



Instituto Tecnológico de Costa Rica

Administración de Tecnologías de
Información

Algoritmos y Estructura de Datos

TI-2402

Tarea Programada 1

Integrantes:

Melvin Brenes Gómez - 2013104822

Max Rodríguez Flores - 2014086106

Kevin Matamoros Serrano - 2014086158

Jueves 4 de Septiembre de 2014

Tabla de Contenidos

- Tabla de contenidos
- Descripción del problema
- Diseño del programa: decisiones de diseño, algoritmos usados, diagramas lógicos
- Librerías usadas: especificación de las librerías usadas, y descripción de uso
- Análisis de resultados: objetivos alcanzados, objetivos no alcanzados, y razones por las cuales no se alcanzaron los objetivos
- Manual de usuario: instrucciones uso, de ejecución y de compilación
- Conclusión personal

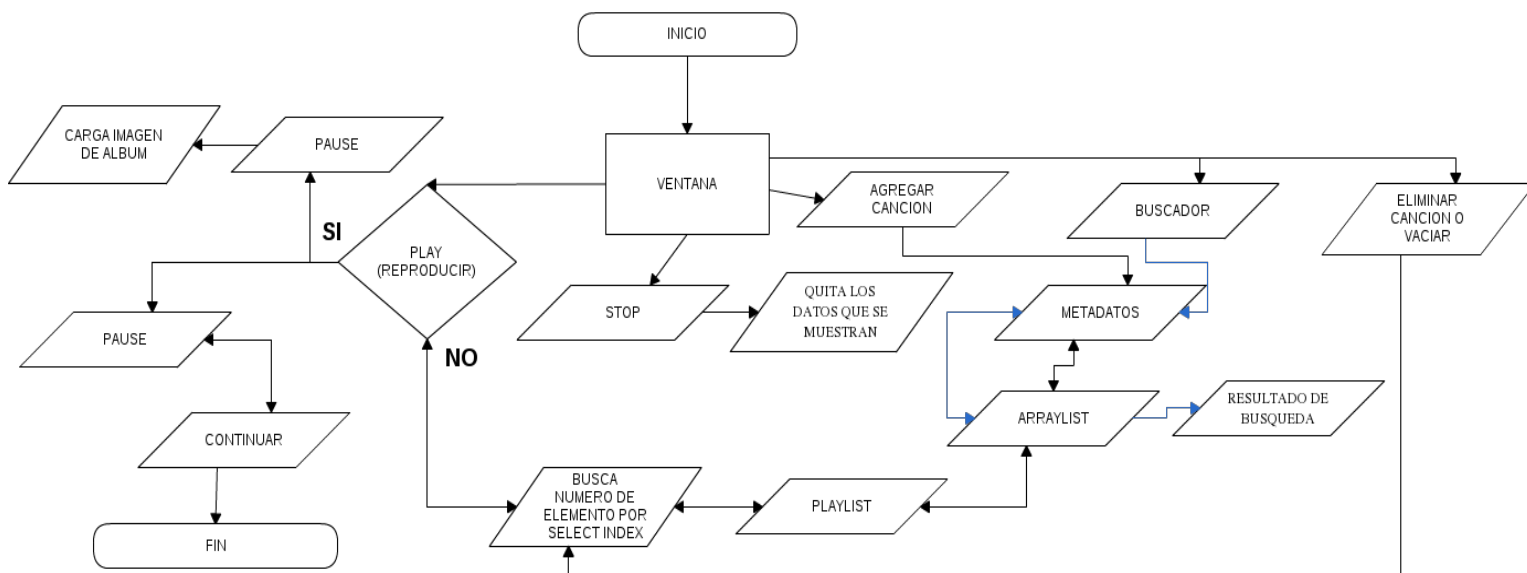
Descripción del problema

Lo solicitado es un programa que funcione de manera similar a iTunes o cualquier reproductor de música en general, capaz de darle soporte por lo menos a archivos MP3, las funciones básicas solicitadas son, Reproducir música(el usuario podrá elegir cualquier canción), Manejo de información de canciones(cada canción deberá enseñar nombre, artista, álbum, género, duración, portada del álbum), Gestión de la biblioteca de música(agregar, modificar, eliminar y consultar canciones), Búsqueda avanzada de canciones(por artista, álbum, género, o nombre). El programa tiene q ser escrito en java y deberá correr en Linux.

