

Técnica de Recuperación y Ejecución de Servicios en Redes Superpuestas Peer to Peer No Estructuradas

Anexos

**Gerardo Iván Ordoñez Zambrano
Gonzalo Ernesto Potosí Gutiérrez**

Universidad del Cauca

**Facultad de Ingeniería Electrónica y Telecomunicaciones
Programa en Ingeniería de Sistemas
Departamento de Sistemas
Grupo IDIS - Investigación y Desarrollo en Ingeniería de Software
Línea de Investigación en Tecnología de Soporte
Popayán, febrero de 2015**

Técnica de Recuperación y Ejecución de Servicios en Redes Superpuestas Peer to Peer No Estructuradas



Gerardo Iván Ordoñez Zambrano
Gonzalo Ernesto Potosí Gutiérrez

Director
Ing. Pablo Augusto Magé Imbachí

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones
Programa en Ingeniería de Sistemas
Departamento de Sistemas
Grupo IDIS - Investigación y Desarrollo en Ingeniería de Software
Línea de Investigación en Tecnología de Soporte
Popayán, febrero de 2015

TABLA DE CONTENIDO

ANEXO A: ALCANCE DEL PROYECTO	8
1. Objetivos del Proyecto	8
1.1. Objetivo General	8
1.2. Objetivos Específicos	8
2. Justificación del Proyecto	8
3. Metodología de Trabajo	9
3.1. Fase de análisis y caracterización	9
3.2. Fase de elaboración	9
3.3. Fase de pruebas y análisis de resultados	10
3.4. Fase de documentación y divulgación	10
4. Resumen de Entregables	10
4.1. Relacionados con la Gestión del Trabajo de Grado	10
4.2. Relacionados con el Prototipo Software	10
5. Criterios de éxito del Proyecto	10
ANEXO B: DOCUMENTO DE ESPECIFICACIÓN DE REQUERIMIENTOS SOFTWARE - SRS	11
1. Introducción	11
1.1. Propósito	11
1.2. Alcance	11
1.3. Definiciones, Siglas y Abreviaciones	11
2. Descripción Global	12
2.1. Perspectiva del Producto	12
2.1.1. Interfaces con el sistema	12
2.1.2. Interfaces con el usuario	12
2.1.3. Interfaces con el hardware	12
2.1.4. Interfaces con el software	12
2.1.5. Interfaces de comunicaciones	12
2.1.6. Restricciones de memoria	13
2.1.7. Requerimientos de adaptación	13

2.2. Funciones del Producto	13
2.3. Características del Usuario	13
2.4. Atenciones y Dependencias	14
2.5. Priorizar los Requerimientos	14
3. Requerimientos Específicos.....	14
3.1. Funcionales	14
3.2. No Funcionales	19
4. Rendimiento	20
5. Restricciones de Diseño	20
ANEXO C: TECNOLOGÍAS PARA LA IMPLEMENTACIÓN DE REDES SUPERPUSTAS PEER-TO-PEER Y LA DIFUSIÓN DEL SERVICIO DE MEDIA STREAMING.....	21
1. Plataforma JXTA.....	21
1.1 Protocolos JXTA	21
1.2. Servicios JXTA	23
ANEXO D: ESPECIFICACIÓN DE LOS EQUIPOS UTILIZADOS EN LAS PRUEBAS.....	24
ANEXO E: MANUAL TÉCNICO Y DE INSTALACIÓN.....	28
1. Requerimientos Software de la Aplicación.....	28
1.1. Instalación de Java	28
1.2. Lista de excepciones Java.....	28
1.3. Instalación VLC.....	28
2. Instalación de la Aplicación.....	29
3. Ejecución de la Aplicación	29
3.1. Cambiar el puerto de JXTA.....	29
ANEXO F: MANUAL DE USUARIO	31
1. Ejecutar la Aplicación.....	31
2. Sección de mensajes	32
3. Salir de la Aplicación.....	33
ANEXO G: DIAGRAMAS DE CLASES	34

1. Diagrama de clases del sistema	34
2. Diagrama de clases nodo cliente	35
3. Diagrama de clases nodo servidor.....	36
ANEXO H: PRUEBAS	37
3. Pruebas manuales	37
REFERENCIAS BIBLIOGRÁFICAS	41

LISTA DE FIGURAS

Figura 1. Protocolos JXTA.....	23
Figura 2. Estructura de directorios	29
Figura 3. Interfaz prototipo cambio de puerto.....	30
Figura 4. Interfaz prototipo	31
Figura 5. Sección de mensajes	32
Figura 6. Cerrar aplicación	33
Figura 7. Diagrama de clases.....	34
Figura 8. Diagrama de clases nodo cliente	35
Figura 9. Diagrama clases nodo servidor.....	36

LISTA DE TABLAS

Tabla 1. Requerimiento - Publicar Servicio de Media Streaming	15
Tabla 2. Requerimiento – Buscar Servicio de Media Streaming	15
Tabla 3. Requerimiento – Monitorear Condición Servicio	15
Tabla 4. Requerimiento - Recuperar el Servicio de Media Streaming.....	16
Tabla 5. Requerimiento - Crear Mensaje Descriptor Para el Servicio	16
Tabla 6. Requerimiento - Obtener Mensaje Descriptor Del Servicio	17
Tabla 7. Requerimiento - Consumir Servicio de Media Streaming.....	17
Tabla 8. Requerimiento - Iniciar el Nodo Servidor.....	17
Tabla 9. Requerimiento - Detener el Nodo Servidor.....	18
Tabla 10. Requerimiento - Iniciar el Nodo Cliente	18
Tabla 11. Requerimiento - Detener el Nodo Cliente.....	19
Tabla 12. Equipo 1	24
Tabla 13. Equipo 2	24
Tabla 14. Equipo 3	24
Tabla 15. Equipo 4	25
Tabla 16. Equipo 5	25
Tabla 17. Equipo 6	25
Tabla 18. Equipo 7	26
Tabla 19. Equipo 8	26
Tabla 20. Equipo 9	26
Tabla 21. Equipo 10	27
Tabla 22. Equipo 11	27
Tabla 23. Equipo 12	27
Tabla 24. Escenarios de pruebas.....	40
Tabla 25. Formato registro de resultados.....	40

ANEXO A: ALCANCE DEL PROYECTO

1. Objetivos del Proyecto

1.1. Objetivo General

- Definir una técnica de recuperación y ejecución de un servicio dentro de un entorno de redes superpuestas peer-to-peer no estructuradas.

1.2. Objetivos Específicos

- Caracterizar¹ una técnica de recuperación y ejecución de servicios para redes superpuestas peer-to-peer no estructuradas.
- Construir o adecuar una técnica de recuperación y ejecución de servicios para redes superpuestas peer-to-peer no estructuradas, basándose en las características de recuperación de servicios existentes.
- Diseñar e implementar un prototipo software que permita evaluar² la técnica propuesta, teniendo en cuenta las características de recuperación seleccionadas.

2. Justificación del Proyecto

Este proyecto de investigación pretende contribuir en el problema de la disponibilidad de prestación de servicios en las redes peer-to-peer no estructuradas. Dada la necesidad de establecer una forma de prestar servicios donde se aprovechen los dispositivos ubicados en la red, y puedan relacionarse entre sí, buscando, localizando y recuperando contenidos y servicios que son ofrecidos en las redes superpuestas peer-to-peer no estructuradas. Por lo tanto, este proyecto trata de hacer una contribución significativa a nivel investigativo en el área de las redes peer-to-peer, con la especificación o adecuación de una técnica de recuperación y ejecución de servicios en redes superpuestas peer-to-peer no estructuradas. Entre los aportes se encuentran:

- Construir o adecuar una técnica de recuperación y ejecución de servicios.
- Establecer las principales características de recuperación y ejecución de servicios en este tipo de redes.

¹ Determinar los atributos peculiares de alguien o de algo, de modo que claramente se distinga de los demás.

² En la caracterización se define el perfil del servicio sobre el cual se hará la evaluación del prototipo

- Realizar la recuperación y ejecución eficiente de un servicio en los nodos presentes en entornos de redes peer-to-peer no estructuradas.

Partiendo del hecho de que los servicios están considerados el núcleo de la mayoría de procesos en la actualidad en cuanto a nivel de redes se refiere, se pretende ayudar en el proceso de recuperación y ejecución de los mismos y a su vez, explotar las diferentes funcionalidades prestadas por las plataformas de desarrollo en redes peer-to-peer.

