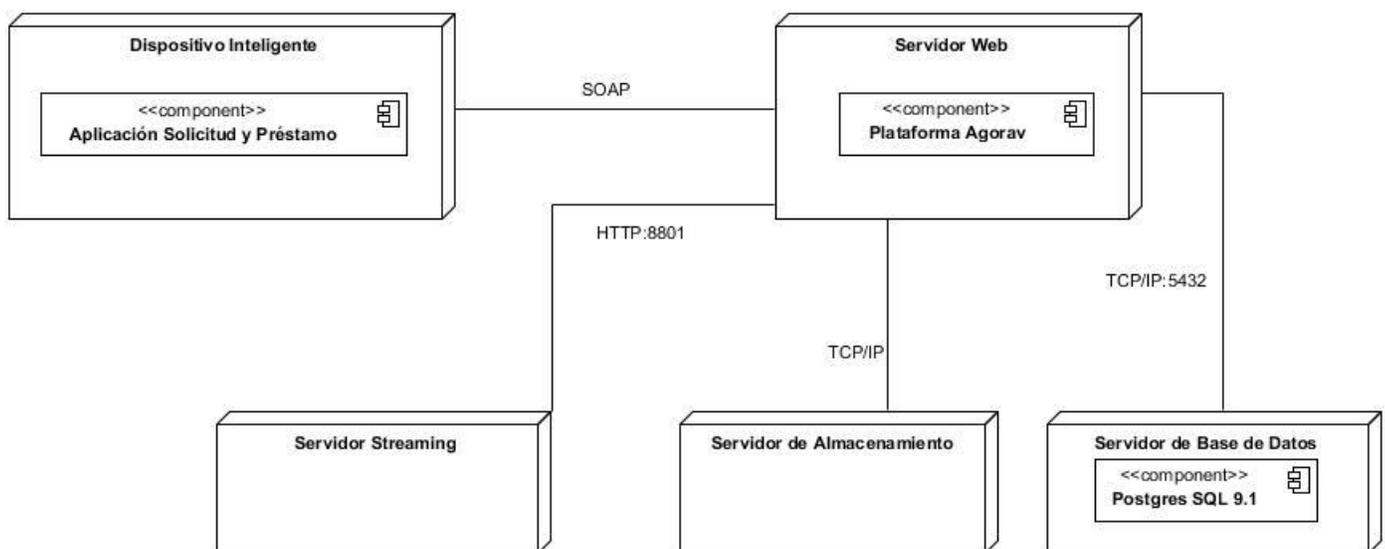


Imagen 6. Diagrama de Despliegue.

### 3.2.1. Descripción de componentes

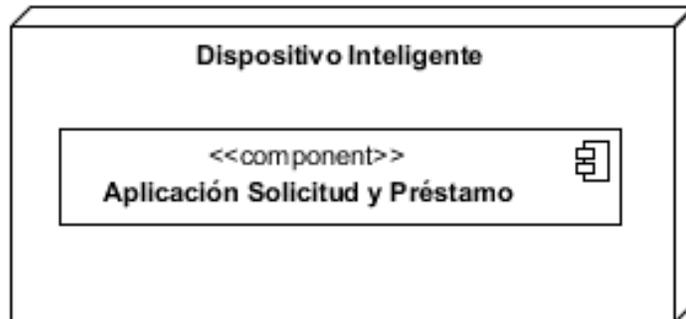


Imagen 7. Nodo Dispositivo Inteligente.

**Nodo Dispositivo Inteligente:** En este nodo se instala la aplicación implementada desde la cual se consumirán los servicios presentes en el sistema Agorav.

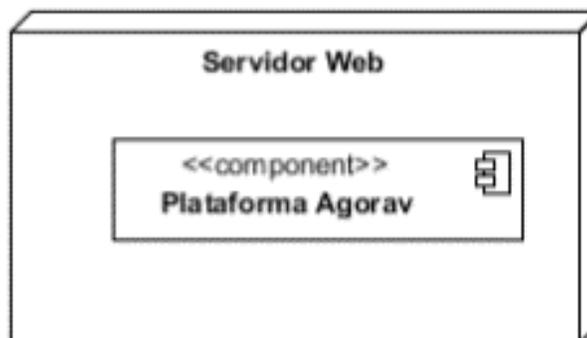


Imagen 8. Nodo Servidor Web.

**Nodo Servidor Web:** En este nodo estará ejecutando el sistema Agorav

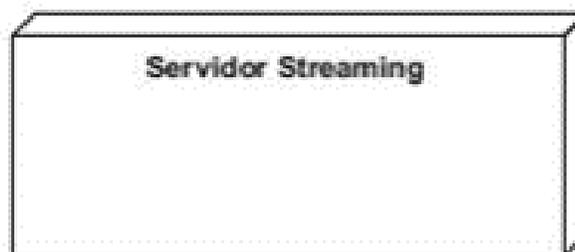
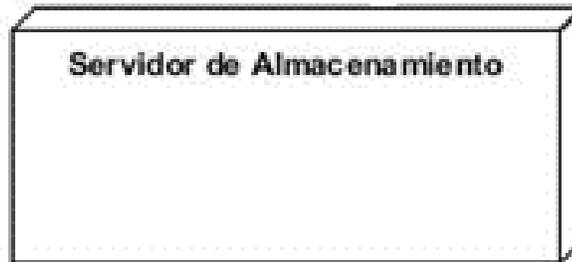


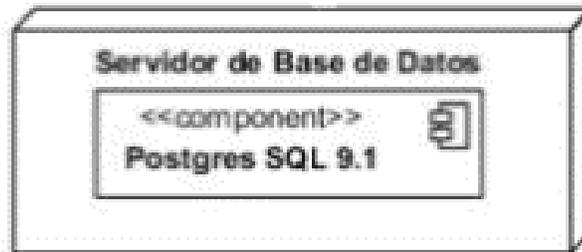
Imagen 9. Nodo Servidor Streaming.

**Nodo Servidor *Streaming*:** En este nodo se guardarán la dirección de los materiales audiovisuales que se visualizarán utilizando la tecnología *streaming*.



*Imagen 10. Nodo Servidor de Almacenamiento.*

**Nodo Servidor de Almacenamiento:** En este nodo se guardan los materiales audiovisuales que sean insertados en el sistema.



*Imagen 11. Nodo Servidor de Base de Datos.*

**Nodo Servidor de Base de Datos:** En este nodo se almacenarán los datos referentes a los materiales audiovisuales presentes en el sistema Agorav.

### 3.3. Estándar y estilos de codificación

Una guía de estilo es un conjunto de reglas para dar formato a los programas. El tener un estilo uniforme facilita la lectura del código. Seguir normas de estilo agiliza encontrar la esencia de los programas (López Gaona, 2012). Al comenzar un software, se establece un estándar de codificación para asegurarse de que todos los programadores del proyecto trabajen de forma coordinada.

Para lograr la legibilidad de un programa es importante considerar aspectos tales como el nombre de los identificadores, escribir el código con cierta alineación y líneas en blanco en lugares apropiados así como realizar una buena documentación.

Para el desarrollo de la herramienta se definieron los siguientes estándares de codificación, los cuales se explican a continuación.

#### **Identificadores:**

Los identificadores deben ser elegidos de tal manera que el nombre describa el uso que se dará dentro del programa, por tanto no es recomendable usar identificadores de una letra, excepto en el *for*, ni abreviaturas raras o ambiguas.