Diseño del programa

En cuanto a decisiones del diseño del programa se usó el color negro de fondo pues pensamos que así se veía mejor, se intentó comprimir lo más posible para que no se vea muy grande ni muy abrumador, en cuanto a los algoritmos usados todos fueron simples, por ejemplo el de cambiar metadatos obtiene el dato introducido lo cambia por el metadato solicitado y obtiene los demás del archivo. O el de reproducir, lee el dato de la lista que crea la playlist y de este extrae la ruta y reproduce el archivo, aparte de extraer los metadatos e imprimirlos en el programa. Para facilitar el diseño gráfico decidimos la utilización del programa Netbeans que nos da opciones de agregar el objeto y luego programarlo.

HayChef-Tunes - Tarea Programada #1



Librerías usadas

- **jaudiotagger-2.0.4:** Librería para parsear metadatos de un mp3, se usó para saber la duración de los archivos mp3.
- **jmf:** usada para abrir archivos multimedia como imágenes o canciones
- **jid3lib-0.5.4:** Librería para parsear los metadatos de un mp3, se usó para obtener los metadatos de los archivos mp3.
- **basicplayer 3.0:** Se utiliza para la reproducción de canciones (play, pause, stop), la misma utiliza las siguientes librerías para poder funcionar correctamente:
 - **j11.0.1:** librería JLayer, que es capaz de decodificar, convertir y reproducir archivos MP3 en tiempo real
 - **mp3spi1.9.5:** encargada de poder cargar los archivos de formatos mp3
 - **org-apache-commons-logging:** registro de información en tiempo de ejecución
 - **tritonus_share:** ofrece las clases de base y de servicios públicos para los plug-ins de aplicación Tritonus Java Sound.