3. Metodología de Trabajo

Este proyecto será llevado bajo los siguientes enfoques metodológicos: a) Metodología de investigación apoyada sobre los métodos de investigación de ingeniería, y se verá reflejada en la primera fase del proyecto. b) Un subconjunto de prácticas de la metodología de eXtreme Programming XP (Programación Extrema) [1], el cual será usado tanto para la gestión del proyecto como para el desarrollo. Con relación a la metodología XP será adaptada al proyecto donde el análisis de requisitos no se puede hacer de forma directa, los cambios son constantes y el equipo de desarrollo es pequeño, como en este caso en particular. El proyecto será dividido en cuatro fases: Fase de análisis y caracterización, fase de elaboración, fase de pruebas y análisis de resultados, y una fase de documentación y divulgación.

3.1. Fase de análisis y caracterización

En esta fase se revisará documentación correspondiente con el objetivo de definir una base conceptual, el estado del arte y una caracterización; para ello se procederá a la consulta e investigación documental por medio de artículos, tesis, consultas a expertos, u otros medios de información relacionados a los temas redes superpuestas peer-to-peer no estructuradas, recuperación y ejecución de servicios.

3.2. Fase de elaboración

Durante esta fase se realizará la adecuación o construcción de una técnica de recuperación de servicios basados en la caracterización de la fase anterior; de igual forma se procederá al diseño y la implementación de un prototipo software de la técnica planteada. Como se mencionó anteriormente, para la implementación del prototipo software se usará la metodología de desarrollo de software eXtreme Programming XP, debido a que, dadas las condiciones particulares del proyecto, existen ciertas características de XP que pueden ser aplicadas directamente. Sin embargo puesto que XP cuenta con prácticas como historias de usuario y juego de planificación, solo se considerarán las fases y artefactos necesarios.

3.3. Fase de pruebas y análisis de resultados

En esta fase se realizarán pruebas del prototipo desarrollado para verificar la técnica propuesta. Este tipo de pruebas se diferencian de aquellas que tienen el objetivo de encontrar errores de programación. Se realizarán pruebas en la red de área local LAN y otros ambientes que se consideren necesarios. Las pruebas en LAN se realizarán en una sala de cómputo y permitirá tener un ambiente plenamente controlado en todos los clientes.

3.4. Fase de documentación y divulgación

En esta fase se llevarán a cabo un conjunto de actividades orientadas a la documentación del desarrollo del proyecto de investigación; estas actividades se harán en forma paralela a las fases anteriormente mencionadas. La divulgación se realizará mediante la elaboración de un artículo, la monografía y la sustentación del trabajo de grado ante los jurados asignados en la facultad de ingeniería electrónica y telecomunicaciones (F.I.E.T) de la Universidad del Cauca.

4. Resumen de Entregables

4.1. Relacionados con la Gestión del Trabajo de Grado

- Monografía del Trabajo de Grado
- Artículo de Investigación

4.2. Relacionados con el Prototipo Software

- Definición de la arquitectura del sistema
- Diagrama de clases
- Manual de instalación
- Manual de usuario
- Prototipo software final que cumpla con los requerimientos establecidos

5. Criterios de éxito del Proyecto

El proyecto será exitoso si como resultado genera:

- Los entregables relacionados con la gestión del trabajo de grado.
- Los entregables relacionados con el prototipo software.

ANEXO B: DOCUMENTO DE ESPECIFICACIÓN DE REQUERIMIENTOS SOFTWARE - SRS

1. Introducción

1.1. Propósito

En este documento se pretende clasificar y describir los requerimientos que han sido identificados para el desarrollo del prototipo software que verificará la técnica de recuperación y ejecución de servicios.

1.2. Alcance

El prototipo software, que será desarrollado, pretende verificar la técnica de recuperación y ejecución de servicios propuesta utilizando un servicio de **Media Streaming**. Por lo tanto, la mayoría de la funcionalidad del prototipo software se ejecutará de manera automática, con reducida interacción con el usuario final del sistema. El prototipo software permitirá ejecutar la funcionalidad tanto de cliente (solicitante del servicio) como de servidor (prestador del servicio) en una sola interfaz de usuario.

El prototipo software integrará dos tecnologías para el desarrollo de redes peer-to-peer y el sistema de media streaming, la primera, JXTA, para el desarrollo de Redes peer-to-peer; y la segunda, VLC (VLCj), para el desarrollo de sistemas de Media Streaming.

1.3. Definiciones, Siglas y Abreviaciones

- **Servicio de Media Streaming:** Es un servicio que realiza distribución digital de multimedia a través de una red de computadoras de manera que el usuario visualiza el video o audio en paralelo mientras se descarga.
- **GUI:** Interfaz Gráfica de Usuario.
- **JDK:** Java Development Kit.
- **JRE:** Java Runtime Environment.
- **VLCj:** Framework Java para el Media Player VLC
- **JXTA:** Framework Desarrollo Peer-to-Peer

2. Descripción Global

2.1. Perspectiva del Producto

2.1.1. Interfaces con el sistema

El prototipo software deberá integrarse con JXTA y VLCj.

2.1.2. Interfaces con el usuario

- El usuario interactuará con el prototipo software a través de una aplicación de escritorio o web.
- El prototipo software tendrá la funcionalidad de cliente, de servidor y de mensajes de depuración en la misma interfaz gráfica de usuario.
- La Interfaz gráfica de usuario deberá ser simple e intuitiva, permitiendo al usuario un fácil manejo de la aplicación.
- Los mensajes de información, depuración y mensajes de error serán mostrados en el panel de notificaciones de información.
- Los mensajes de error graves serán desplegados en cajas de diálogos.

2.1.3. Interfaces con el hardware

El prototipo software se podrá ejecutar en computadores personales o de escritorio, que tengan una conexión a Internet.

2.1.4. Interfaces con el software

El prototipo software estará implementado en java, al igual que las tecnologías de desarrollo VLCj y JXTA. Por la tanto, el computador donde se ejecute el prototipo deberá tener instalada una Máquina Virtual Java y las librerías LibVLC (Media Player VLC). Se recomienda la versión 1.7.0_21 del JDK o del JRE de Oracle.

2.1.5. Interfaces de comunicaciones

El prototipo software será soportado por los protocolos de comunicaciones JXTA y su respectiva arquitectura.

2.1.6. Restricciones de memoria

- Se necesitarán como mínimo 512 MB de memoria RAM y un procesador con una velocidad de 1.2 GHz para ejecutar la aplicación.
- Se recomienda una memoria RAM de 2GB o más y un procesador con una velocidad de 2.0 GHz o más para un rendimiento óptimo de la aplicación.

2.1.7. Requerimientos de adaptación

El prototipo software no tendrá en cuenta aspectos de seguridad, tales como nombre de usuarios o contraseñas, o sistemas de autenticación.

2.2. Funciones del Producto

El prototipo software permitirá el despliegue del servicio de media streaming a través de la red peer-to-peer permitiendo conexiones de tipo punto a multipunto en el nivel de aplicación. Teniendo como base una fuente principal de streaming (Servidor de Streaming) ajeno a la red peer-to-peer. Los nodos servidor que ingresen a la red peer-to-peer prestarán el servicio de retransmisión de la fuente principal. Por otra parte, los nodos clientes que ingresen a la red peer-to-peer buscarán el servicio de streaming, y lo consumirán de un determinado nodo servidor elegido aleatoriamente. En caso de fallas de red o de desconexión en el nodo servidor, el nodo cliente estará en la capacidad de recuperar el servicio de streaming con otro nodo servidor.

Además, el prototipo permitirá ejecutar la funcionalidad de cliente y servidor en una sola interfaz de usuario y mostrará mensajes de información y depuración del proceso que se está llevando a cabo.

2.3. Características del Usuario

La mayoría de la funcionalidad del prototipo software se ejecutará de manera automática, con reducida interacción con el usuario final del sistema. La única interacción del usuario final con el sistema, será la selección manual del tipo de funcionamiento del sistema, es decir el sistema puede comportarse como cliente, o como servidor.