Además de eso es recomendable que se escriban:

- Empezando con mayúscula si se trata del nombre de una clase o interfaz, y empezando cada palabra en identificador con mayúscula.
- Empezando con minúscula si es el nombre de cualquier otro identificador. De preferencia el nombre de cualquier método debe ser un verbo en infinitivo y el de todo atributo un sustantivo.

Ejemplo:

```

//Acción del botón insertar subclip
select.setOnClickListener(new OnClickListener()
{
    @Override
    public void onClick(View arg0)
    {
        if (valueInicio != 0 || valueFin != totalDuration)
        {
            lista.add(new SubClip(getTimeI(), getTimeF(), 1, nombre));
            Toast.makeText(getApplicationContext(), "Subclip insertado", Toast.LENGTH_SHORT).show();
        }
    }
});

```

### Archivos fuente:

Cada programa en Java es una colección de uno o más archivos. El programa ejecutable se obtiene compilando estos archivos. En cada archivo especifica su contenido como sigue:

1. Los paquetes (instrucción *package*).
2. Los archivos de biblioteca (Instrucciones *import*).
3. Un comentario explicando el objetivo del archivo.
4. Las clases que defines en ese archivo.

Ejemplo:

```

package cu.uci.solicitudprestamo.activity; //Paquete de vista

```

### Clases:

Cada clase debe ir precedida por un comentario que explique su objetivo. Es recomendable especificar sus elementos como sigue:

1. Estructura de los objetos. Primero las variables y luego las constantes.
2. Elementos estáticos.

3. Constructores.

4. Métodos públicos y privados.

5. Métodos estáticos.

6. Clases internas.

Dejar una línea en blanco después de cada método.

Todos los elementos deben estar precedidos por *public*, *private* o *protected*.

**Alineación y espacios en blanco:**

- Todos los bloques deben estar alineados de tal manera que sean fácilmente distinguibles. Dentro de un bloque todas las instrucciones van a la misma altura.
- En proposiciones compuestas el paréntesis que abre debe ir en la misma línea que empieza tal proposición. El paréntesis que cierra debe estar sólo en la última línea de la proposición y a la misma altura que la línea de inicio.
- Usa líneas en blanco para separar partes de un método que son lógicamente distintas.

Ejemplo:

```

//Acción del botón insertar media completa y mostrar la lista de subclip insertados
accept.setOnClickListener(new OnClickListener()
{
    @Override
    public void onClick(View arg0)
    {
        if (totalDuration>0)
        {
            lista.add(0, new SubClip("00:00:00", getTimeD(), 1,nombre));
            Intent a = new Intent(getApplicationContext(),ListaSubClip.class);
            a.putParcelableArrayListExtra("list", lista);
            startActivity(a);
        }
        else
        {
            Toast.makeText(
                getApplicationContext(),
                "No se puede realizar la solicitud puesto que el material no se cargó correctamente.",
                Toast.LENGTH_SHORT).show();
        }
    }
});

```

### 3.4. Proceso de pruebas

La realización de pruebas durante el desarrollo de software es una tarea a la que se debe prestar suma importancia, dado que estas garantizarán la calidad del producto final y representan una revisión de las especificaciones, el diseño y la codificación.

#### 3.4.1. Estrategia de pruebas

La Estrategia de Prueba de software integra un conjunto de actividades que describen los pasos que hay que llevar a cabo en un proceso de prueba: la planificación, el diseño de casos de prueba, la ejecución y los resultados, tomando en consideración cuánto esfuerzo y recursos se van a requerir, con el fin de obtener como resultado una correcta construcción del software (Anna. C Grimán, 2014).

Según (Pressman, 2010) cuando se le van a aplicar pruebas a un software, se tienen en cuenta una serie de objetivos en diferentes escenarios y niveles de trabajo. Los niveles de pruebas verifican y validan un producto de software en diferentes ángulos con la capacidad de determinar no conformidades en relación con el uso que se le vaya a dar a la herramienta. Existen varios niveles de pruebas posibles a realizar entre las que se encuentran: las unitarias, las de integración, las de sistema y las de aceptación. Se

decidió aplicar el nivel de prueba de sistema, ya que estas tienen como objetivo verificar el sistema software para comprobar si este cumple sus requisitos.

### **3.4.2. Tipo de prueba**

Existen varios tipos de pruebas que se le pueden realizar a un software. En este caso se realizará el tipo de prueba de funcionalidad, las cuales se caracterizan por ser específicas, concretas y exhaustivas, con la meta de probar y validar que el software hace lo que debe y sobre todo, lo que se ha especificado.

### **3.4.3. Método de prueba**

*“Las pruebas de caja negra, también denominada prueba de comportamiento, se centran en los requisitos funcionales del software. O sea, la prueba de caja negra permite obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Se trata de un enfoque complementario que intenta descubrir diferentes tipos de errores” (Pressman, 2005).*

Las pruebas de caja negra ignoran intencionadamente; la estructura de control, centrandose su atención en el campo de la información, es decir que estas pruebas se llevan a cabo sobre la interfaz del software, obviando el comportamiento interno y la estructura del programa. Los casos de prueba de la caja negra pretenden demostrar que las funciones del software son operativas, la entrada se acepta de forma correcta, se produce una salida correcta y la integridad de la información externa se mantiene.

### **3.4.4. Técnica de prueba**

Dentro del método de Caja Negra la técnica de partición de equivalencia es una de las más efectivas, pues permite examinar los valores válidos e inválidos de las entradas existentes en la herramienta.

Según (Pressman, 2005) *“La partición de equivalencia es 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 (por ejemplo, proceso incorrecto de todos los datos de carácter) que, de otro modo, requerirán la ejecución de muchos casos antes de detectar el error genérico. La Partición de Equivalencia 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”.*

### 3.5. Diseño de casos de pruebas

Tabla 3. Secciones de prueba para el CU Gestionar Solicitud.

Nombre de Sección	Descripción
Sección 1: Adicionar solicitud	Se inserta en la base de datos del sistema Agorav una solicitud donde quedan recogidos los datos del cliente y de la media o subclip deseados.
Sección 2: Modificar solicitud	Se modifica una solicitud hecha por un usuario
Sección 3: Eliminar solicitud	Se elimina una solicitud hecha por un usuario
Sección 4: Listar solicitudes	Muestra una lista con las solicitudes hechas desde un DI determinado, según su mac.

Tabla 4. Descripción de las variables correspondientes al caso de prueba para el CU Gestionar Solicitud.

No	Nombre del campo	Clasificación	Valor nulo	Descripción
V1	Nombre del cliente	Campo de texto	no	Campo que solo admite caracteres alfabéticos.
V2	Apellidos del cliente	Campo de texto	no	Campo que solo admite caracteres alfabéticos.
V3	Carnet de Identidad del cliente	Campo de texto	no	Campo que solo admite caracteres numéricos. Estrictamente 11

				caracteres.
<b>V4</b>	Institución	Campo de texto	no	Campo que admite caracteres de cualquier tipo.
<b>V5</b>	Título de la media	Campo de texto	no	Este campo se llenará automáticamente al presionar el botón solicitar, dentro de la lista de subclips realizados, tomando el valor del campo nombre que presente en la lista.
<b>V6</b>	Tiempo de inicio	Campo de texto	no	Este campo se llenará automáticamente al presionar el botón solicitar, dentro de la lista de subclips realizados, tomando el valor del campo marca inicio que presente en la lista.
<b>V7</b>	Tiempo de fin	Campo de texto	no	Este campo se llenará automáticamente al presionar el botón solicitar, dentro de la lista de subclips realizados, tomando el valor del campo marca fin que presente en la lista.
<b>V8</b>	Formato	Selección	no	Se proporcionan dos tipos de formatos de los cuales se debe elegir uno.

