



INSTITUTO POLITÉCNICO NACIONAL

Escuela Superior De Computo



Proyecto final “ReaSync”

Aplicaciones Para Comunicaciones De Red

Alumnos:

Diaz Rodríguez Julio

Ramos Reyes Brayan

Profesora: Bautista Rosales Sandra Ivette

Grupo: 3CV8

Contenido

Introducción	3
Planteamiento del problema	3
Descripción general del proyecto	3
ReaSync Server	3
Modo de trabajo	4
ReaSync	4
Justificación	5
Algunos de los temas vistos en el curso que usaremos son:	5
Look and Feel del proyecto	5
Vistas ReaSync Server	5
Bienvenida	5
Iniciar Servicios	6
Tabla de clientes	6
Log	7
Vistas ReaSync	7
Bienvenida	7
Servicios	7
Sincronización	8
Log	8
Instalación e implementación de ReaSync	9
Conclusiones	9

Introducción

Como proyecto final de la unidad de aprendizaje de aplicaciones para comunicaciones en red, se nos indicó aplicar temas vistos a lo largo del semestre para darle solución a alguna problemática que nos planteáramos. Nosotros decidimos hacer un sincronizador de música entre dispositivos personales (PC).

Planteamiento del problema

Cuando tenemos distintos dispositivos en nuestro hogar, como una PC una LAPTOP, o incluso varias. Es posible que mientras escuchemos música alguna canción nos guste y decidamos descargarla. O simplemente guardemos un disco que hayamos comprado y lo pongamos en nuestra biblioteca de música para ya no tener que insertarlo. Pero cuando pasamos de la PC a la laptop o viceversa, ese nuevo dispositivo no tendrá esas canciones a menos que usemos un servicio para transferir esa música al otro dispositivo, o copiemos con una USB de un dispositivo a otro. En la actualidad hay software basando su funcionamiento en la nube, los cuales nos permiten tener sincronizados los directorios que le indiquemos en cualquier dispositivo, donde dependemos de la capacidad de almacenamiento que este servicio nos brinde, además que está hecho para cualquier tipo de archivo.

Descripción general del proyecto

ReaSync es un sistema basado en la arquitectura cliente – servidor. El cual permite la sincronización de música entre dispositivos personales pensado para funcionar eficazmente en el hogar.

ReaSync consta de dos programas para funcionar:

1. ReaSync Server. El cual se encarga de gestionar los archivos de música con los que los clientes cuentan, llevando un registro en una base de datos. También gestiona uno o varios clientes que se conecten a este, y atenderá las solicitudes de cada uno de manera independiente.
2. ReaSync. Un cliente que se conecta a ReaSync server y se encarga de detectar los cambios locales contra los registros del servidor y hacer los cambios pertinentes para llevar a cabo la sincronización.

ReaSync Server

ReaSync Server para su funcionamiento requiere de dos servicios adicionales para proporcionar los suyos.

1. Un servidor FTP
 - a. En el caso de nuestra implementación hacemos uso de FileZilla Server. Este servicio es el que nos provee el canal adecuado para la transferencia de archivos entre cliente y servidor.
2. Un Gestor de base de datos
 - a. En nuestra implementación hacemos uso de MySQL. Este servicio nos permite guardar en la base de datos los registros de los archivos con los que cuenta el servidor.

Modo de trabajo

ReaSync Server trabaja por medio de sockets para establecer la comunicación con los clientes que se conectan. Cuando inicia el server, este queda a la espera de conexiones de algún cliente (esta permanecerá activa hasta que el cliente cierre la aplicación o se apague su equipo). Cuando el server detecta una nueva conexión de un cliente, esta queda almacenada en un gestor de conexiones las cuales estarán guardadas en memoria en una lista. Cada cliente que se conecte formara parte de un pull de conexiones. De esta manera cada cliente será atendido de manera independiente.

Cada hilo cliente, cuenta con un esperador de peticiones, el cual cuando detecte una nueva, este determinara que acción realizar y así darle la pertinente respuesta al cliente.