2.4. Atenciones y Dependencias

Existen algunos factores que pueden afectar el desarrollo del prototipo, entre ellos se mencionan:

- Poco conocimiento de las tecnologías que se van a utilizar en el desarrollo del prototipo.
- Actividades no programadas, como por ejemplo capacitación y manejo de las tecnologías de desarrollo utilizadas.
- Cambios en las prioridades del equipo de trabajo.
- Resistencia a los cambios planteados.
- Fechas límites de entrega.

2.5. Priorizar los Requerimientos

- **Prioridad 1:** Por medio de la plataforma JXTA, desarrollar la funcionalidad del sistema y de las comunicaciones para la red peer-to-peer del prototipo software, e integrar la tecnología VLC a manera de servicio de media streaming en los nodos servidor.
- **Prioridad 2:** En el sistema peer-to-peer implementar las técnicas de búsqueda, recuperación y ejecución del servicio de media streaming.

3. Requerimientos Específicos

3.1. Funcionales

REQUERIMIENTO: PUBLICAR EL SERVICIO DE MEDIA STREAMING	
Código	R-1
Nombre	Publicar el Servicio de Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	X Alta/Esencial Media/Deseado Baja/Opcional
Descripción	El sistema debe crear y publicar el servicio de media streaming.
Prerrequisito	Fuente principal de streaming disponible (Servidor de Streaming ajeno a la red peer-to-peer). Se debe haber creado un mensaje descriptor del servicio para integrarlo en la publicación el servicio de media streaming.
Manejo de	E1: El servicio de streaming no se puede publicar. El nodo

Errores	servidor queda a la espera del levantamiento del servicio de streaming en la fuente principal. El sistema muestra un mensaje de notificación.
----------------	---

Tabla 1. Requerimiento - Publicar Servicio de Media Streaming

REQUERIMIENTO: BUSCAR EL SERVICIO DE MEDIA STREAMING	
Código	R-2
Nombre	Buscar el Servicio de Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	X Alta/Esencial Media/Deseado Baja/Opcional
Descripción	El sistema debe buscar el servicio de media streaming a través de la consulta sobre las publicaciones de servicio de los nodos servidor.
Prerrequisito	Ninguno
Manejo de Errores	E1: El servicio de media streaming no se encuentra. El sistema muestra un mensaje de notificación y sigue buscando el servicio indefinidamente.

Tabla 2. Requerimiento – Buscar Servicio de Media Streaming

REQUERIMIENTO: MONITOREAR CONDICIÓN SERVICIO DE MEDIA STREAMING	
Código	R-3
Nombre	Monitorear Condición Servicio de Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	X Alta/Esencial Media/Deseado Baja/Opcional
Descripción	El sistema debe permitir el monitoreo de la condición del servicio de media streaming que un nodo cliente se encuentra consumiendo. En caso de detectar desconexión o pérdida del servicio se debe disparar inmediatamente el mecanismo de la recuperación. Esta funcionalidad pertenece a un nodo cliente.
Prerrequisito	Consumo actual del servicio de media streaming por parte de un nodo cliente.
Manejo de Errores	

Tabla 3. Requerimiento – Monitorear Condición Servicio

REQUERIMIENTO: RECUPERAR EL SERVICIO DE MEDIA STREAMING	
Código	R-4

Nombre	Recuperar el Servicio de Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	X Alta/Esencial Media/Deseado Baja/Opcional
Descripción	El sistema debe recuperar el servicio de media streaming. Un nodo cliente del sistema realiza búsqueda en el cache local y/o búsqueda remota con el objetivo de encontrar nuevos servidores disponibles y recuperar el servicio.
Prerrequisito	No se encuentra ningún nodo servidor que presta el servicio de media streaming.
Manejo de Errores	E1: El servicio de media streaming no se encuentra. El sistema muestra un mensaje de notificación y sigue buscando el servicio indefinidamente, hasta poder recuperarlo. Un peer cliente elimina de su lista local (caché) los nodos servidor que se encuentran disponibles.

Tabla 4. Requerimiento - Recuperar el Servicio de Media Streaming

REQUERIMIENTO: CREAR MENSAJE DESCRIPTOR PARA EL SERVICIO DE MEDIA STREAMING	
Código	R-5
Nombre	Crear Mensaje Descriptor Servicio Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	X Alta/Esencial Media/Deseado Baja/Opcional
Descripción	El sistema debe crear un mensaje descriptor del servicio de media streaming a integrar en la publicación del servicio. Esta funcionalidad pertenece a un nodo servidor.
Prerrequisito	La fuente principal de streaming debe estar disponible (Servidor de Streaming ajeno a la red peer-to-peer)..
Manejo de Errores	E1: La fuente principal de streaming no está disponible. El sistema muestra un mensaje de notificación y espera hasta el restablecimiento de la fuente principal.

Tabla 5. Requerimiento - Crear Mensaje Descriptor Para el Servicio

REQUERIMIENTO: OBTENER MENSAJE DESCRIPTOR DEL SERVICIO DE MEDIA STREAMING	
Código	R-6
Nombre	Obtener Mensaje Descriptor Servicio Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	X Alta/Esencial Media/Deseado Baja/Opcional

Descripción	El sistema debe consultar una publicación de servicio de un nodo servidor para poder obtener el descriptor del servicio. Esta funcionalidad pertenece a un nodo cliente.
Prerrequisito	Publicaciones de servicio por parte de nodos servidor.
Manejo de Errores	

Tabla 6. Requerimiento - Obtener Mensaje Descriptor Del Servicio

REQUERIMIENTO: CONSUMIR SERVICIO DE MEDIA STREAMING	
Código	R-7
Nombre	Consumir Servicio Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	X Alta/Esencial Media/Deseado Baja/Opcional
Descripción	El sistema debe permitirle a través del descriptor del servicio iniciar el consumo del servicio de media streaming que presta un nodo servidor. Esta funcionalidad pertenece a un nodo cliente.
Prerrequisito	Publicaciones de servicio por parte de nodos servidor.
Manejo de Errores	

Tabla 7. Requerimiento - Consumir Servicio de Media Streaming

REQUERIMIENTO: INICIAR EL NODO SERVIDOR DEL SERVICIO DE MEDIA STREAMING	
Código	R-8
Nombre	Iniciar el Nodo Servidor del Servicio de Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	Alta/Esencial X Media/Deseado Baja/Opcional
Descripción	El sistema debe permitir al usuario iniciar el nodo servidor.
Prerrequisito	El nodo servidor debe estar detenido.
Manejo de Errores	E1: El nodo servidor no se puede iniciar. El sistema muestra un mensaje de error.

Tabla 8. Requerimiento - Iniciar el Nodo Servidor

REQUERIMIENTO: DETENER EL NODO SERVIDOR DEL SERVICIO DE MEDIA STREAMING	
Código	R-9
Nombre	Detener el Nodo Servidor del Servicio de Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	Alta/Esencial X Media/Deseado Baja/Opcional
Descripción	El sistema debe permitir al usuario detener el nodo servidor.
Prerrequisito	El nodo servidor debe estar funcionando.
Manejo de Errores	E1: El nodo servidor no se puede detener. El sistema muestra un mensaje de error.

Tabla 9. Requerimiento - Detener el Nodo Servidor

REQUERIMIENTO: INICIAR EL NODO CLIENTE DEL SERVICIO DE MEDIA STREAMING	
Código	R-10
Nombre	Iniciar el Nodo Cliente del Servicio de Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	Alta/Esencial X Media/Deseado Baja/Opcional
Descripción	El sistema debe permitir al usuario iniciar el nodo cliente.
Prerrequisito	El nodo cliente debe estar detenido.
Manejo de Errores	E1: El nodo cliente no se puede iniciar. El sistema muestra un mensaje de error.

Tabla 10. Requerimiento - Iniciar el Nodo Cliente

REQUERIMIENTO: DETENER EL NODO CLIENTE DEL SERVICIO DE MEDIA STREAMING	
Código	R-11
Nombre	Detener el Nodo Cliente del Servicio de Media Streaming
Fuente	Información recolectada en la fase de investigación.
Prioridad	Alta/Esencial X Media/Deseado Baja/Opcional
Descripción	El sistema debe permitir al usuario detener el nodo cliente.
Prerrequisito	El nodo cliente debe estar funcionando.