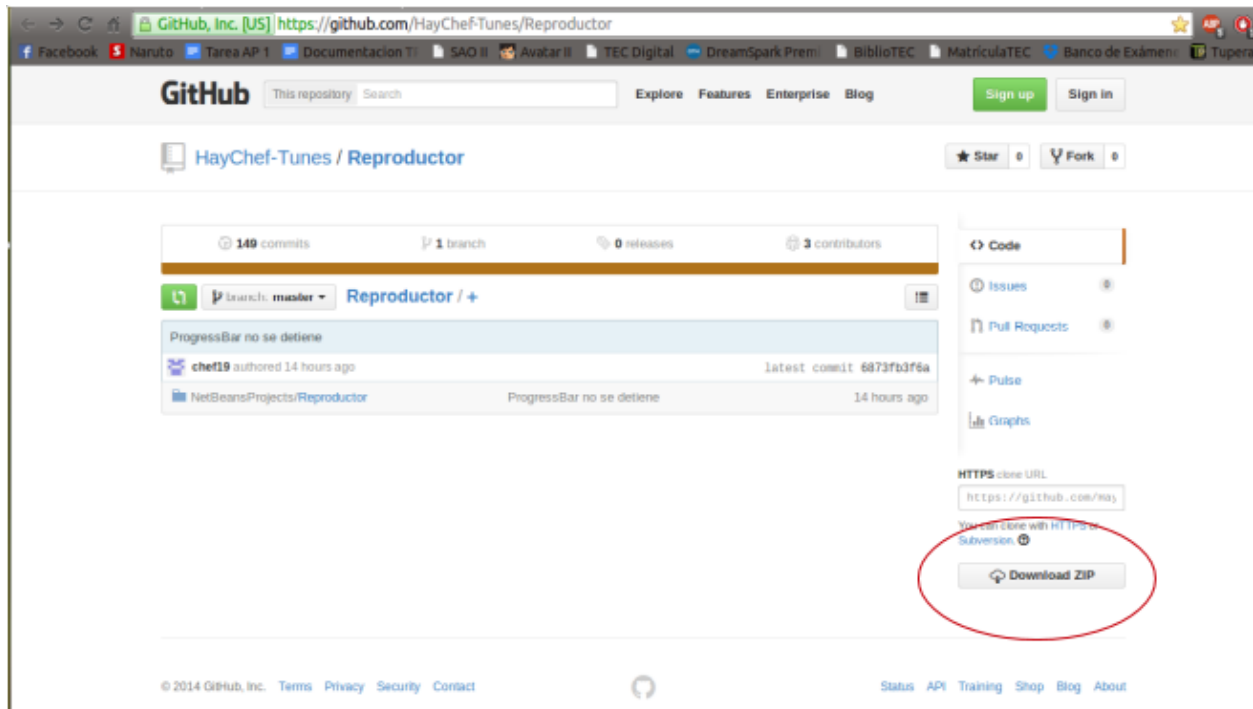
Análisis de Resultados

Las metas alcanzadas en este proyecto fueron completas, logrando implementar que el reproductor abriera la canción, extraiga los datos para mostrarlos en una caja de texto la cual no puede ser modificada, además se extrajo la imagen con un bufferedimage.

Se implementó la playlist con ArrayList para que apareciera el título de las canciones que se van agregando se utilizó un objeto de lista y se le agrego el nombre; de la misma ArrayList se usó para la búsqueda avanzada. Como extra se creó la progressbar que muestra el avance de la canción y el reloj que determina la duración total de la canción y el tiempo que lleva actualmente.

Manual de Usuario

En el siguiente enlace, se encuentra el repositorio en donde esta almacenada la programación, [https://github.com/HayChef-Tunes/Reproductor.git](https://github.com/HayChef-Tunes/Reproductor)



El botón que se encuentra encerrado en un círculo rojo generará un clon del repositorio en el escritorio y creará un archivo de formato zip, este archivo se descomprime para al final tener una carpeta en el escritorio llamada “Reproductor - master”. Una vez realizado este paso, se procederá, desde la línea de comandos, a acceder a esta carpeta hasta donde se encuentra el archivo “build.xml”, el cual compila el ANT.

```
mell@MellxD: ~/Escritorio/Reproductor-master/NetBeansProjects/Reproductor
mell@MellxD:~$ cd Escritorio/Reproductor-master/NetBeansProjects/Reproductor/
mell@MellxD:~/Escritorio/Reproductor-master/NetBeansProjects/Reproductor$ ls
build  build.xml  dist  manifest.mf  nbproject  src
mell@MellxD:~/Escritorio/Reproductor-master/NetBeansProjects/Reproductor$
```