En el modo de sincronización, el cliente envía peticiones al servidor como:

- registroArchivosMusica - El server hace una compilación de todos los registros que tiene, los serializa y vía la conexión socket se los envía al cliente.
- guardarRegistroArchivoMusica – El cliente manda un archivo de música con la información requerida para que el server guarde en la base de datos el registro de dicho archivo
- eliminarRegistroArchivoMusica – El cliente manda un archivo de música con la información requerida para que el server elimine de la base de datos el registro de dicho archivo.

ReaSync

ReaSync para su funcionamiento solo requiere que todos los servicios de ReaSync Server se encuentren en línea.

Al igual que el server, ReaSync mantiene una conexión activa con el servidor por medio de un socket.

ReaSync realiza las siguientes tareas en el equipo y directorio que se le indica en el dispositivo:

- Verifica que exista un archivo llamado cambios.dat, este archivo es un registro serializado de una lista de elementos generados tras recorrer el árbol de archivos del directorio que se desea sincronizar en el dispositivo. Si lo encuentra lo cargara cuando se le solicite para otras funciones que se especificaran más adelante.
- Al momento de requerirse el proceso de sincronización, el programa genera un estado actual del árbol de archivos del directorio de sincronización
- Cuando se es solicitado, se genera un objeto llamado “Cambios Locales “los cuales contendrán los archivos del estado anterior y el estado actual de archivos, los cuales determinaran que archivos son nuevos, iguales y eliminados del directorio de sincronización.
- Cuando el proceso de sincronización comienza los cambios locales se comparan contra los registros del servidor y de esta manera se determinan que archivos se deben subir al servidor, así como eliminar de él y localmente y descargar también.
- Cuando el proceso de cambios locales y remotos finaliza, el programa crea un nuevo objeto de cambios locales el cual será el nuevo estado anterior con el que se tendrá registro (cambios.dat).
- El proceso de sincronización se puede iniciar de manera automática (buscando cambios cada 60 segundos), o ejecutando la acción manual con un botón en la vista del programa.

Justificación

Consideramos que este es un proyecto que contiene muchos de los temas vistos a lo largo del semestre. Además, tratamos de dar solución una problemática real que se puede presentar en nuestras vidas diarias. Con este proyecto estamos creando un espacio donde de manera automática con muy pocos clics, la música de los usuarios se sincronice de manera automática, rápida y sin esfuerzo.

Algunos de los temas vistos en el curso que usamos son:

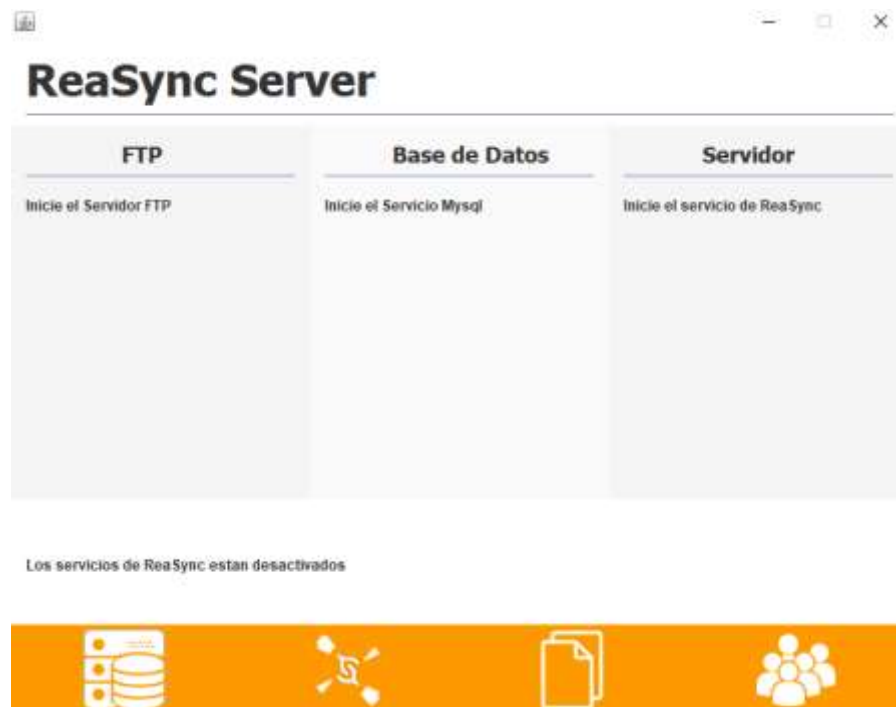
- FTP (Transferencia de archivos)
- ServerSocket/Socket
- Hilos/ExecutorService (Pull de conexiones)
- Arquitectura Cliente/Servidor
- Serialización (Para el envío de mapeo de archivos en los directorios para sincronizar, así como de peticiones y respuestas)

