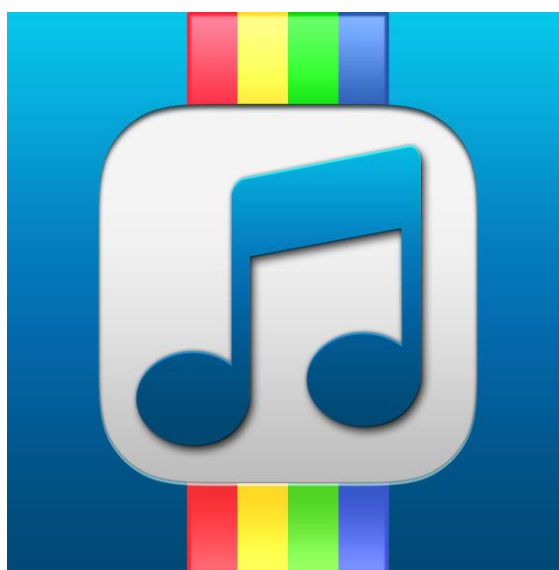


## ESI MUSIC

Gestión de Sistemas de Información

3º Ingeniería Informática



Fernando Huertas Olivares  
Manuel Hurtado Lillo

23 de abril de 2018

## ÍNDICE

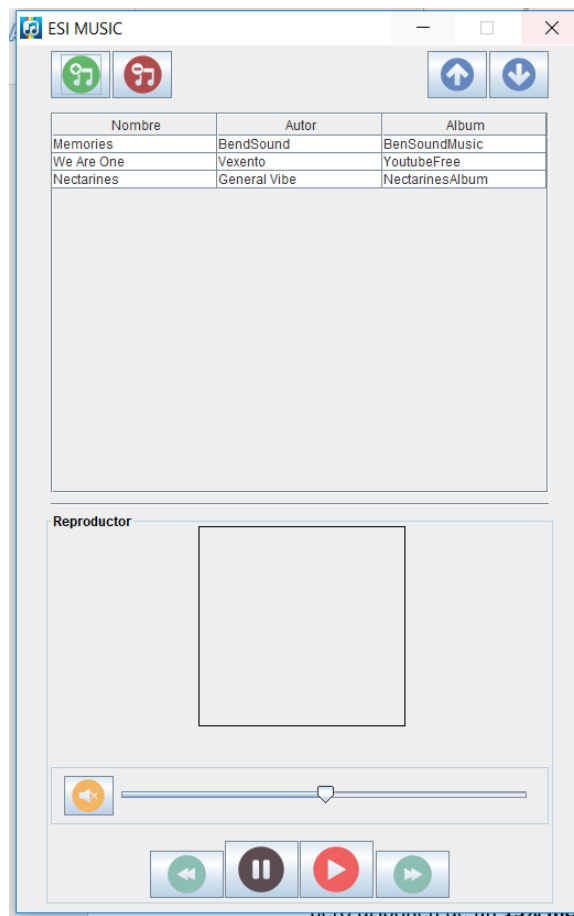
1. FUNCIONAMIENTO DEL PROGRAMA .....	3
2. TECNOLOGÍA UTILIZADA .....	5
2.1 JAVA y Windows Builder .....	5
2.2 MYSQL .....	5
2.3 LIBRERÍA JLGUI DE JAVAZOOM. ....	6
3. LABBOOK .....	6
4. PROBLEMAS DURANTE EL DESARROLLO .....	7

## 1. FUNCIONAMIENTO DEL PROGRAMA

**ESI Music** es una aplicación basada en la reproducción de música la cual está conectada a una base de datos, esta base de datos puede ser tanto local como alojada en un servidor.

La aplicación tiene una interfaz sencilla y que cumple con las funcionalidades necesarias de un reproductor simple.