A continuación se presentan los casos de prueba correspondientes al CU “Gestionar Solicitud”. El resto de los casos de prueba pueden ser consultados en los **Anexos**.

### 3.5.1. Caso de prueba para el CU “Gestionar Solicitud”

Tabla 5. Caso de prueba para el CU Gestionar Solicitud: Sección 1: “Adicionar solicitud”.

Sección 1: “Adicionar solicitud”											
Escenario	Descripción	V1	V2	V3	V4	V5	V6	V7	V8	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Insertar datos de la solicitud correctamente.	Se insertan todos los datos de la solicitud.	V	V	V	V	V	V	V	V	Se inserta una solicitud en la base de datos del sistema. Muestra la lista de solicitudes realizadas y muestra un mensaje indicando que la solicitud fue insertada satisfactoriamente	1. Se presiona el botón insertar solicitud de la lista de subclib realizados. 2. Se introducen los datos. 3. Se presiona el botón aceptar.
<b>EC 1.2</b> Dejar campos vacíos	Se intenta insertar una solicitud dejando campos vacíos.	( )	( )	( )	( )	V	V	V	( )	Muestra un mensaje indicando que campo le falta por llenar.	1. Se presiona el botón insertar solicitud de la lista de subclib realizados. 2. Se introducen los datos. 3. Se presiona el botón aceptar. 4. Se muestra un mensaje de error.
<b>EC 1.3</b> Insertar datos	Se intenta insertar una	I	I	I	V	V	V	V	V	Muestra un mensaje	1. Se presiona el botón

inválidos	solicitud con datos incorrectos.									indicando en que campo ha introducido datos incorrectos.	<p>insertar solicitud de la lista de subclib realizados.</p> <p>2. Se introducen los datos.</p> <p>3. Se presiona el botón aceptar.</p> <p>4. Se muestra un mensaje de error.</p>
-----------	----------------------------------	--	--	--	--	--	--	--	--	--	---

Tabla 6. Caso de prueba para el CU Gestionar Solicitud: Sección 1: "Modificar solicitud".

Sección 2: "Modificar solicitud"						
Escenario	Descripción	V6	V7	V8	Respuesta del sistema	Flujo central
<b>EC 2.1</b> Insertar datos modificados de la solicitud correctamente.	Se guardan correctamente los datos modificados.	V	V	V	Se guarda la solicitud con los nuevos datos modificados. Muestra un mensaje indicando que la solicitud fue modificada satisfactoriamente.	<p>1. Se presiona el botón modificar.</p> <p>2. Se modifican los datos.</p> <p>3. Se presiona el botón aceptar.</p>
<b>EC 2.2</b> Dejar campos vacíos	Se intenta modificar una solicitud dejando campos vacíos.	( )	( )	( )	Muestra un mensaje indicando que campo le falta por llenar.	<p>1. Se presiona el botón modificar.</p> <p>2. Se modifican los</p>

						datos. 3. Se presiona el botón aceptar. 4. Se muestra un mensaje de error.
<b>EC 2.3</b> Editar con datos inválidos	Se intenta modificar una solicitud con datos incorrectos.	I	I	I	Muestra un mensaje indicando en que campo ha introducido datos incorrectos.	1. Se presiona el botón modificar. 2. Se modifican los datos. 3. Se presiona el botón aceptar. 4. Se muestra un mensaje de error.

Tabla 7. Caso de prueba para el CU Gestionar Solicitud: Sección 1: "Eliminar solicitud".

<b>Sección 3: "Eliminar solicitud"</b>			
<b>Escenario</b>	<b>Descripción</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
<b>EC 3.1</b> Eliminar correctamente una solicitud	Se elimina una solicitud de la base de datos sistema.	Elimina una solicitud de la base de datos del sistema.	1. Se presiona el botón eliminar. 2. Se muestra un mensaje comunicando que la solicitud ha sido eliminada satisfactoriamente.

Tabla 8. Caso de prueba para el CU Gestionar Solicitud: Sección 1: "Listar solicitudes".

<b>Sección 4: "Listar solicitudes"</b>			
<b>Escenario</b>	<b>Descripción</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
<b>EC 3.1</b> Listar las solicitudes realizadas desde un determinado DI, según su mac.	Se listan y muestran al cliente todas las solicitudes realizadas desde ese DI.	Muestra al usuario todas las solicitudes realizadas desde un determinado DI, según su mac.	1. Se presiona el botón listar solicitudes, presente en la barra de tareas de las vistas de las actividades del CU Buscar Material (Búsqueda básica, Búsqueda avanzada, Resultados).

### 3.6. Resultados de las Pruebas

Para validar el correcto funcionamiento de la aplicación fueron realizadas tres iteraciones de pruebas. En la siguiente gráfica se muestra un resumen general de todas las dificultades encontradas por iteraciones en las pruebas realizadas, donde:

NCE: No Conformidades encontradas

NCR: No Conformidades resueltas

NCP: No Conformidades pendientes

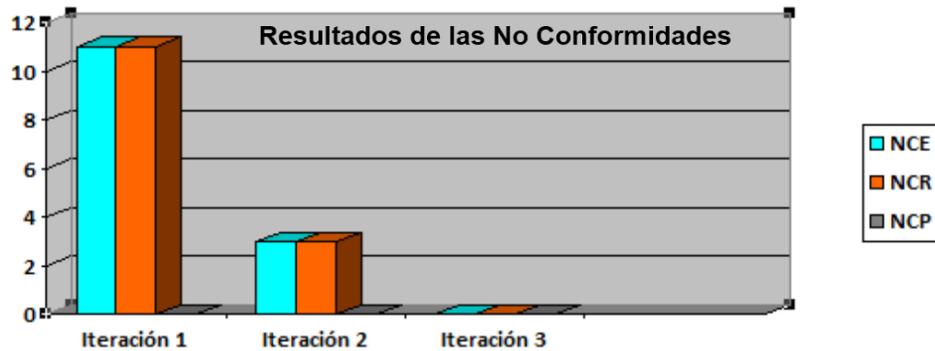


Imagen 12. Gráfico de resultado de las no conformidades.

En el gráfico anterior se muestran las once no conformidades encontradas en una primera iteración, ocho eran de validación y tres de errores ortográficos, estas fueron corregidas posteriormente. Una segunda iteración arrojó tres nuevas no conformidades en la validación y fueron corregidas. Al realizar la tercera iteración propuesta no se encontraron no conformidades.

### Conclusiones del Capítulo 3

- Mediante el diagrama de componentes se representó una vista estática de la aplicación, mostrando la organización y dependencia que existe entre los componentes físicos que se necesitan para ejecutar la misma.
- La realización del diagrama de despliegue permitió comprender la distribución física del subsistema.
- El uso de los estándares de codificación y estilos de programación permiten el entendimiento del código por otros programadores que no sean del equipo de desarrollo en caso de mejoras a la aplicación en un futuro.
- Con el desarrollo del proceso de pruebas se logró identificar y resolver los errores en la implementación aumentando la calidad del producto final obtenido.
- Se comprobó que la aplicación satisface las funcionalidades requeridas.