Look and Feel del Proyecto

Vistas ReaSync Server

Bienvenida

En la vista de bienvenida el usuario puede ver el estatus de los servicios con los que funciona el servidor



Iniciar Servicios

En esta vista se pueden iniciar y comprobar los servicios FTP y MYSQL, además de iniciar el servicios de ReaSync cuando los anteriores están en línea.

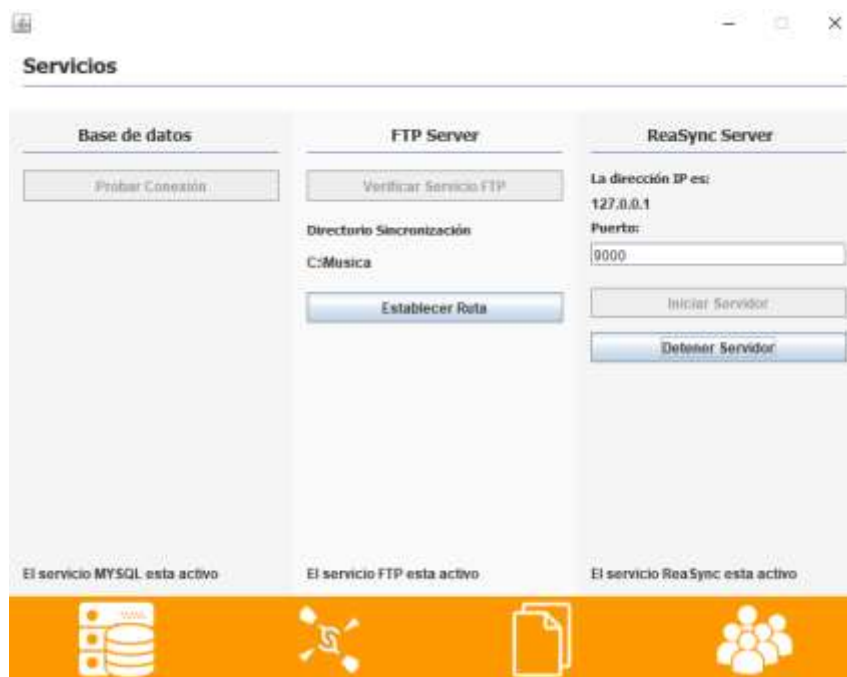


Tabla de clientes

En esta vista se pueden visualizar los clientes que se encuentran conectados al servicio, se muestra su id su dirección IP y su nombre de usuario



Log

En esta vista se puede consultar con un poco de detalle en fecha y hora que ha pasado en el servidor.