Al iniciar la aplicación se mostrará la interfaz de lista de reproducción que está asociada con el reproductor:

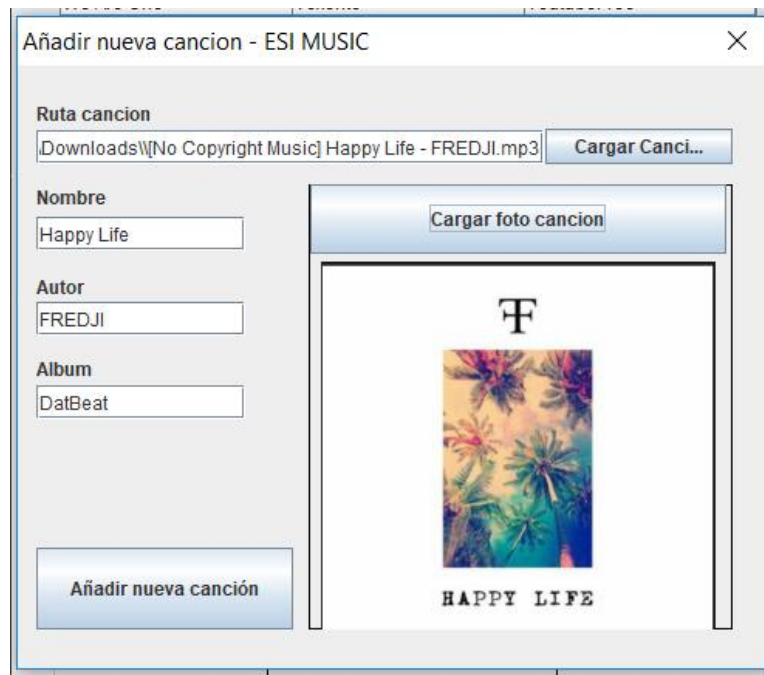


Podemos ver que todo es muy intuitivo y sigue el diseño establecido de un reproductor estándar.

Las funcionalidades son las siguientes:

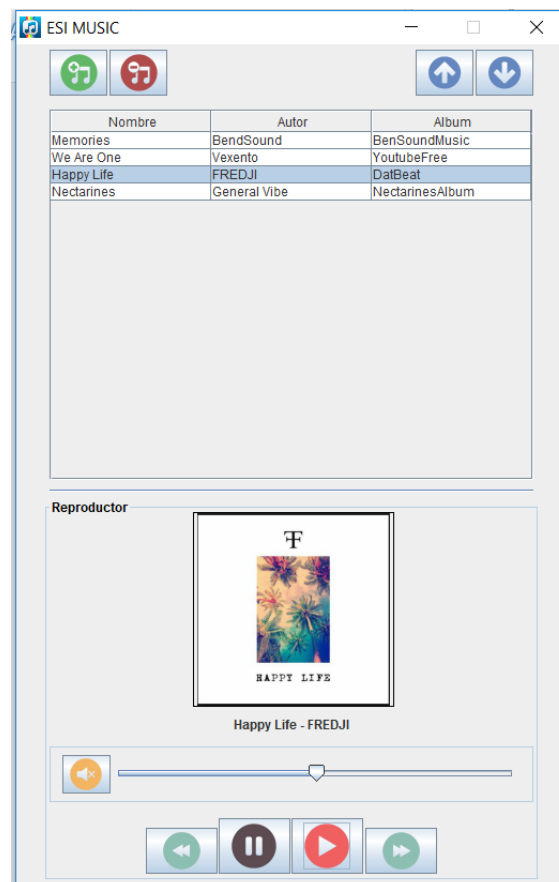
La primera vez que iniciemos el reproductor si la base de datos esta vacía deberemos **añadir una canción nueva** si queremos reproducir algo, para ello se pulsara en el botón de arriba a la izquierda, concretamente el que tiene el símbolo de “añadir una canción”

Esto nos abrirá una ventana donde tendremos que añadir los datos de la canción: titulo, autor, álbum, ruta de la canción y ruta de la imagen del álbum. Un ejemplo seria el siguiente:



Al añadirla, la canción será agregada a la base de datos y la interfaz principal de nuestra canción.