### **Conclusiones Generales**

Con la investigación e implementación descritas, se logró dar cumplimiento a las tareas de la investigación, así como al objetivo planteado inicialmente, obteniendo como resultado una aplicación que cumple con todos los requisitos y la calidad requerida, lo cual permite arribar a las siguientes conclusiones:

- A partir de la realización del análisis y diseño del subsistema se obtuvieron como resultado los diagramas y artefactos necesarios para guiar el desarrollo del mismo.
- Los 18 Requisitos Funcionales que se definieron fueron debidamente implementados y se incluyó en la propuesta de solución las exigencias de todos los Requisitos No Funcionales detectados.
- Las tecnologías y herramientas empleadas, posibilitaron el desarrollo de una aplicación que cumple con los requisitos planteados, obteniéndose así, un producto en correspondencia a los resultados esperados.
- Los diseños de Casos de Prueba desarrollados, como parte de las pruebas de Caja Negra y la ejecución de las pruebas de funcionalidad permitieron validar los requisitos de la aplicación.
- Como resultado se obtuvo el Subsistema de Solicitud y Préstamo del sistema Agorav para Android, el cual permite la correcta visualización y funcionamiento del Subsistema de Solicitud y Préstamo presente en el sistema Agorav en DI con SO Android, además permite la búsqueda, visualización, solicitud y descarga de los materiales audiovisuales con que cuenta este sistema.

## **Recomendaciones**

Al concluir la investigación y el desarrollo del Subsistema de Solicitud y Préstamo de Agorav para Android, se proponen las siguientes recomendaciones:

- Desarrollar el Subsistema de Solicitud y Préstamo de Agorav, en otras plataformas móviles, como iOS o Windows Phone.

## Referencias bibliográficas

**Adames, Rheiber. 2013.** monografias.com. *monografias.com*. [En línea] 4 de julio de 2013. [Citado el: 10 de diciembre de 2014.] <http://www.monografias.com/trabajos99/el-telefono/el-telefono2.shtml>.

**Alegsa, Leandro. 2009.** Definición de Requisitos (desarrollo de sistemas/software). [En línea] 2009. [Citado el: 21 de marzo de 2014.] <http://www.alegsa.com.ar/Dic/requerimientos.php>.

**Alegsa, Leandro. 2009.** ALEGSA.com.ar. *ALEGSA.com.ar*. [En línea] 2009. <http://www.alegsa.com.ar/Dic/lenguajes%20de%20programacion.php>.

**Alegsa, Leandro. 2010.** ALEGSA.com.ar. *ALEGSA.com.ar*. [En línea] 2010. <http://www.alegsa.com.ar/Dic/ide.php>.

**Alegsa, Leandro. 2014.** ALEGSA.com.ar. *ALEGSA.com.ar*. [En línea] 2014. <http://www.alegsa.com.ar/Dic/smartphone.php>.

**Álvarez, Miguel Angel. 2001.** desarrolloweb.com. *desarrolloweb.com*. [En línea] 18 de julio de 2001. <http://www.desarrolloweb.com/articulos/497.php>.

**Angeles, Sara. 2014.** BUSINESS NEW DAILY. *BUSINESS NEW DAILY*. [En línea] 15 de julio de 2014. [Citado el: 15 de junio de 2015.] [www.businessnewsdaily.com/6783-mobile-website-vs-mobile-app.html](http://www.businessnewsdaily.com/6783-mobile-website-vs-mobile-app.html).

**Anna. C Grimán, María Pérez, Luis. E Mendoza. 2014.** Laboratorio de Investigación en Sistemas de Información. *Laboratorio de Investigación en Sistemas de Información*. [En línea] 30 de septiembre de 2014. [Citado el: 20 de mayo de 2015.] [http://www.lisi.usb.ve/publicaciones/03%20evaluacion/evaluacion\\_15.pdf](http://www.lisi.usb.ve/publicaciones/03%20evaluacion/evaluacion_15.pdf).

**Aperador, Miguel. 2012.** Prezi. *Prezi*. [En línea] 1 de octubre de 2012. [Citado el: 15 de junio de 2015.] <https://prezi.com/lwbginvwiod4/caracteristicas-de-los-servicios-web-y-como-consumirlos>.

**Belmonte, Oscar, Granell, Carlos y Erdozain, María. 2010.** Desarrollo de proyectos informáticos con tecnología Java. [En línea] 2010. [Citado el: 15 de abril de 2014.] <http://www3.uji.es/~belfern/pdf/libroJavaConTapa.pdf>.

- Beech, Alex. 2015.** softonic. *softonic*. [En línea] 23 de abril de 2015. [Citado el: 29 de abril de 2015.] <http://youtube.softonic.com/android>.
- Paradigm, Visual. 2014.** Visual Paradigm. *Visual Paradigm*. [En línea] 2014. <http://www.visual-paradigm.com/>.
- Calahorro, Nacho Brito. 2009.** *Manual\_de\_desarrollo\_web\_con\_Grails*. Madrid : ImaginaWorks Software Factory, 2009. ISBN: 978-84-613-2651.
- Castells, Manuel. 2013.** OpenMind. *OpenMind*. [En línea] 2013. [Citado el: 15 de junio de 2015.] <https://www.bbvaopenmind.com/articulo/el-impacto-de-internet-en-la-sociedad-una-perspectiva-global/?fullscreen=true>.
- Cloninger, Eric. 2012.** eclipse marketplace. *eclipse marketplace*. [En línea] 9 de mayo de 2012. [Citado el: 12 de diciembre de 2014.] <http://marketplace.eclipse.org/content/android-development-tools-eclipse>.
- Corona, Ernesto Gutiérrez. 2013.** UDLAP. UDLAP. [En línea] 8 de junio de 2013. [Citado el: 24 de junio de 2015.] [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/mcc/gutierrez\\_c\\_e/capitulo1.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/mcc/gutierrez_c_e/capitulo1.pdf).
- Cruz, Carlos. 2012.** PortalProgramas. *PortalProgramas*. [En línea] 9 de enero de 2012. [Citado el: 12 de diciembre de 2015.] <http://www.portalprogramas.com/vimeo/android/>.
- Developer. 2014.** *Developer*. [En línea] 15 de abril de 2014. [Citado el: 22 de abril de 2015.] <http://developer.android.com/tools/sdk/tools-notes.html>.
- Edmondson, Ray. 2002.** UNESCO. [En línea] UNESCO, 2002. [Citado el: 14 de junio de 2015.] <http://www.unesco.org/webworld/publications/philos/philos3.htm>.
- Edukanda. 2010.** Los formatos digitales. *Edukanda*. [En línea] 2010. [Citado el: 14 de junio de 2015.] [http://www.edukanda.es/mediatecaweb/data/zip/1163/page\\_15.htm](http://www.edukanda.es/mediatecaweb/data/zip/1163/page_15.htm).
- Flores, Miriam Milagros Díaz. 2012.** slideshare. *slideshare*. [En línea] 13 de noviembre de 2012. [Citado el: 14 de diciembre de 2014.] [http://es.slideshare.net/Portal\\_do\\_estudiante\\_ADS/rup-vs-xp](http://es.slideshare.net/Portal_do_estudiante_ADS/rup-vs-xp).

**Gonzalez, Alejandro Nieto. 2011.** Xataka Android. *Xataka Android*. [En línea] 8 de febrero de 2011. <http://www.xatakandroid.com/sistema-operativo/que-es-android>.

**González, Mónica. 2014.** slideshare. *slideshare*. [En línea] 16 de enero de 2014. [Citado el: 19 de febrero de 2015.] <http://es.slideshare.net/MonicaGlez1/ingeniera-de-software-30069503>.