Vistas ReaSync

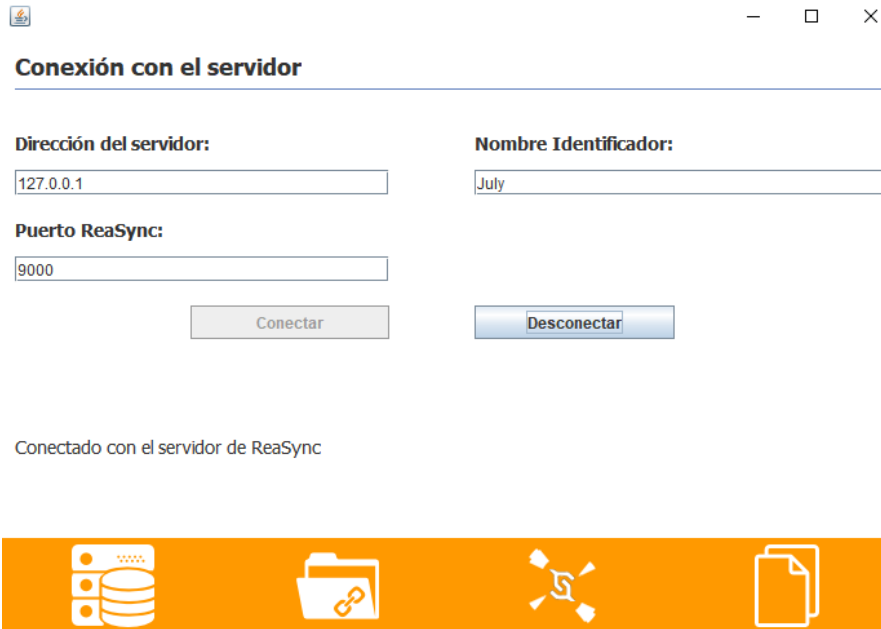
Bienvenida

Esta es la pantalla que vera el cliente cuando inicie el programa, se muestran los estados de conexión al servidor y de la sincronización.



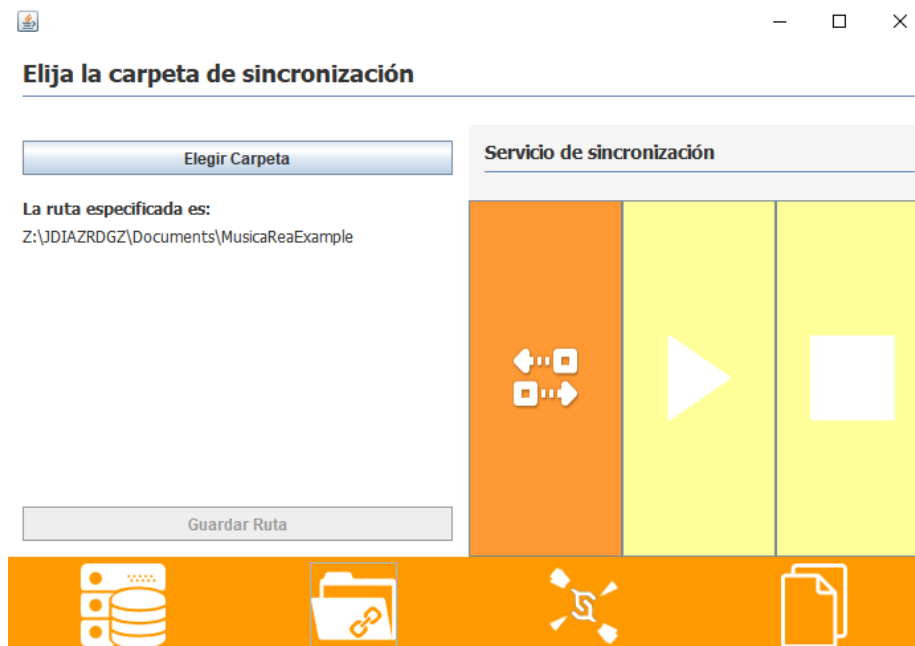
Servicios

En esta pantalla el cliente hace la conexión con el servidor, además especifica su nombre identificador en el servidor.