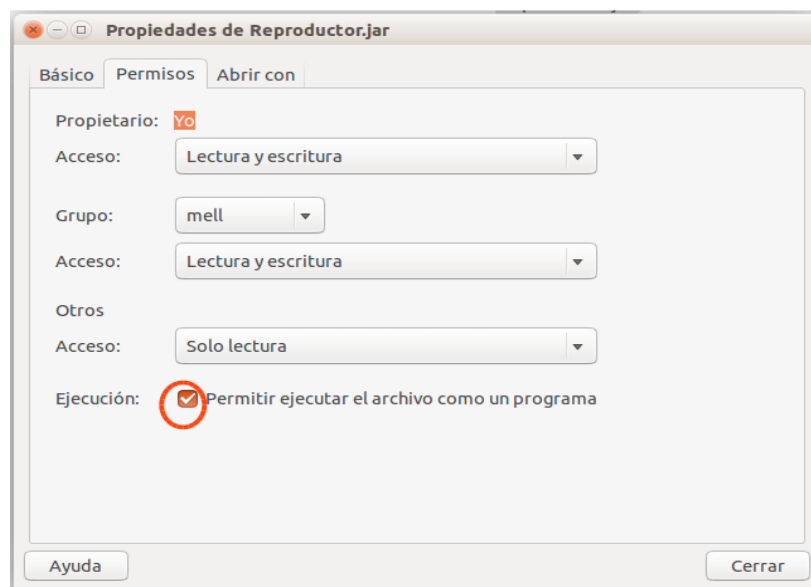
Después se procede a realizar la compilación con ant, así que en esa misma dirección se procede a escribir “ant” y luego teclear “enter”. Al momento se podrá observar en la consola lo siguiente:

```
mell@MellxD: ~/Escritorio/Reproductor-master/NetBeansProjects/Reproductor
[javadoc] ^
[javadoc] /home/mell/Escritorio/Reproductor-master/NetBeansProjects/Reproduct
r/src/Reproductor.java:54: warning: no description for @throws
[javadoc] * @throws Exception
[javadoc] ^
[javadoc] /home/mell/Escritorio/Reproductor-master/NetBeansProjects/Reproduct
r/src/Reproductor.java:61: warning: no description for @param
[javadoc] * @param args
[javadoc] ^
[javadoc] Building index for all the packages and classes...
[javadoc] Building index for all classes...
[javadoc] Generating /home/mell/Escritorio/Reproductor-master/NetBeansProjects
/Reproductor/dist/javadoc/help-doc.html...
[javadoc] 18 warnings

C-javadoc-browse:
javadoc:
default:

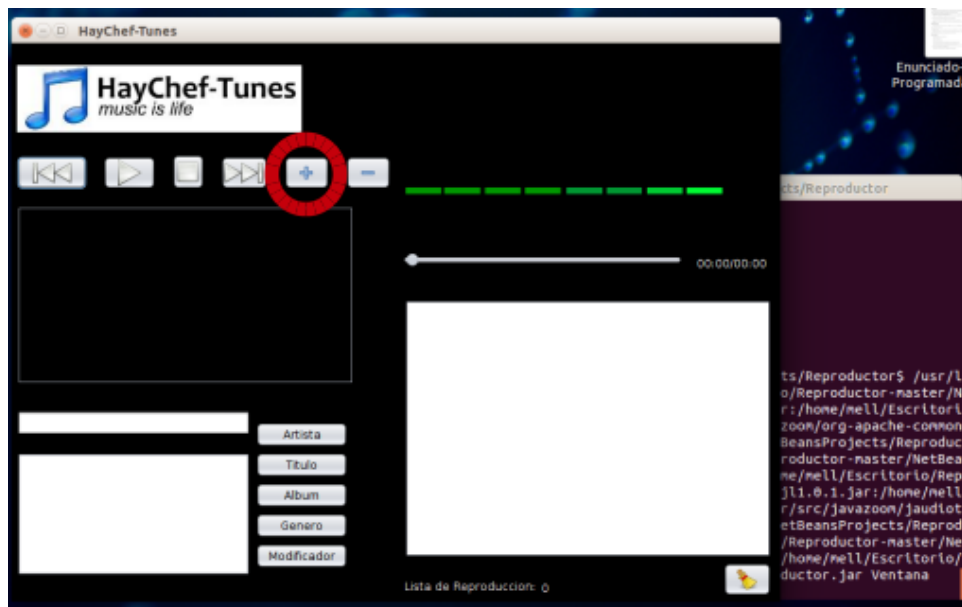
BUILD SUCCESSFUL
Total time: 6 seconds
mell@MellxD:~/Escritorio/Reproductor-master/NetBeansProjects/Reproductor$
```

Cuando se realiza este paso, en la carpeta donde se encuentra el “build.xml” se generan dos carpetas creadas por “ant”, la carpeta “build” y “dist”. En esta última encontramos el Reproductor.jar, este archivo es la programación. El mismo puede llamarse desde la consola, ingresando en la dirección en que se encuentra por ejemplo “/home/mell/Reproductor/NetBeansProjects/Reproductor/dist” y luego correrlo con el siguiente comando: “java -jar Reproductor.jar”. Pero también, se puede correr el programa al dar doble click al archivo Reproductor.jar; siempre y cuando tenga el permiso para ejecutarse, como se muestra en la imagen.