**IAVILAE. 2013.** Proyecto Simio. *Proyecto Simio*. [En línea] 14 de diciembre de 2013. [Citado el: 20 de marzo de 2015.] <http://www.proyectosimio.com/es/programacion-android-base-de-datos-i-modelo-vista-controlador/>.

**INEI. 1999.** *Herramientas Case: COLECCION CULTURA INFORMATICA*. s.l. : Talleres de la Oficina de Impresiones de la Oficina Técnica de Difusión del Instituto Nacional de Estadística e Informática (INEI), 1999.

**Jiménez, Emmanuel. 2013 .** ANDROID AYUDA. *ANDROID AYUDA*. [En línea] 26 de julio de 2013 . [Citado el: 15 de junio de 2015.] <http://androidayuda.com/2013/07/26/desarrollando-para-android-i-android-studio/>.

**Kofler, Birgit. 1991.** [En línea] UNESCO, 1991. [Citado el: 14 de junio de 2015.] <http://www.unesco.org/webworld/publications/philos/philos3.htm>.

**Larman, Craig. 2003.** UML y Patrones. 2a Edición. 2003.

**Lana. 2013.** OMG(OBJECT MANAGEMENT GROUP). *OMG(OBJECT MANAGEMENT GROUP)*. [En línea] 16 de 12 de 2013. [Citado el: 12 de noviembre de 2014.] <http://www.omg.org/technology/readingroom/XML.htm>.

**Lana. 2013.** OMG(OBJECT MANAGEMENT GROUP). *OMG(OBJECT MANAGEMENT GROUP)*. [En línea] 16 de diciembre de 2013. [Citado el: 16 de noviembre de 2014.] <http://www.omg.org/technology/readingroom/SOA.htm>.

**Larman, Craig. 1999.** *UML y PATRONES. Introducción al análisis y diseño orientado a objetos*. México : PRENTICE HALL, 1999. ISBN: 970-17-0261-1.

**Lee, Wei-Meng. 2011.** *BEGINNING Android Application Development*. Indianapolis : Wiley Publishing, Inc, 2011. ISBN: 978-1-118-01711-1.

**Lorenzo, Amaya Alvarez, y otros. 2013.** Informática en salud. *Informática en salud*. [En línea] 2013. [Citado el: 13 de junio de 2015.] <http://www.informatica2013.sld.cu/index.php/informaticasalud/2013/paper/download/235/195>.

Marca, Hugo y Quisbert, Nancy. 2010. Diagrama de Despliegue. [En línea] 2010. [Citado el: 14 de marzo de 2014.] [virtual.usalesiana.edu.bo/web/practica/archiv/despliegue.doc](http://virtual.usalesiana.edu.bo/web/practica/archiv/despliegue.doc).

**Martinez, Chus. 2010.** PicandoJava.com. *PicandoJava.com*. [En línea] 29 de septiembre de 2010. [Citado el: 10 de marzo de 2015.] <https://sites.google.com/site/picandojavacom/mundo-java/ide-eclipse>.

**Martínez, Evelio. 2011.** Academia.edu. *Academia.edu*. [En línea] mayo de 2011. [Citado el: 10 de febrero de 2015.] [http://www.academia.edu/6042142/LECTURAS\\_La\\_evoluci%C3%B3n\\_de\\_la\\_telefon%C3%ADa\\_m%C3%B3vil\\_La\\_guerra\\_de\\_los\\_celulares](http://www.academia.edu/6042142/LECTURAS_La_evoluci%C3%B3n_de_la_telefon%C3%ADa_m%C3%B3vil_La_guerra_de_los_celulares).

**Márquez, Orlando Planchart. 2002.** Ponce. *Ponce*. [En línea] 2002. [Citado el: 11 de junio de 2015.] <http://ponce.inter.edu/cai/tesis/oplanchart/inicio.pdf#page=41>.

**Orozco, David. 2011.** CONCEPTODEFINICIÓN.DE. *CONCEPTODEFINICIÓN.DE*. [En línea] 1 de junio de 2011. [Citado el: 20 de noviembre de 2014.] <http://conceptodefinicion.de/android/>.

**Pérez, Miriam Ferreira. 2014.** Prezi. *Prezi*. [En línea] 20 de noviembre de 2014. [Citado el: junio de 10 de 2015.] <https://prezi.com/fvyjuksososa/tecnicas-tradicionales-para-el-registro-almacenamiento-y-re/>.

**Petrazzini, Gabriel Osmar Pedrozo. 2012.** UNIVERSIDAD NACIONAL DEL NORDESTE. *UNIVERSIDAD NACIONAL DEL NORDESTE*. [En línea] 2012. [Citado el: 15 de junio de 2015.] [http://exa.unne.edu.ar/informatica/SO/Sistemas\\_Operativos\\_en\\_Dispositivos\\_Moviles.pdf](http://exa.unne.edu.ar/informatica/SO/Sistemas_Operativos_en_Dispositivos_Moviles.pdf).

**Pressman, Roger S. 2005.** *Ingeniería del Software. Un enfoque Práctico*. 2005.

**Pressman, Roger S. 2010.** *Ingeniería de Software. Un enfoque Práctico*. 7ma Edición. 2010.

**Ramos, Rubén.** lincua.org-Asociacion de usuarios GNU/Linux en Cantabria. *lincua.org-Asociacion de usuarios GNU/Linux en Cantabria*. [En línea] 20 de marzo de 2003. [Citado el: 20 de octubre de 2014.] <http://www.lincua.org>.

**Schmuller, Joseph. 2001.** Aprendiendo UML en 24 horas. 2001. 968444463X.

SparxSystems. 2007. Diagrama de Componentes UML 2. [En línea] Sparx Systems Pty Ltd., 2007. [Citado el: 11 de marzo de 2015.] [http://www.sparxsystems.com.ar/resources/tutorial/uml2\\_componentdiagram.html](http://www.sparxsystems.com.ar/resources/tutorial/uml2_componentdiagram.html).

**Toledo, Gerardo. 2010.** Bitácoras. *Bitácoras*. [En línea] 2010. <http://bitacoras.com/anotaciones/la-solicitud-concepto-caracteristicas-estructura-ejemplo/7619611/>.

**Tugurium. 2006.** *Tugurium*. [En línea] 4 de septiembre de 2006. <http://www.tugurium.com/gti/termino.php?Tr=smart+device>.

**UCOL. 2006.** Las herramientas CASE. [En línea] Universidad de Colima, 2006. [Citado el: 12 de 03 de 2014.] [http://docente.ucol.mx/al961223/public\\_html/centro6.htm](http://docente.ucol.mx/al961223/public_html/centro6.htm).

**UNESCO.** [En línea] [Citado el: 12 de diciembre de 2014.] <http://www.unesco.org/webworld/publications/philos/philos3.htm>.

**Ugarte, Antonio. 2014.** Buenas Tareas. *Buenas Tareas*. [En línea] 4 de noviembre de 2014. [Citado el: 10 de marzo de 2015.] <http://www.buenastareas.com/ensayos/Nada/62282104.html>.

**Vessuri, Matias. 2014.** iPodTotal. *iPodTotal*. [En línea] 9 de julio de 2014. [Citado el: 16 de noviembre de 2014.] <http://www.ipodtotal.com/noticias/vimeo-apple-tv-se-actualiza-con-interfaz-simplificada-nuevas-funciones>.

## Anexos

### 1. Descripción de los CU

Tabla 9. Descripción del CU Reproducir Material.