Manejo de Errores	E1: El nodo cliente no se puede detener. El sistema muestra un mensaje de error.
--------------------------	--

Tabla 11. Requerimiento - Detener el Nodo Cliente

3.2. No Funcionales

- **Fiabilidad:** El prototipo software debe ser tolerante a fallos en especial a fallos de red o desconexión voluntaria o involuntaria por parte del usuario. Debe poder recuperarse de los diferentes errores que puedan presentarse. Es importante decir que los fallos por componentes hardware pueden estar presentes y no se tienen en cuenta en el desarrollo de este prototipo, así como también es importante decir que los fallos de software siempre están latentes en una aplicación y algunas veces pueden ser inesperados.
- **Disponibilidad:** Una de las características de las redes peer-to-peer es que los nodos pueden entrar y salir de ésta arbitrariamente, por lo tanto el grado de disponibilidad de la aplicación depende de la disponibilidad del servicio en la red peer-to-peer.
- **Seguridad:** El prototipo software no tendrá en cuenta aspectos de seguridad, tales como nombre de usuarios o contraseñas, o sistemas de autenticación.
- **Consistencia:** La consistencia es un punto importante en las aplicaciones peer-to-peer, sin ella, el sistema no funcionará de una manera adecuada. Por lo tanto, la aplicación debe garantizar consistencia en el envío y recepción de mensajes, y en actualización de interfaz de usuario.
- **Mantenimiento:** El desarrollo del prototipo software no incluye mantenimiento. Sin embargo, si hay una mejora en las actualizaciones de las herramientas de desarrollo o las plataformas utilizadas, se incluirán esas nuevas actualizaciones, siempre y cuando el prototipo no se vea afectado de una manera negativa.
- **Portabilidad:** El prototipo software se podrá ejecutar en computadores personales o de escritorio, que tengan una conexión a Internet. El prototipo software deberá poder ejecutarse en cualquier sistema operativo a través de un Applet Viewer o un Navegador Web, que tenga instalada una Máquina Virtual Java. Se recomienda la versión 1.7.0_21 del JDK o la v7 del JRE de Oracle.

4. Rendimiento

El prototipo software deberá cumplir con las siguientes características de rendimiento:

- Partiendo de la condición de que siempre se tengan disponibles nodos servidor prestando el servicio de media streaming, en caso de pérdida del servicio los nodos clientes deben estar en la capacidad de recuperar el servicio en tiempos considerablemente bajos, debido a que se está prestando un servicio de media streaming.

5. Restricciones de Diseño

No se ha identificado ninguna restricción de diseño.

ANEXO C: TECNOLOGÍAS PARA LA IMPLEMENTACIÓN DE REDES SUPERPUESTAS PEER-TO-PEER Y LA DIFUSIÓN DEL SERVICIO DE MEDIA STREAMING

1. Plataforma JXTA

1.1 Protocolos JXTA

JXTA define una serie de mensajes XML, también conocidos como protocolos, para la comunicación entre nodos. Los nodos usan estos protocolos para descubrir otros nodos, anunciar y descubrir recursos de red y para la comunicación y el enrutamiento de mensajes [2]

JXTA define seis protocolos estándar [3]

- **Peer Discovery Protocol (PDP):** Usado por los nodos para anunciar sus propios recursos y descubrir recursos de otros nodos. Cada recurso de un nodo es descrito y publicado usando un *advertisement*. Un recurso puede ser un nodo, un grupo, un pipe, un servicio o cualquier otro recurso que se tenga un *advertisement*. Este protocolo permite a un nodo buscar *advertisements* sobre la red. La actual implementación usa una combinación de IP multicast y el uso de súper-nodos mediante los nodos *rendezvous*.
- **Peer Information Protocol (PIP):** Usado por los nodos para obtener información del estado de otros nodos. Este protocolo provee un conjunto de mensajes para obtener la información de estado actual de los nodos en la red. Un mensaje *ping* es enviado a otro nodo para verificar si el nodo está vivo y obtener información acerca de este. El mensaje *ping* especifica que puede ser retransmitida una respuesta completa, mediante un *advertisement*, o una respuesta simple conocimiento, como por ejemplo *live* o *uptime*. Un mensaje *PeerInfo* es usado para enviar un mensaje de respuesta a un mensaje *ping*. El mensaje *PingInfo* contiene la credencial del remitente, el ID del nodo fuente, el ID del nodo destino, el tiempo de vida y el *advertisement* del nodo.
- **Peer Resolver Protocol (PRP):** Permite a los nodos enviar una consulta genérica a uno o más nodos y recibir una o múltiples respuestas a la consulta. Las consultas pueden ir dirigidas a todos los nodos en un grupo o a nodos específicos dentro del grupo. A diferencia de PDP y PIP, los cuales son usados para realizar consultas específicas con información predefinida, este protocolo permite a los nodos y servicios definir e

intercambiar cualquier información arbitraria que necesiten. Este protocolo usa el Servicio Rendezvous para diseminar una consulta a múltiples nodos y usa mensajes unicast para enviar consultas a nodos específicos. Los protocolos PDP y PIP están construidos usando PRP.

- **Pipe Binding Protocol (PBP):** Usado por los nodos para establecer un canal de comunicación virtual, o *pipe*, entre uno o más nodos. Este protocolo es usado por un nodo para establecer dos o más puntos de conexión finales, también llamados *endpoints*. El enlace virtual del pipe puede encapsular cualquier cantidad de enlaces de transporte de redes físicas. Cada punto final del pipe trabaja para mantener el enlace virtual y para establecerlo de nuevo cuando sea necesario, mediante establecimiento o búsqueda de los endpoints de los pipes actuales.
- **Endpoint Routing Protocol (ERP):** Usado por los nodos para buscar rutas hacia puertos de destino en otros nodos. La información de enrutamiento incluye una secuencia ordenada de los IDs de los nodos que pueden ser usados para enviar un mensaje a su destino. Este protocolo permite a los nodos enviar mensajes a otros nodos sin que exista una conexión directa entre ellos. El mensaje es pasado a través de nodos intermediarios para alcanzar su destino final.
- **Rendezvous Protocol (RVP):** Usado por los nodos para encontrar recursos, propagar mensajes dentro de un grupo y anunciar recursos locales. Es usado por los nodos *rendezvous* para organizarse con otros nodos *rendezvous*, compartir el espacio de direcciones de las DHT y propagar mensajes en un entorno controlado. Este protocolo posee mecanismos que permiten que la propagación de mensajes se ejecute de una manera controlada y eficiente.

En [2], se muestra cada uno de los protocolos JXTA y su relación con los otros protocolos [3]

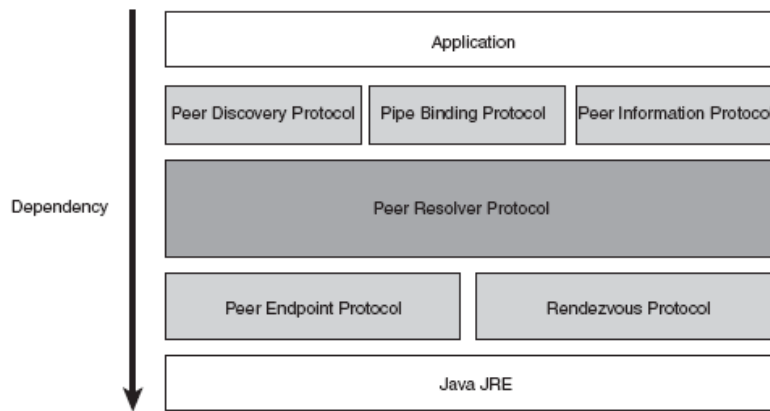


Figura 1. Protocolos JXTA

Todos los protocolos de JXTA son asíncronos y están basados en el modelo consulta/respuesta. Por ejemplo, un nodo JXTA usa uno de los protocolos para enviar una consulta a uno o más nodos dentro del grupo, pero este nodo puede recibir cero, una o más respuestas a la consulta realizada. No es necesario que los nodos JXTA implementen los seis protocolos, solo necesitan implementar los protocolos que van a usar [4].

1.2. Servicios JXTA

Todos los protocolos estándar de JXTA son implementados como servicios, llamados servicios del núcleo o *core services*, los cuales son los siguiente [2]:

- Discovery Service
- Peer Info Service
- Resolver Service
- Pipe Service
- Endpoint Service
- Rendezvous Service

Una instancia de un servicio es asociada con un grupo de nodos específico. Solo los nodos que son miembros del mismo grupo son capaces de comunicarse con otros nodos del grupo por medio de estos servicios. Por defecto, todos los nodos tiene un grupo común llamado *Net Peer Group*, este grupo permite que todos los nodos y sus advertisements sean descubiertos. Los servicios proveen a los desarrolladores un nivel de abstracción, ocultando para ellos algunas cosas que no son necesarias conocer o manipular de los protocolos [2].