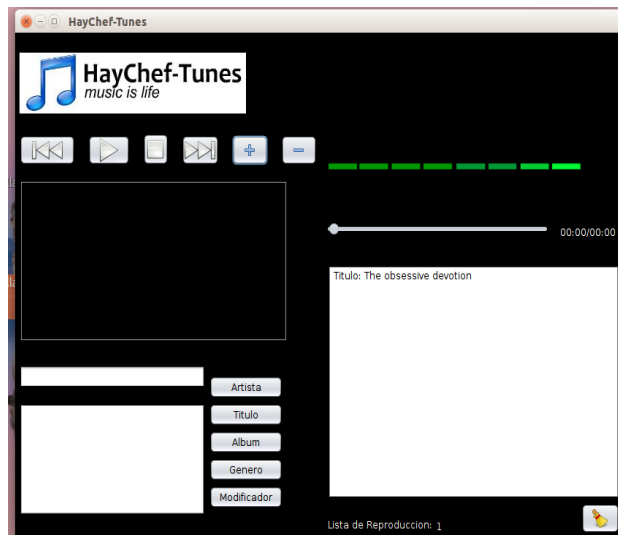


Una vez iniciado se podrá agregar canciones en formato mp3 y reproducirlas. A continuación se muestra la ventana del programa abierto.

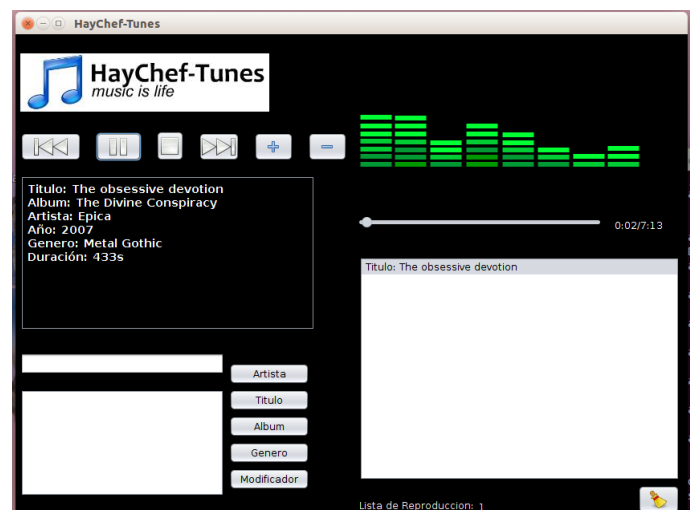
Para poder empezar a usar el reproductor, como primer paso se agregan las canciones en el símbolo del “más (+)”, encerrado en el círculo rojo en la siguiente imagen. Al dar click, abrirá un buscador en donde se podrá agregar el mp3 deseado.



Ya seleccionado el mp3, se agrega en el espacio en blanco de la derecha del reproductor, el cual es la “Playlist” y para reproducir la canción hay que seleccionarla y darle click al botón “Play (▶)”. En la siguiente imagen, al lado izquierdo, se muestra cuando la canción es agregada pero no se da click a reproducir, y al lado derecho se selecciona la canción y se reproduce al dar click en “Play (▶)”.