<b>Objetivo</b>	Realizar acciones de reproducción seleccionadas por el actor.	
<b>Actores</b>	Usuario:(Inicia) reproducción de un material.	
<b>Resumen</b>	El CU se inicia cuando se selecciona el material o subclip que se desea solicitar. Las opciones de reproducción estarán disponibles mientras esté cargado en el reproductor un material. El CU termina cuando el usuario realiza alguna opción de reproducción sobre el material o la reproducción del mismo llega a su fin.	
<b>Complejidad</b>	Media	
<b>Prioridad</b>	Alta	
<b>Precondiciones</b>	Se ha obtenido la lista con los resultados de la búsqueda de las medias deseadas.	
<b>Poscondiciones</b>	Acciones de reproducción realizadas por el usuario sobre el material.	
<b>Flujo de eventos</b>		
<b>Flujo básico:</b> Reproducir material.		
	<b>Actor</b>	<b>Sistema</b>
1	Selecciona el material que va a solicitar.	
2		Carga el material seleccionado en el reproductor. El material comienza a reproducirse por defecto.
3	Puede realizar las siguientes acciones de reproducción: - Pausar/Continuar reproducción. - Parar reproducción. Ver sección1: "Parar reproducción". - Ir a un instante de tiempo. Ver sección2:"Ir a un instante de tiempo".	

	<ul style="list-style-type: none"> <li>- Controlar volumen. Ver sección3: “Controlar volumen”.</li> <li>- Crear marca de inicio /fin. Ver sección4: “Crear marca de inicio /fin”.</li> <li>- Reproducir fragmento entre marcas. Ver sección5: “Reproducir fragmento entre marcas”</li> <li>- Desplazarse entre marcas. Ver sección6: “Desplazarse entre marcas”.</li> <li>- Crear fragmento entre marcas. Ver sección7: “Crear fragmento de marcas”</li> <li>- Presiona el botón Seleccionar.</li> <li>- Presiona el botón Aceptar.</li> </ul>	
4	<p>Presiona el botón “Pausar o Continuar”.</p> <p><b>Observación:</b> Si el material está reproduciéndose el botón estará activo con la función de pausa, si el material está pausado el botón estará activo con la función de continuar la reproducción.</p>	
5		Se pausa o continúa la reproducción del material.
6	Presiona el botón “ Aceptar”	
7		El sistema muestra la lista de Subclip creada por el usuario. Termina el CU.
<b>Sección1:</b> “Parar reproducción”.		
<b>Flujo básico:</b> Parar reproducción.		
	<b>Actor</b>	<b>Sistema</b>
1	Presiona el botón “Parar”.	
		El reproductor sitúa la reproducción en pausa en el inicio del material. Termina el CU.
<b>Sección2:</b> “Ir a un instante de tiempo”.		
<b>Flujo básico:</b> Ir a un instante de tiempo		
	<b>Actor</b>	<b>Sistema</b>
1	El usuario puede situarse sobre una posición en la línea de tiempo de reproducción y presionar para ir hasta este instante de	

	tiempo.	
2		El reproductor sitúa la reproducción del material en el tiempo indicado por el usuario.
<b>Sección3:</b> “Controlar volumen”.		
<b>Flujo básico:</b> Controlar volumen.		
	<b>Actor</b>	<b>Sistema</b>
1	Desplaza el control de volumen hacia la dirección de aumento.	
2		Se percibe aumento en el volumen del sonido del material que se reproduce.
<b>Flujos alternos</b>		
<b>1a.Evento:</b> Disminuir el volumen.		
	<b>Actor</b>	<b>Sistema</b>
1a	Desplaza el control del volumen hacia la dirección de disminución.	
2a		Se percibe disminución en el volumen del sonido Del material que se reproduce.
<b>Sección4:</b> “Crear marca de inicio/fin”.		
<b>Flujo básico:</b> Crear marca de inicio/fin.		
	<b>Actor</b>	<b>Sistema</b>
1	Puede presionar los botones de creación de marcas de inicio y/o fin.	
2		El reproductor representa las marcas creadas sobre la línea de reproducción del material. El sistema habilita el botón “Reproducir fragmento entre marcas” El sistema habilita el botón “Ir a marca de inicio” El sistema habilita el botón “Ir a marca de fin”
<b>Sección5:</b> “Reproducir fragmento entre marcas”.		
<b>Flujo básico:</b> Reproducir fragmento entre marcas.		
	<b>Actor</b>	<b>Sistema</b>
1	Presiona el botón “Reproducir fragmento entre marcas”	
2		El reproductor comienza la reproducción del material desde el tiempo de la marca de inicio establecida y para la misma en el tiempo de la

		<p>Marca final establecida.</p> <p>Termina el CU.</p>
<b>Flujos alternos</b>		
<b>1a.Evento:</b> Solo se encuentra señalada la marca de inicio.		
	<b>Actor</b>	<b>Sistema</b>
<b>1a</b>		<p>Si el actor solo identificó la marca de inicio, el reproductor reproduce el material desde el tiempo que indica la marca de inicio hasta el final del material.</p> <p>Termina el CU.</p>
<b>Flujos alternos</b>		
<b>1b.Evento:</b> Solo se encuentra señalada la marca de fin.		
	<b>Actor</b>	<b>Sistema</b>
<b>1b</b>		<p>Si el actor solo identificó la marca de fin, el reproductor reproduce el material desde el tiempo de inicio hasta la marca de fin establecida.</p> <p>Termina el CU.</p>
<b>Sección6:</b> “Desplazarse entre marcas”.		
<b>Flujo básico:</b> Desplazarse entre marcas.		
	<b>Actor</b>	<b>Sistema</b>
<b>1</b>	Presiona el botón “Ir a marca de inicio”.	
<b>2</b>		El cursor de reproducción se sitúa en la marca de inicio establecida.
<b>Flujos alternos</b>		
<b>1aEvento:</b> Selecciona la opción ir a marca de fin.		
	<b>Actor</b>	<b>Sistema</b>
<b>1a</b>	Presiona el botón “Ir a marca de fin”.	
<b>2a</b>		El cursor de reproducción se sitúa en la marca de fin establecida.
<b>Sección7:</b> “Crear fragmento entre marcas”.		
<b>Flujo básico:</b> Crear fragmento entre marcas.		
	<b>Actor</b>	<b>Sistema</b>

1	Presiona el botón Seleccionar para crear el fragmento entre marcas.	
2		Se crea en la base de datos el fragmento seleccionado.
<b>Flujos alternos</b>		
<b>1a.Evento:</b> Selecciona la opción crear fragmento sin haber realizado marcas.		
	<b>Actor</b>	<b>Sistema</b>
1a		El sistema muestra un mensaje indicando que se debe realizar previamente las marcas de inicio y fin y después seleccionar el fragmento. Regresa al paso 1 del <b>Flujo básico</b> "Crear marca de inicio/fin".
<b>Relaciones</b>	<b>CU Incluidos</b>	No existen.
	<b>CU Extendidos</b>	No existen.
<b>Requisitos funcionales</b>	no	RnF1 RnF2 RnF3
<b>Asuntos pendientes</b>		No existen.

Tabla 10. Descripción del CU Gestionar Solicitud.