La funcionalidad principal de nuestra aplicación es la **reproducción** de estas canciones, para ello seleccionamos una canción de la lista y le damos a "play". La aplicación conectará con la base de datos para coger la ruta de la canción y la reproducirá:



Cuando una canción esté en el reproductor, podremos pausarla y reanudarla con el botón de “pausa”, volver a reproducirla desde el inicio con “play”, ajustar su volumen con la slidebar o mutearla con el botón de “mute”, además también seremos capaces de pasar a la siguiente o a la anterior canción de la lista de reproducción.

Por último, podemos **eliminar canciones** de la lista con el botón “eliminar”, estas canciones también serán eliminadas de la base de datos.

## 2. TECNOLOGÍA UTILIZADA

Para el desarrollo de nuestra aplicación hemos usado las siguientes tecnologías:

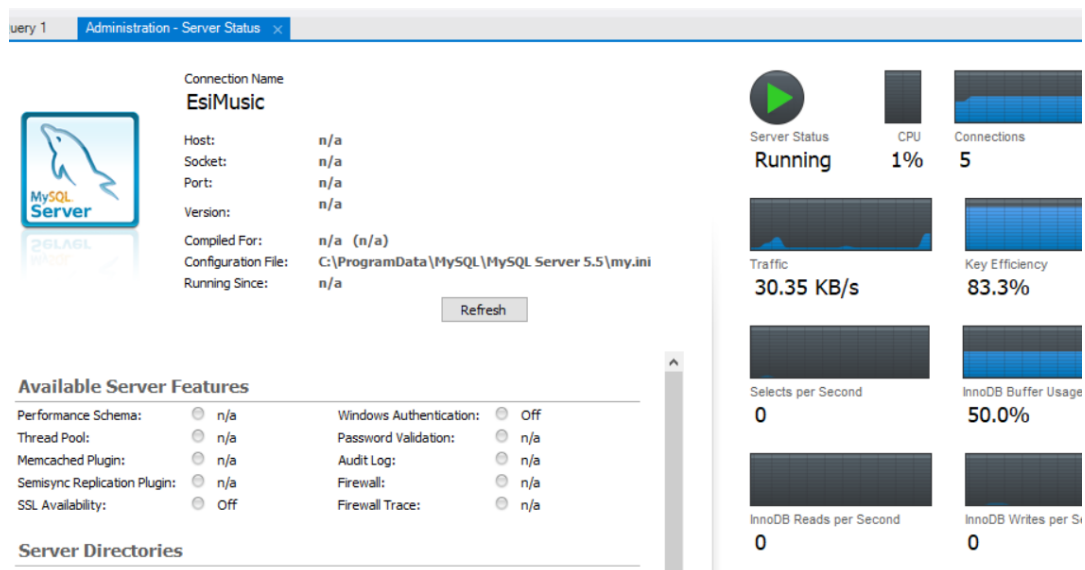
### 2.1 JAVA Y WINDOWS BUILDER.

Hemos elegido **JAVA** porque es el lenguaje que mejor conocemos y el que para nosotros, mas facil es realizar aplicaciones e interfaces gracias a **Windows Builder**. Para asociar estas dos herramientas hemos usado **Eclipse**.

### 2.2 MYSQL

**MySQL** es un servicio para crear y realizar bases de datos, nosotros hemos realizado una base de datos con una sola tabla: “canciones” en ella se guarda los metadatos de las canciones: titulo, autor, álbum, ruta de la canción y ruta de la imagen del álbum, como se puede ser este es un sistema de almacenamiento de base de datos multimedia enlazado, ya que no hemos guardado la canción en binario sino una referencia a donde esta almacenado tanto el audio como una imagen.

Al no tener un servidor especializado, la hemos desplegado en **local**:



Tampoco disponemos de un servidor de almacenamiento asique hemos usado el mismo ordenador en el que se despliega la base de datos y la aplicación como almacenamiento.

Para conectar la base de datos con nuestra aplicación hemos usado la librería de MySQL llamado mysql-connector.