ANEXO D: ESPECIFICACIÓN DE LOS EQUIPOS UTILIZADOS EN LAS PRUEBAS

A continuación se muestra la ficha técnica con las respectivas especificaciones de cada equipo usado en las pruebas y la verificación de la técnica de recuperación y ejecución de un servicio de media streaming. Para la fuente de streaming principal se utilizó VLC como servidor de media streaming, usando como recurso un video con códec de video H.264 y MP3 para el códec de audio.

EQUIPO			
SERIAL	PC-01		
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	2 GB
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.00 GHz
SISTEMA OPERATIVO	Windows 7 Professional 32bits		
TARJETA GRAFICA	Integrada Intel		

Tabla 12. Equipo 1

EQUIPO			
SERIAL	PC-02		
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	2 GB
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.4 GHz
SISTEMA OPERATIVO	Windows 7 Professional 32bits		
TARJETA GRAFICA	Integrada Intel		

Tabla 13. Equipo 2

EQUIPO			
SERIAL	PC-03		
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	2 GB
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.4 GHz
SISTEMA OPERATIVO	Windows 7 Professional 32bits		
TARJETA GRAFICA	Integrada Intel		

Tabla 14. Equipo 3

EQUIPO			
SERIAL	PC-04		
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	2 GB
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.4 GHz
SISTEMA OPERATIVO	Windows 7 Professional 32bits		
TARJETA GRAFICA	Integrada Intel		

Tabla 15. Equipo 4

EQUIPO			
SERIAL	PC-05		
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	2 GB
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.00 GHz
SISTEMA OPERATIVO	Windows 7 Professional 32bits		
TARJETA GRAFICA	Integrada Intel		

Tabla 16. Equipo 5

EQUIPO			
SERIAL	PC-06		
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	2 GB
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.00 GHz
SISTEMA OPERATIVO	Windows 7 Professional 32bits		
TARJETA GRAFICA	Integrada Intel		

Tabla 17. Equipo 6

EQUIPO				
SERIAL	PC-07			
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento	
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	4 GB	
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.4 GHz	
SISTEMA OPERATIVO	Windows 7 Professional 32bits			
TARJETA GRAFICA	Integrada Intel			

Tabla 18. Equipo 7

EQUIPO				
SERIAL	PC-08			
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento	
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	4 GB	
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.4 GHz	
SISTEMA OPERATIVO	Windows 7 Professional 32bits			
TARJETA GRAFICA	Integrada Intel			

Tabla 19. Equipo 8

EQUIPO				
SERIAL	PC-09			
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento	
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	2 GB	
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.00 GHz	
SISTEMA OPERATIVO	Windows 7 Professional 32bits			
TARJETA GRAFICA	Integrada Intel			

Tabla 20. Equipo 9

EQUIPO				
SERIAL	PC-10			
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento	
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	2 GB	
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.00 GHz	
SISTEMA OPERATIVO	Windows 7 Professional 32bits			
TARJETA GRAFICA	Integrada Intel			

Tabla 21. Equipo 10

EQUIPO				
SERIAL	PC-11			
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento	
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	4 GB	
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	2.4 GHz	
SISTEMA OPERATIVO	Windows 7 Professional 32bits			
TARJETA GRAFICA	Integrada Intel			

Tabla 22. Equipo 11

EQUIPO				
SERIAL	PC-12			
ESTADO ACTUAL	En Funcionamiento	SI	Sin Funcionamiento	
MEMORIA RAM	DDR-SDRAM	CAPACIDAD	1 GB	
TIPO PROCESADOR	Intel (R) Pentium (R) Dual CPU	VELOCIDAD	1.8 GHz	
SISTEMA OPERATIVO	Windows XP Professional 32bits			
TARJETA GRAFICA	Integrada Intel			

Tabla 23. Equipo 12

ANEXO E: MANUAL TÉCNICO Y DE INSTALACIÓN

1. Requerimientos Software de la Aplicación

1.1. Instalación de Java

Es necesario instalar Java en el equipo en el cual se va ejecutar la aplicación. Solo es necesario instalar el Entorno de Ejecución Java (JRE), pero si se desea puede instalarse el Java Development Kit (JDK). Se recomienda la versión de Java 8 Update 25.

En [4] se puede encontrar una guía detallada para la instalación del JDK y JRE 8 para distintos sistemas operativos.

1.2. Lista de excepciones Java

Para activar y poder ejecutar aplicaciones Java se debe configurar el apartado de seguridad en el panel de control de Java. Para esto, abra el '*Panel de control*', y en la sección '*Java*' diríjase al apartado de seguridad.

- Para versiones anteriores a Java 8, coloque el nivel de seguridad más bajo.
- Para versiones iguales o superiores a Java 8, agregue en la sección '*Editar lista de sitios*' la dirección URL que aparece en el navegador al iniciar la aplicación.

1.3. Instalación VLC

Es necesario instalar el paquete de VLC Media Player en el equipo que se ejecutará la aplicación, ya que contiene la librería núcleo libVLC para interactuar con las funcionalidades de media streaming. Sólo es necesario instalar el software VLC Media Player en su versión 2.0.10 o superiores.

En [5] se puede encontrar una guía de instalación rápida del VLC Media Player para distintos sistemas operativos.

2. Instalación de la Aplicación

La aplicación no necesita instalarse. Sólo necesita copiarse en un disco que permita la lectura y escritura de archivos, debido a que, tanto JXTA y VLC, crean archivos temporales en el directorio de ejecución de la aplicación.

La estructura del directorio quedaría así:

```
[AppletPeer]
- [lib]
  - jna-3.5.2.jar
  - jxta.jar
  - platform-3.5.2.jar
  - vlcj-2.4.1.jar
- AppletPeer.jar
- launch.html
- launch.jnlp
```

Figura 2. Estructura de directorios

3. Ejecución de la Aplicación

Después de copiar el directorio con todos los archivos a un disco que permita la lectura y escritura, abrir en un navegador web el archivo `launch.html`, para iniciar la aplicación.

3.1. Cambiar el puerto de JXTA

Por defecto JXTA se ejecuta en el puerto 9702. Si desea cambiar el puerto en el cual se ejecuta JXTA.

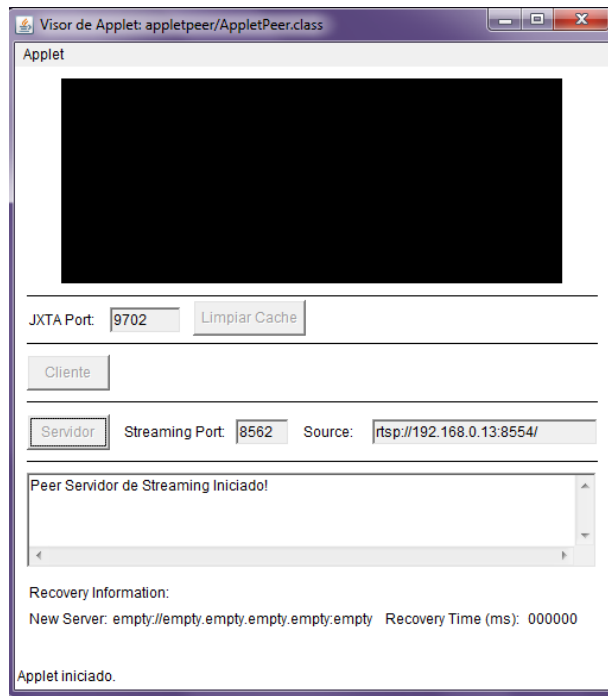


Figura 3. Interfaz prototipo cambio de puerto

ANEXO F: MANUAL DE USUARIO

1. Ejecutar la Aplicación

Ejecute el archivo `launch.html`, para iniciar la aplicación.