<b>Objetivo</b>	Realizar, modificar o eliminar la solicitud de las medias deseadas en el sistema.
<b>Actores</b>	Usuario: (Inicia) Inserta, edita y elimina los datos de una solicitud.
<b>Resumen</b>	El CU inicia cuando el usuario selecciona adicionar, modificar o eliminar una solicitud, para ello introduce los datos necesarios, el CU termina cuando se inserte, modifique o elimine una solicitud.
<b>Complejidad</b>	Media
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	
<b>Poscondiciones</b>	Se adicionó una solicitud al sistema. Se modificaron los datos de una solicitud. Se eliminó una solicitud.
<b>Flujo de eventos</b>	

Flujo básico: Gestionar solicitud		
	Actor	Sistema
1	Selecciona una de las opciones siguientes: <ul style="list-style-type: none"> <li>- Adicionar Solicitud.</li> <li>- Modificar Solicitud. Ver sección1 “Modificar Solicitud”.</li> <li>- Eliminar Solicitud. Ver sección2 “Eliminar Solicitud”.</li> </ul>	
2		Muestra una interfaz con los datos necesarios para realizar la solicitud.
3	Inserta los datos que se solicitan y presiona la opción Aceptar.	
4		Valida que los datos estén correctos.
5		Inserta los datos en la base de datos.
6		Muestra un mensaje diciendo que la solicitud fue insertada correctamente. Termina el CU.
Flujos alternos		
4aEvento: Datos incorrectos		
	Actor	Sistema
4a		Muestra un mensaje de error indicando que faltan datos o hay error en los mismos. Regresa al paso 3 del flujo básico.
Sección1: “Modificar solicitud”		
Flujo básico: Modificar Solicitud		
	Actor	Sistema
1	Selecciona la solicitud que desea modificar.	
2		Muestra una interfaz con los datos de la solicitud que desea modificar.
3	Cambia alguno de los atributos y presiona el botón Aceptar.	
4		Valida que los datos insertados estén correctos.
5		Busca en la base de datos las solicitudes de acuerdo a su identificador y modifica los datos que fueron cambiados.
6		Muestra un mensaje indicando que la solicitud

		fue modificada correctamente. Termina el CU
<b>Flujos alternos</b>		
<b>4aEvento: Datos Incorrectos</b>		
	<b>Actor</b>	<b>Sistema</b>
4a		Muestra un mensaje de error indicando que faltan datos o hay error en los mismos. Regresa al paso 4 del flujo básico Modificar Solicitud.
<b>Sección2: "Eliminar solicitud"</b>		
<b>Flujo básico: Eliminar solicitud</b>		
	<b>Actor</b>	<b>Sistema</b>
1	Selecciona la solicitud que desea eliminar. Y presiona el botón "Eliminar".	
2		Muestra un mensaje de confirmación para saber si desea o no eliminar la solicitud.
3	Selecciona la opción "Aceptar".	
4		Elimina la solicitud. Termina el CU.
<b>Relaciones</b>	<b>CU Incluidos</b>	No existen.
	<b>CU Extendidos</b>	Descargar Material
<b>Requisitos funcionales</b>	<b>no</b> RnF1 RnF2 RnF3	
<b>Asuntos pendientes</b>	<b>No existen.</b>	

Tabla 11. Descripción del CU Atender Solicitud.

<b>Objetivo</b>	Dar respuesta a las solicitudes hechas por el usuario básico.
<b>Actores</b>	Atención al público:(Inicia) Buscar las solicitudes hechas por el usuario.
<b>Resumen</b>	El CU inicia cuando el usuario de atención al público realiza una búsqueda de las solicitudes hechas para darle atención a las mismas. El CU termina cuando sean atendidas las solicitudes.
<b>Complejidad</b>	Alta

<b>Prioridad</b>	Media	
<b>Precondiciones</b>	El usuario esta autenticado y tiene permisos para atender las solicitudes.	
<b>Poscondiciones</b>	Solicitud Atendida.	
<b>Flujo de eventos</b>		
<b>Flujo básico: Atender Solicitud</b>		
	<b>Actor</b>	<b>Sistema</b>
1	Introduce los datos para buscar la solicitud deseada teniendo en cuenta los siguientes parámetros:	
2		Valida que los datos sean correctos.
3		Muestra la lista de solicitudes encontradas de acuerdo a los parámetros establecidos, indicando el estado en que se encuentran (codificando, lista). En caso de estar listas permite la opción de descargar el material. Termina el CU.
<b>Flujos alternos</b>		
<b>2aEvento: Datos Incorrectos</b>		
	<b>Actor</b>	<b>Sistema</b>
2a		Muestra un mensaje de error indicando que faltan datos o hay error en los mismos. Regresa al paso 1 del flujo básico Atender Solicitud.
<b>Relaciones</b>	<b>CU Incluidos</b>	No existen.
	<b>CU Extendidos</b>	Descargar Material
<b>Requisitos funcionales</b>	<b>no</b>	RnF1 RnF2 RnF3
<b>Asuntos pendientes</b>	No existen.	

Tabla 12. Descripción del CU Descargar Material.

<b>Objetivo</b>	Descargar los materiales que el usuario solicitó.
<b>Actores</b>	Usuario, Atención al público:(Inicia) Descarga los materiales solicitados.

<b>Resumen</b>	El CU inicia cuando el usuario atiende o gestiona las solicitudes y desea descargar el material. El CU termina cuando se descarguen las medias deseadas.	
<b>Complejidad</b>	Alta	
<b>Prioridad</b>	Media	
<b>Precondiciones</b>	El estado de la media debe ser Listo.	
<b>Poscondiciones</b>	Descarga el material.	
<b>Flujo de eventos</b>		
<b>Flujo básico: Descargar Material</b>		
	<b>Actor</b>	<b>Sistema</b>
1	Selecciona la opción "Descargar"	
2		La media se descarga en la sdcard del móvil.
<b>Relaciones</b>	<b>CU Incluidos</b>	No existen.
	<b>CU Extendidos</b>	No existen.
<b>Requisitos funcionales</b>	<b>no</b>	RnF1 RnF2 RnF3
<b>Asuntos pendientes</b>	No existen.	

## 2. Casos de prueba

### 2.1. Caso de prueba *Buscar Material*

Tabla 13. Secciones de prueba para el CU *Buscar Material*.

Nombre de Sección	Descripción
Sección 1: Búsqueda básica	Se inserta en la base de datos del sistema Agorav una solicitud donde quedan recogidos los datos del cliente y de la media o subclip deseados.
Sección 2: Búsqueda avanzada	Se elimina una solicitud hecha por un usuario

Sección 3: Mostrar resultados	Se modifica una solicitud hecha por un usuario
-------------------------------	--

Tabla 14. Descripción de las variables correspondientes al caso de prueba para el CU Buscar Material.

No	Nombre del campo	Clasificación	Valor nulo	Descripción
V1	Disponible	Checkbox	si	Si está marcado su valor es 1 si no es 0.
V2	Catalogado	Checkbox	si	Si está marcado su valor es 1 si no es 0.
V3	Duración de la media	Campo de texto	si	Campo que solo admite caracteres numéricos, si son más de dos números deben estar divididos por dos puntos (:).
V4	Nombre físico de la media	Campo de texto	si	Campo que admite caracteres de cualquier tipo.
V5	Título de la media	Campo de texto	si	Campo que admite caracteres de cualquier tipo.
V6	Fecha de inicio	Campo de texto	si	Campo que admite caracteres de tipo numérico con el formato (mm/dd/yyyy).
V7	Fecha de fin	Campo de texto	si	Campo que admite caracteres de tipo numérico con el formato (mm/dd/yyyy).

<b>V8</b>	Texto	Campo de texto	si	Campo que admite caracteres de cualquier tipo.
<b>V9</b>	Ámbito	Selección	si	Se proporcionan cuatro tipos de ámbito de los cuales se debe elegir uno.

Tabla 15. Caso de prueba para el CU Buscar Material: Sección 1: "Búsqueda básica".