### 2.3 LIBRERÍA JLGUI DE JAVA ZOOM

Por ultimo, hemos usado una librería externa para facilitar la reproducción de archivos, concretamente la librería “**jlgui**”, la cual ha sido desarrollada por javazoom y su uso es concretamente libre. Esta librería es bastante completa y nosotros nos hemos centrado en la parte de “**BasicPlayer**” el cual facilita la implementación de las funcionalidades que tiene un reproductor.



Para poder comprender como funciona la librería hemos usado una biblioteca con todos los métodos que nos facilita el autor. Puede consultarse [aquí](#).

## 3. LABBOOK


En este apartado vamos a hablar sobre el calendario que hemos seguido para desarrollar la aplicación, en principio intentamos seguir el calendario previsto al empezar el proyecto, pero por acumulación de trabajo de otras asignaturas o de esta misma algunas semanas no lo hemos empleado todo el tiempo que teníamos planeado, a su vez, otras semanas le hemos dedicado más tiempo.

Le hemos dedicado una hora y cuarto todos los martes quedando en la biblioteca para facilitar el trabajo en equipo. Este es el desglose de fechas y trabajo desde entonces hasta el día de hoy:

FECHA	TRABAJO REALIZADO
13-02-2018:	Fase inicial del proyecto, pensamos como y de que lo vamos a realizar.
20-02-2018:	Llegamos a la conclusión de que queremos realizar un reproductor de música asociado a una base de datos.
27-02-2018:	Buscamos información sobre como podemos realizarlo y empezamos a desarrollar las interfaces del proyecto.
6-03-2018:	Acabamos el desarrollo del interfaz y empezamos a trabajar en la capa de persistencia.
13-03-2018:	Creamos la base de datos en mysql y acabamos la capa de persistencia para poder conectarnos a ella. Se realiza la primera conexión con la base de datos de manera satisfactoria.
20-03-2018:	Trabajamos en el paso de la información de la interfaz a la base de datos y dejamos listo el tema de añadir canciones a la base de datos y que mantenga la integración con la interfaz.
27-03-2018:	Buscamos información sobre como realizar reproducción de archivos de audio en java y hallamos las librerías de JavaZoom
3-04-2018:	Conseguimos aplicar estas librerías y escuchamos por primera vez audio saliendo de nuestra aplicación.
10-04-2018:	Vemos que la librería usada no es lo suficientemente compleja y usamos una mas avanzada del mismo creador para poder cumplir todas las funcionalidades de nuestra aplicación.
17-04-2018:	Pulimos algunos detalles y realizamos la documentación.
24-04-2018:	Entrega y defensa del proyecto.

#### 4. PROBLEMAS DURANTE EL DESARROLLO

Hemos presentado algunos problemas a la hora de desarrollar el proyecto, pero nada que no se haya podido solucionar. Son los siguientes:

 **Uso de servidor y base de datos en local:** Al no contar con un servidor de almacenamiento o un sitio donde desplegar nuestra base de datos hemos tenido que hacerlo en local, esto es una sensación agri dulce ya que aunque simule el uso de uno externo, no es tan satisfactorio como uno desea. El principal problema de esto es que al mover la aplicación para desplegarla en otro equipo tendremos que ejecutar en ese equipo también la base de datos la cual estará vacía ya que los archivos permanecieron en el primer equipo. Con un servidor de almacenamiento en la nube esto estaría solucionado.

Además, hemos elegido una librería que permite reproducir desde URL, así será bastante fácil realizar el salto a mantener todo en algo externo al lugar donde se despliega la aplicación.

- ✚ **Cambio de librería debido a pocas funcionalidades:** Al principio usamos la librería “Player” que nos facilita JavaZoom, esta es muy fácil de implementar y permite las acciones de reproducir, pausar y parar. El problema venia al no permitir trabajar con el volumen cuando era una de nuestras funcionalidades de nuestra aplicación, por ello dimos el salto a “BasicPlayer” una versión superior, la cual es más difícil y engorrosa de desarrollar. Esta permite trabajar con volumen y bastantes cosas más que podrían servir si en un futuro se decide mejorar la aplicación.