Si la aplicación no se ejecuta automáticamente, siga los pasos mencionados en el manual técnico y de instalación. En caso de ejecución exitosa, aparecerá la siguiente ventana:

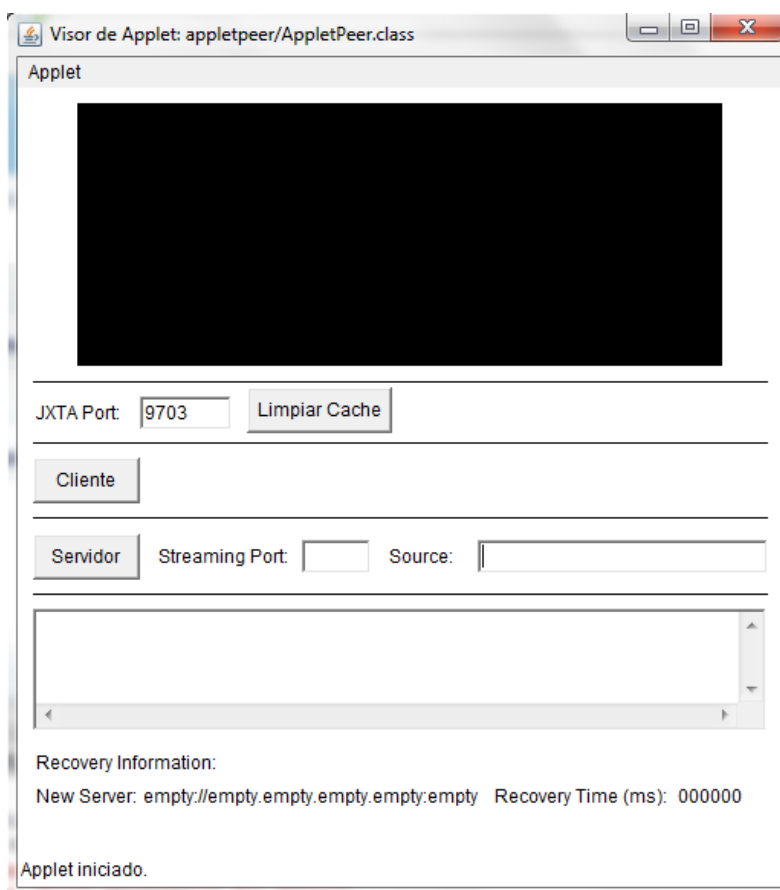


Figura 4. Interfaz prototipo

- Si el nodo está destinado a operar como Servidor en un mismo equipo, digite el puerto y la dirección IP del servidor que actuará de proveedor del servicio de streaming; en caso de que sean nodos únicos, sólo digite la dirección IP del servidor, y a continuación presione el botón 'Servidor'.
- Si el nodo está destinado a operar como Cliente, solamente presione el botón 'Cliente'.

2. Sección de mensajes

La sección de `Mensajes` muestra los mensajes en tiempo real, que describen las funciones que se están ejecutando actualmente en la aplicación. Los mensajes describen desde la conexión con un nodo servidor, así como también la búsqueda del servicio y un nodo que lo esté prestando en caso de falla en la primera conexión, el tiempo de respuesta del servidor, el tiempo de recuperación, entre otros.

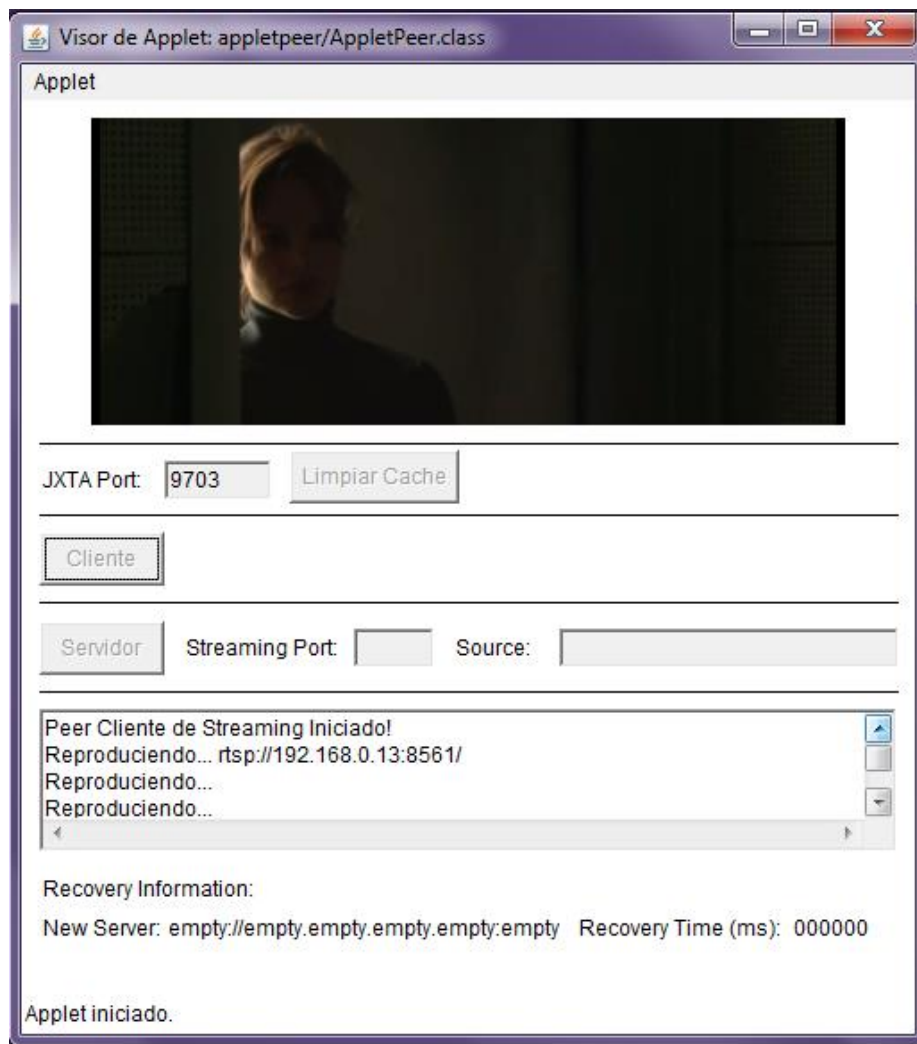


Figura 5. Sección de mensajes

En la figura, se muestra la sección de mensajes con algunos de ellos que se pueden visualizar en la aplicación.

3. Salir de la Aplicación

Para salir de la aplicación, desactivar un servidor o cerrar un cliente, puede cerrar la ventana en la que se está ejecutando la aplicación.