Sección 1: "Búsqueda básica"								
Escenario	Descripción	V1	V2	V3	V4	V5	Respuesta del sistema	Flujo central
EC 1.1	Realizar búsqueda correctamente. Se llena al menos un criterio de búsqueda.	V	( )	( )	( )	( )	Se muestra una lista con los archivos multimedia que cumplan con los criterios de búsqueda introducidos por el usuario.	1. Se introducen los criterios de búsqueda. 2. Se presiona el botón buscar.
EC 1.2	Dejar todos los criterios de búsqueda vacíos. Se intenta realizar una búsqueda sin introducir ningún criterio de búsqueda.	( )	( )	( )	( )	( )	Muestra un mensaje indicando que debe introducir al menos un criterio de búsqueda.	1. Se introducen los criterios de búsqueda. 2. Se presiona el botón buscar. 3. Se muestra un mensaje de error.
EC 1.3	Insertar datos inválidos. Se intenta realizar una búsqueda introduciendo datos inválidos.	V	V	I	V	V	Muestra un mensaje indicando en que ha introducido datos inválidos en el campo duración.	1. Se introducen los criterios de búsqueda. 2. Se presiona el botón buscar. 3. Se muestra un mensaje de error.

Tabla 16. Caso de prueba para el CU Buscar Material: Sección 1: "Búsqueda avanzada".

Sección 1: "Búsqueda avanzada"								
Escenario	Descripción	V6	V7	V8	V9	Respuesta del sistema	Flujo central	
EC 1.1	Realizar búsqueda correctamente. Se llena al menos un criterio de búsqueda.	V	( )	( )	( )	Se muestra una lista con los archivos multimedia que cumplan con los criterios de búsqueda introducidos por el	1. Se introducen los criterios de búsqueda. 2. Se presiona el botón buscar.	

.						usuario.	
<b>EC 1.2</b> Dejar todos los criterios de búsqueda vacíos	Se intenta realizar una búsqueda sin introducir ningún criterio de búsqueda.	( )	( )	( )	( )	Muestra un mensaje indicando que debe introducir al menos un criterio de búsqueda.	<ol style="list-style-type: none"> <li>1. Se introducen los criterios de búsqueda.</li> <li>2. Se presiona el botón buscar.</li> <li>3. Se muestra un mensaje de error.</li> </ol>
<b>EC 1.3</b> Insertar datos inválidos	Se intenta realizar una búsqueda introduciendo datos inválidos.	V	V	I	V	Muestra un mensaje indicando en que ha introducido datos inválidos.	<ol style="list-style-type: none"> <li>1. Se introducen los criterios de búsqueda.</li> <li>2. Se presiona el botón buscar.</li> <li>3. Se muestra un mensaje de error.</li> </ol>

Tabla 17. Caso de prueba para el CU Buscar Material: Sección 1: "Búsqueda avanzada".

Sección 1: "Mostrar resultados"			
Escenario	Descripción	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Mostrar resultados.	Se muestran en una lista, los videos que coincidan con los criterios de búsqueda introducidos por el usuario.	Se muestra una lista con los archivos multimedia que cumplan con los criterios de búsqueda introducidos por el usuario.	<ol style="list-style-type: none"> <li>1. Se introducen los criterios de búsqueda en una de las búsquedas de la aplicación.</li> <li>2. Se presiona el botón buscar.</li> </ol>

**2.2. Caso de prueba Reproducir Material**

*Tabla 18. Secciones de prueba para el CU Reproducir Material.*

Nombre de Sección	Descripción
Sección 1: Reproducir Material	Se reproduce un material seleccionado de la lista de videos mostrados en los resultados de las búsquedas.

*Tabla 19. Descripción de las variables correspondientes al caso de prueba para el CU Reproducir Material.*

No	Nombre del campo	Clasificación	Valor nulo	Descripción
V1	Reproducir/Pausar	Botón		Botón Reproducir/Pausar
V2	Detener	Botón		Botón Detener
V3	Marca de inicio	Botón		Botón Marca de inicio
V4	Marca de fin	Botón		Botón Marca de fin
V5	Ir a marca de inicio	Botón		Botón Ir a marca de inicio
V6	Ir a marca de fin	Botón		Botón Ir a marca de fin
V7	Reproducir entre marcas	Botón		Botón Reproducir entre marcas

<b>V8</b>	Insertar subclip	Botón		Botón Insertar subclip
<b>V9</b>	Aceptar	Botón		Botón Aceptar

Tabla 20. Caso de prueba para el CU Reproducir Material: Sección 1: “Reproducir Material”.

<b>Sección 1: “Reproducir Material”</b>				
<b>Escenario</b>	<b>Descripción</b>	<b>Variables</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
<b>EC 1.1</b> Reproducir/ Pausar reproducción	Reproduce/ pausa el video.	<b>V1</b>	El video se reproduce/pausa.	1. Se presiona el botón Reproducir/Pausar del reproductor.
<b>EC 1.2</b> Detener reproducción	Detiene la reproducción del video.	<b>V2</b>	La reproducción del video se detiene y el marcador de progreso se ubica al principio de la barra de progreso.	1. Se presiona el botón Detener del reproductor.
<b>EC 1.3</b> Insertar marca de inicio	Se inserta en la barra de progreso una marca de inicio.	<b>V3</b>	Se inserta en la barra de progreso una marca de inicio mostrándole al usuario el tiempo de la misma.	1. Se presiona el botón Marca de inicio del reproductor.
<b>EC 1.4</b> Insertar marca de fin	Se inserta en la barra de progreso una marca de fin.	<b>V4</b>	Se inserta en la barra de progreso una marca de fin mostrándole al usuario el tiempo de la misma.	1. Se presiona el botón Marca de fin del reproductor.
<b>EC 1.5</b> Ir a marca de	El marcador de progreso se posiciona en la marca de inicio	<b>V5</b>	El marcador de progreso se ubica donde fue creada la marca de inicio en la barra de	1. Se presiona el botón Ir a marca de inicio del reproductor.

inicio	hecha por el usuario.		progreso y comienza a reproducir desde esta.	
<b>EC 1.6</b> Ir a marca de fin	El marcador de progreso se posiciona en la marca de inicio hecha por el usuario.	<b>V6</b>	El marcador de progreso se ubica donde fue creada la marca de fin en la barra de progreso y comienza a reproducir desde esta.	1. Se presiona el botón Ir a marca de fin del reproductor.
<b>EC 1.7</b> Reproducir entre marcas	El marcador de progreso se posiciona en la marca de inicio hecha por el usuario y comienza a reproducir desde esa posición hasta llegar a la marca de fin.	<b>V7</b>	El marcador de progreso se ubica donde fue creada la marca de inicio en la barra de progreso y comienza a reproducir desde esta hasta llegar a la marca de fin y se detiene la reproducción.	1. Se presiona el botón Reproducir entre marcas del reproductor.
<b>EC 1.8</b> Insertar subclip	Inserta en una lista de subclip el subclip seleccionado por el usuario.	<b>V8</b>	Se inserta en una lista el subclip creado por el usuario, conteniendo el nombre de la media y las marcas echas por este. Se inserta además al media completa.	1. Se presiona el botón Insertar subclip del reproductor.
<b>EC 1.9</b> Aceptar	Muestra al usuario una lista con los subclip insertados.	<b>V9</b>	Muestra al usuario una lista con los subclip creados por el usuario, conteniendo también la media completa por si este desea solicitarla.	1. Se presiona el botón Aceptar del reproductor.