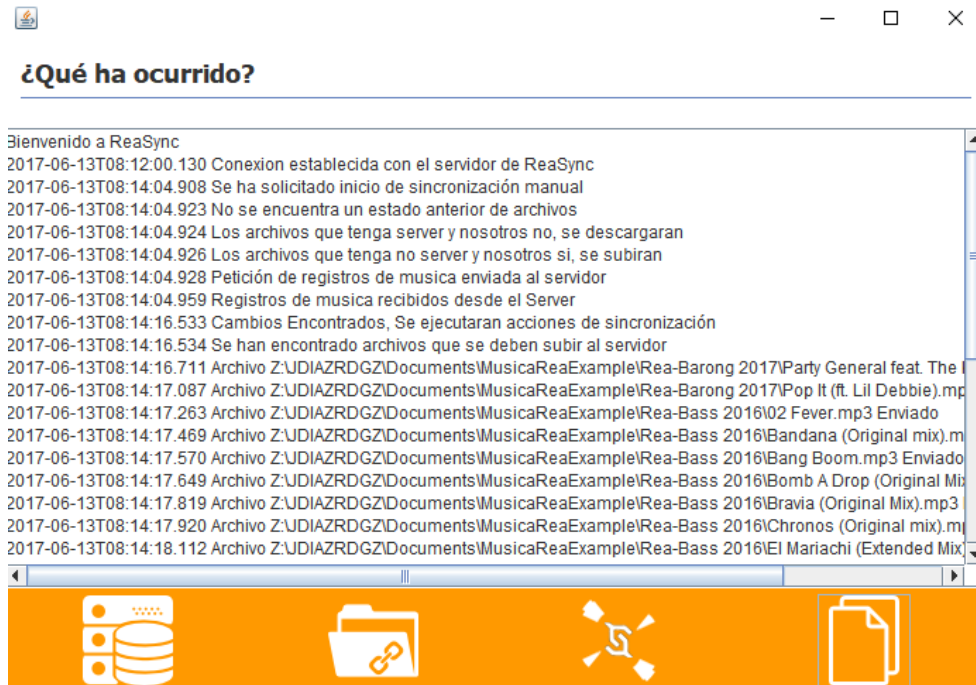
Sincronización

En esta vista el cliente escoge el directorio que desea sincronizar, además activa la sincronización manual o automática.



Log

En esta ventana el cliente puede consultar de manera detallada los estados de su programa así como el transcurso de la sincronización.



Instalación e implementación de ReaSync

El único programa que requiere componentes externos a ReaSync es ReaSync server y debe ser configurado correctamente de la siguiente manera:

- Se debe tener instalado el servicio MySQL, dentro de este debe haber un usuario **root**, con un password **46143**/**. Además, se tiene que importar el script .sql que viene junto con los archivos del proyecto; esto para crear la base de datos y sus tablas correspondientes.
- Se debe tener instalado y activo un servidor FTP, en nuestro caso nosotros probamos con FilleZilla server. Debe estar corriendo en el puerto **21** y con un usuario **reasync** así como un password **reasync**. A este usuario desde el server se le debe otorgar un directorio donde podrá escribir los archivos en gestión.
- Cuando se inicie por primera vez el servidor ReaSync o en lo posterior, El directorio que se especifica en el server FTP debe ser el mismo que se escoge en la ruta de sincronización en el server. Si esto no es así no funcionara la sincronización.

Conclusiones

A lo largo del curso vimos distintas formas en las que las aplicaciones pueden comunicarse entre sí. Aprendimos que todas las aplicaciones pueden comunicarse a través de su capa de aplicación a través de sockets, sea de la forma más pura o por medio de un protocolo como FTP, HTTP, P2P etc. Al mismo tiempo, la manera en la que las aplicaciones pueden comunicarse puede variar de acuerdo a la arquitectura y patrón con los que fueron diseñados, como cliente/servidor o RMI entre otros.

En este proyecto implementamos algunas de las herramientas que vimos a lo largo del semestre. Con cada practica fuimos mejorando la forma en la que tratábamos a cada socket cliente hasta llegar el punto de tratar a cada uno con un hilo específico. También en este proyecto implementamos muchas funciones de lectura/escritura de archivos de manera local utilizando las nuevas tecnologías

que proporciona el JDK 8 de java, esto, nos ayudó a practicar, crear y reforzar conocimientos sobre nuevas metodologías que nos proporciona este lenguaje para hacer tratamiento de listas y lectoescritura.

No cabe duda que en este proyecto utilizamos muchos conocimientos que adquirimos a lo largo del semestre. Con este proyecto utilizamos bastantes nuevas técnicas utilizando nuevas metodologías en el lenguaje JAVA. Y lo más importante es que este proyecto lo podemos ocupar en nuestra vida cotidiana para transferir nuestra música de manera rápida y sencilla.