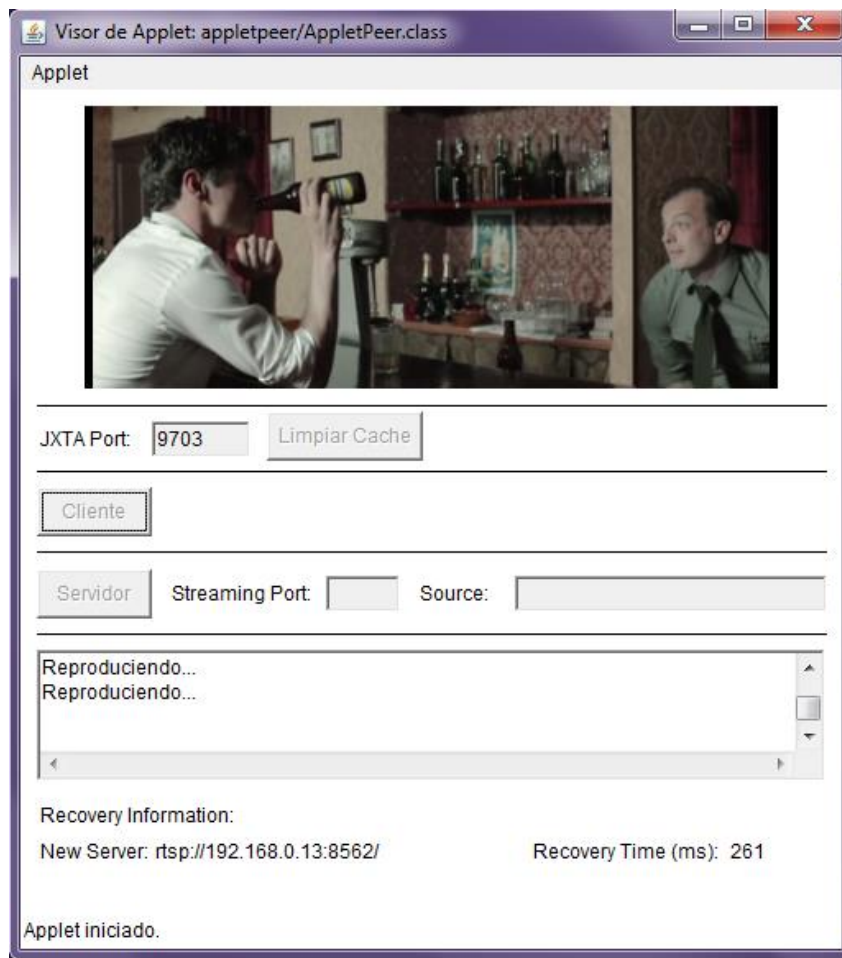


Figura 6. Cerrar aplicación

2. Diagrama de clases nodo cliente

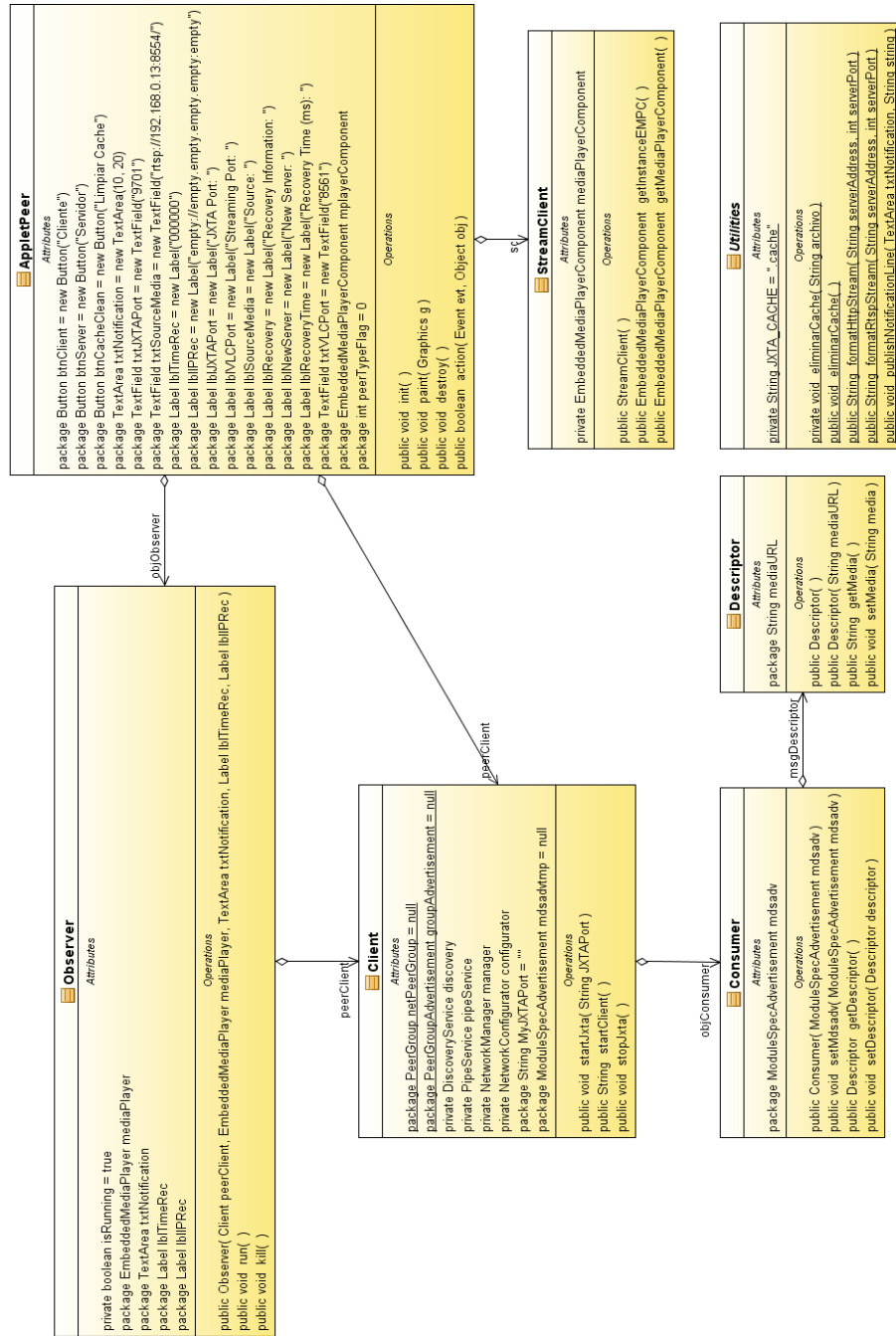


Figura 8. Diagrama de clases nodo cliente⁴

⁴ Para ver en detalle el diagrama \Documentos\Anexos (Fuentes)\Diagrama de Clases - Nodo Cliente.png del CD-ROM anexo.

3. Diagrama de clases nodo servidor

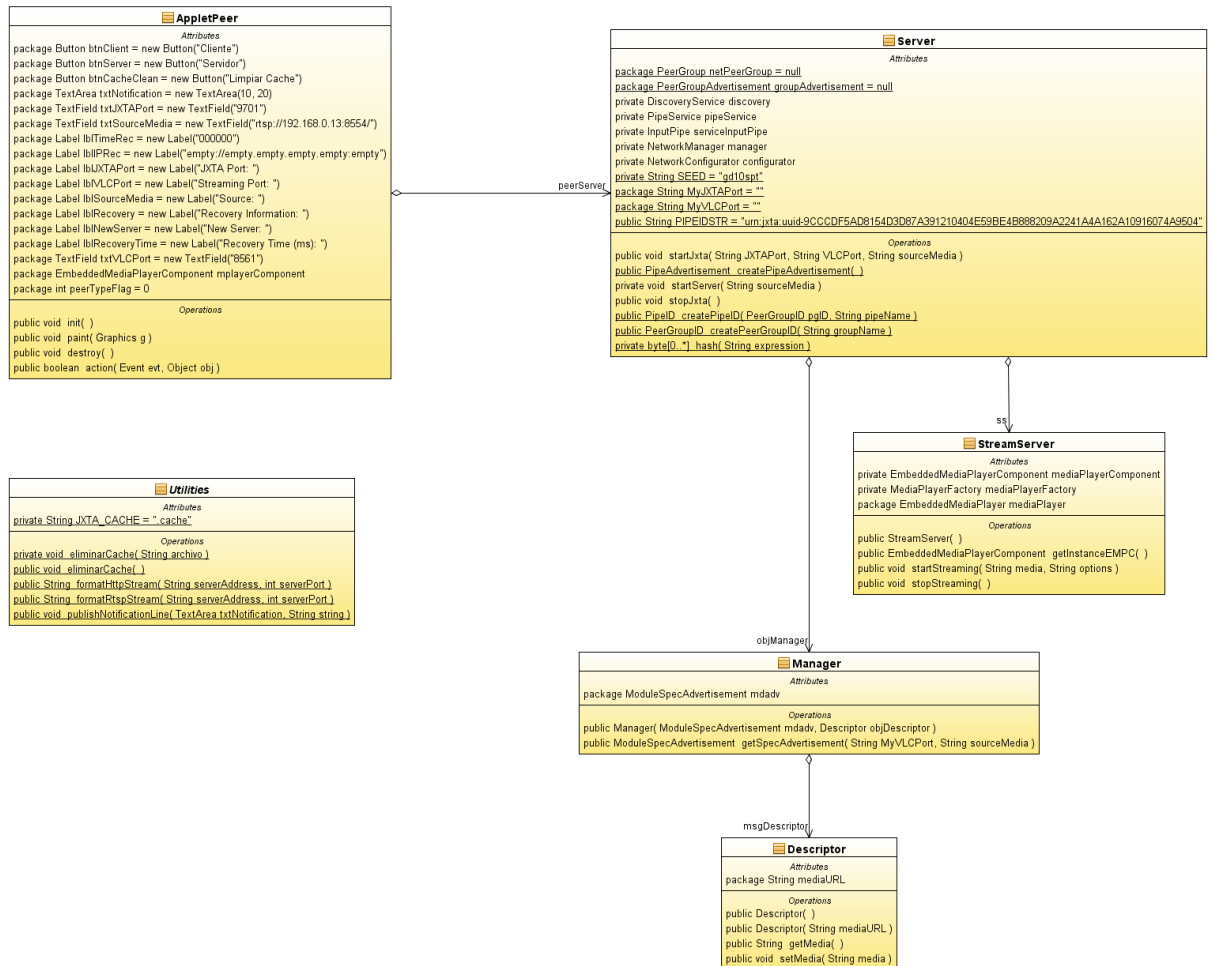


Figura 9. Diagrama clases nodo servidor⁵

⁵ Para ver en detalle el diagrama \Documentos\Anexos (Fuentes)\Diagrama de Clases - Nodo Servidor.png del CD-ROM anexo

ANEXO H: PRUEBAS

3. Pruebas manuales

La complejidad de aplicación peer-to-peer provoca que las pruebas unitarias se queden cortas en el momento de verificar la validez del prototipo software. Por lo cual, se dispone de un conjunto de pruebas manuales a través de la creación de múltiples instancias de nodos peer-to-peer y escenarios.

