



TRACKIO

ADRIÁN BARRIO ANDRÉS

JAVA.SWING



Índice

Contenido

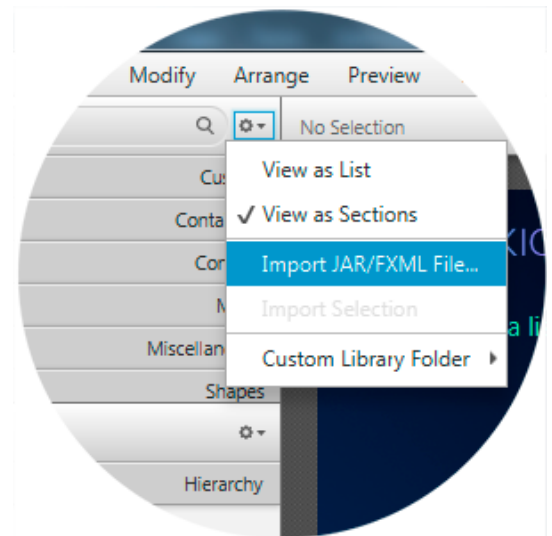
Componentes	1
Estructura de la aplicación	2
Características	3
Capturas	5
Snippets útiles	6

Componentes

Para los iconos he usado la librería [FontAwesome](#) adaptada a JavaFX por Jerady que se puede encontrar [aquí](#). Nos provee una serie de iconos para usar como controles personalizados que son en formato SVG o vectorial.

Para importarla en SceneBuilder hay que pulsar sobre el pequeño engranaje y que hay en la lista de componentes y pulsar sobre "Import JAR/FXML File...", después buscamos el JAR que está en la raíz del proyecto en la carpeta "lib" y lo importamos.

Automáticamente tendremos disponibles los iconos para usar en JavaFX como cualquier otro control o widget en la sección "Custom".



Estructura de la aplicación

Se ha utilizado el patrón de diseño MVC incluido en JavaFX y la estructura descrita a continuación

- **model** – Contiene el modelo de datos con el retorno de los datos necesarios que el controlador necesita.
- **view** – Contiene el diseño de las vistas con las que el usuario interactúa y los controladores.
- **view.css** – Archivos de CSS.
- **view.img** – Archivos de imágenes.
- **util** – Contiene utilidades para interactuar con funciones de los controladores, la mayoría de ellas estáticas.

Funcionamiento

Cuando creas una lista se almacena en el registro de Windows, mediante la API Preferences de Java, en caso de MacOSX se almacena en un archivo XML. Cuando pulsamos sobre una lista recorre su directorio y nos muestra las canciones en MP3 que el reproductor puede reproducir, cuando hacemos doble click sobre una canción se desencadenan una serie de eventos que hacen que todo funcione en tiempo real.

Github

<https://github.com/statickidz/Trackio>

Estructura de archivos

com.statickidz.trackio.MainApp: Inicializa la aplicación, inserta el contenedor del reproductor dentro del Root e inicia ambos. También es el encargado de mostrar el diálogo que sirve para añadir una lista de reproducción.

com.statickidz.trackio.util.TrackListUtil: Sirve para guardar, traer, borrar y refrescar las listas de reproducción.

com.statickidz.trackio.util.TrackUtil: Sirve para recorrer directorios en busca de archivos mp3, extraer los metadatos del fichero (imágenes, título, autor, año...).

com.statickidz.trackio.util.EffectUtil: Aplica efectos sobre imágenes para que se muevan al cambiar de canción.

com.statickidz.trackio.view.PlayerController: Se encarga de recibir los eventos de la vista Player.fxml y realiza operaciones sobre los controles.

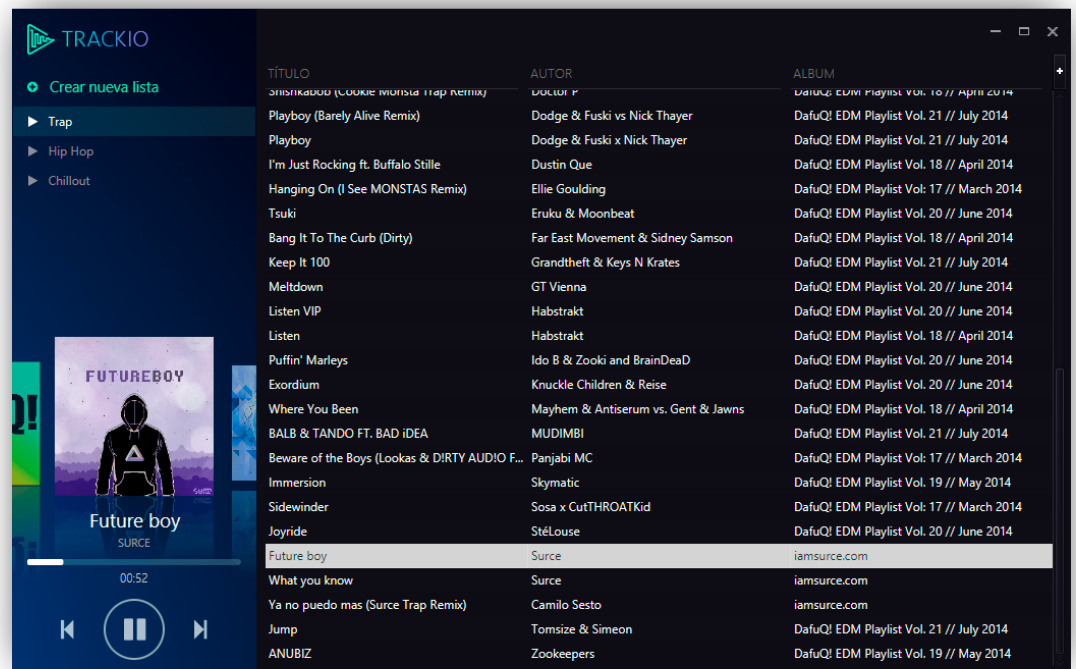
com.statickidz.trackio.view.TrackListEditDialogController: Muestra un diálogo y asocia eventos sobre TrackListEditDialog.fxml para la creación de una vista.

com.statickidz.trackio.view.css.Trackio: El archivo principal de CSS para aplicar estilos sobre toda la aplicación.

Características

- Listas basadas en carpetas con música MP3.
- Visualización de título, autor, álbum y año.
- Posibilidad de adelantar la canción o atrasarla.
- Visualización del tiempo de la canción.
- Visualización de carátulas, tanto de la canción actual, como de la anterior y la siguiente.
- Crear y editar listas (no funciona muy fino el editar y borrar, hay que invertir más tiempo)
- Animación de carátulas al cambiar de canción.

Capturas



Snippets útiles

Recorrer un directorio y obtener los archivos mp3 que contiene

```
List<String> SUPPORTED_FILE_EXTENSIONS = Arrays.asList(".mp3");
File dir = new File("C:\\UnSitio\\En\\Mi\\Pc");

for (String file : dir.listFiles((File dir1, String name) -> {
    for (String ext : SUPPORTED_FILE_EXTENSIONS) {
        if (name.endsWith(ext)) {
            return true;
        }
    }
    return false;
}))
{
    String fileURL = dir + "\\ " + file;
    //Hacer algo
}
```


Guardar algo en las preferencias de Java y recuperarlo

```
Preferences prefs = Preferences.userRoot().node(TRACKLIST_NODE);

//Guardar un String y un int en Las preferencias
prefs.put("TEST_STRING", "Mi cadena de texto");

//Extraer algo de Las preferencias
String extraido = prefs.get("TEST_STRING"); //Mi cadena de texto
```