Los escenarios definidos para la verificación de la técnica son los que se describen a continuación:

Nro. de Escenario	Nodos Servidores	Nodos Clientes	Característica
1	Uno	Muchos	Los nodos se conectarán a la red y los nodos clientes serán iniciados primero, luego se inicia el nodo servidor proveedor del servicio de streaming y que será consumido por los nodos cliente ya conectados.
2	Uno	Muchos	Los nodos se conectarán a la red y el nodo será servidor iniciado primero, luego se inician los nodos clientes que consumirán el servicio de streaming que provee el primer nodo servidor lanzado.
3	Dos	Muchos	Primero se inicia un nodo servidor que proveerá el servicio de streaming, luego se inician los nodos clientes que empezarán a consumir el servicio, para luego conectar y lanzar el segundo servidor. Por último se detiene el primer servidor y verificar que los nodos clientes recuperan el servicio conectándose al segundo nodo servidor.
4	Dos	Muchos	Aplica la secuencia del escenario anterior, para luego iniciar de nuevo el primer servidor y detener el segundo servidor.
5	Dos	Muchos	Primero se inician todos los nodos servidores que proveerán el servicio de streaming; luego se inician los nodos clientes que empezarán a consumir el servicio.
6	Dos	Muchos	Primero se inician todos los nodos clientes; luego se inician los nodos servidores que empezarán a prestar el servicio de

			streaming. Una vez los clientes estén conectados al primer servidor consumiendo el servicio, se conecta y lanza el segundo nodo servidor.
7	Tres	Muchos	Primero se inician todos los nodos servidores que proveerá el servicio de streaming; luego se inician los nodos clientes que empezarán a consumir el servicio conectándose de forma aleatoria a los nodos servidores. Por último se detiene un nodo servidor aleatorio y verificar que los nodos clientes recuperan el servicio conectándose al cualquier nodo servidor que siga conectado.
8	Tres	Muchos	Primero se inician dos nodos servidores que proveerá el servicio de streaming; luego se inician todos los nodos clientes que empezarán a consumir el servicio conectándose de forma aleatoria a los nodos servidores que están conectados. Por último se conecta el tercer nodo servidor y se detienen los dos primeros iniciados anteriormente y verificar que los nodos clientes recuperan el servicio conectándose al último nodo servidor que se conectó.
9	Cuatro	Muchos	Primero se inician todos los nodos servidores que proveerá el servicio de streaming; luego se inician los nodos clientes que empezarán a consumir el servicio conectándose de forma aleatoria a los nodos servidores. Por último se detiene un nodo servidor aleatorio y verificar que los nodos clientes recuperan el servicio conectándose al cualquier nodo servidor que siga conectado.
10	Cuatro	Muchos	Primero se inician tres nodos servidores que proveerá el servicio de streaming; luego se inician todos los nodos clientes que empezarán a consumir el servicio conectándose de forma aleatoria a los nodos servidores que están conectados. Por último se conecta el cuarto nodo servidor y se detienen los tres primeros iniciados anteriormente y verificar que los nodos clientes recuperan el servicio conectándose al último nodo servidor que

			se conectó.
11	Cinco	Muchos	Primero se inician todos los nodos servidores que proveerá el servicio de streaming; luego se inician los nodos clientes que empezarán a consumir el servicio conectándose de forma aleatoria a los nodos servidores. Por último se detiene un nodo servidor aleatorio y verificar que los nodos clientes recuperan el servicio conectándose al cualquier nodo servidor que siga conectado.
12	Cinco	Muchos	Primero se inician cuatro nodos servidores que proveerá el servicio de streaming; luego se inician todos los nodos clientes que empezarán a consumir el servicio conectándose de forma aleatoria a los nodos servidores que están conectados. Por último se conecta el quinto nodo servidor y se detienen los cuatro primeros iniciados anteriormente y verificar que los nodos clientes recuperan el servicio conectándose al último nodo servidor que se conectó.
13	Seis	Muchos	Primero se inician todos los nodos servidores que proveerá el servicio de streaming; luego se inician los nodos clientes que empezarán a consumir el servicio conectándose de forma aleatoria a los nodos servidores. Por último se detiene un nodo servidor aleatorio y verificar que los nodos clientes recuperan el servicio conectándose al cualquier nodo servidor que siga conectado.
14	Seis	Muchos	Primero se inician cinco nodos servidores que proveerá el servicio de streaming; luego se inician todos los nodos clientes que empezarán a consumir el servicio conectándose de forma aleatoria a los nodos servidores que están conectados. Por último se conecta el sexto nodo servidor y se detienen los cinco primeros iniciados anteriormente y verificar que los nodos clientes recuperan el servicio conectándose al último nodo servidor que

			se conectó.
--	--	--	-------------

Tabla 24. Escenarios de pruebas

En la tabla 25, se muestra el formato utilizado para registrar los resultados de los escenarios de prueba. Estas pruebas se realizaron de manera local en un mismo equipo de cómputo diferenciando los nodos a través de la definición de un puerto JXTA y también otro para el servicio media de streaming.

Escenario:						
Instancia local	Rol	Puerto JXTA	Puerto Streaming	Puerto JXTA Consumo Actual	Puerto JXTA Consumo Nuevo	Tiempo de Recuperación Milisegundos
Descripción de resultado:						

Tabla 25. Formato registro de resultados

A continuación se describen los campos del formato de registro de resultados:

- **Escenario:** Número y descripción del escenario especificado anteriormente.
- **Instancia Local:** Corresponde al número de instancia local que se le asigno previamente.
- **Rol:** Es el rol que desempeña cada nodo en la red. Puede ser un nodo cliente o un nodo servidor.
- **Puerto JXTA:** Es el puerto JXTA de cada nodo conectado en la red.
- **Puerto Streaming:** Es puerto utilizado por la instancia local servidor para el servicio de streaming.
- **Puerto JXTA Consumo Actual:** Representa el nodo local por el cual está consumiendo el servicio.
- **Puerto JXTA Consumo Nuevo:** Corresponde al nodo servidor local al cual se conectó nuevamente después de un fallo en su puerto JXTA Consumo Actual.
- **Tiempo de Recuperación:** Representa el tiempo en milisegundos que tarda el nodo cliente en recuperar el servicio y ejecutarlo nuevamente.
- **Descripción de resultado:** Es la descripción de los resultados que se obtuvieron en el desarrollo del escenario de prueba.

REFERENCIAS BIBLIOGRÁFICAS

- [1] K. Beck, *Extreme Programming Explained: Embrace Change*. Addison-Wesley Professional, 1999, p. 224.
- [2] I. T. N. DawningStreams, *Practical JXTA II - Cracking the P2P puzzle*. Lulu Enterprises, Inc., 2010.
- [3] A. Andrade, D. Luna, and P. Magé, “Técnica de Búsqueda para la Prestación de Servicios, basada en Sistemas Multi-Agente y Redes P2P,” *Sist. Telemática*, pp. 65–75, 2012.
- [4] Oracle, “Guía de Instalación JDK 8 y JRE 8,” 2015. [Online]. Available: http://docs.oracle.com/javase/8/docs/technotes/guides/install/install_overview.html.
- [5] V. Wiki, “Guía de Instalación de VLC,” 2015. [Online]. Available: https://wiki.videolan.org/Documentation:Installing_VLC/.