Canción no seleccionada



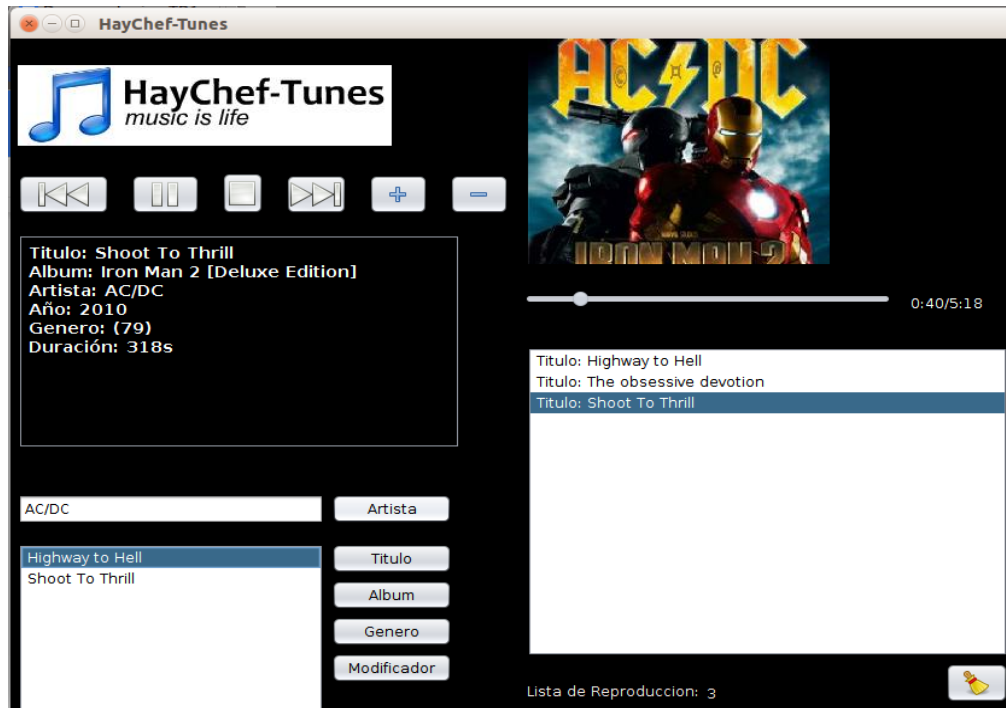
Canción seleccionada y reproduciendo

Si se desea agregar una canción más realizamos el paso anterior y se agregara en la “Playlist”. Para detener, pausar o reanudar la canción, primero se debe de seleccionar la canción en la “Playlist” y se dará click en el botón deseado:

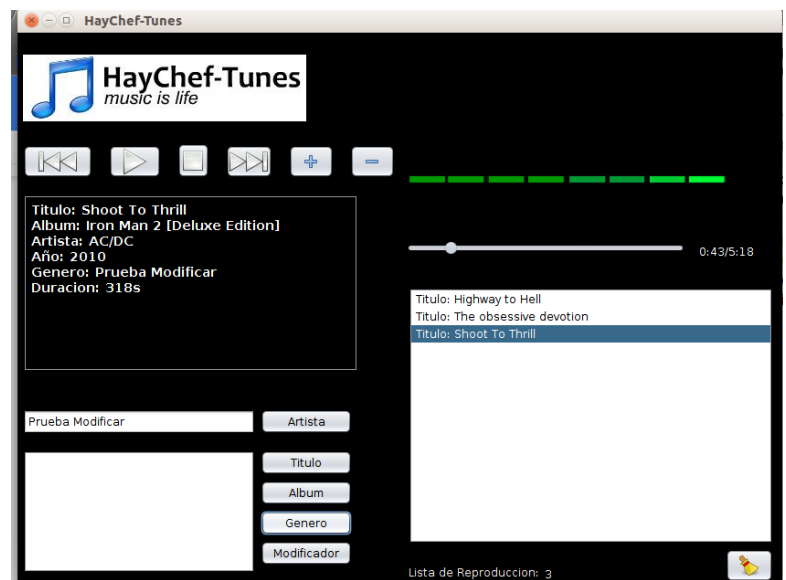
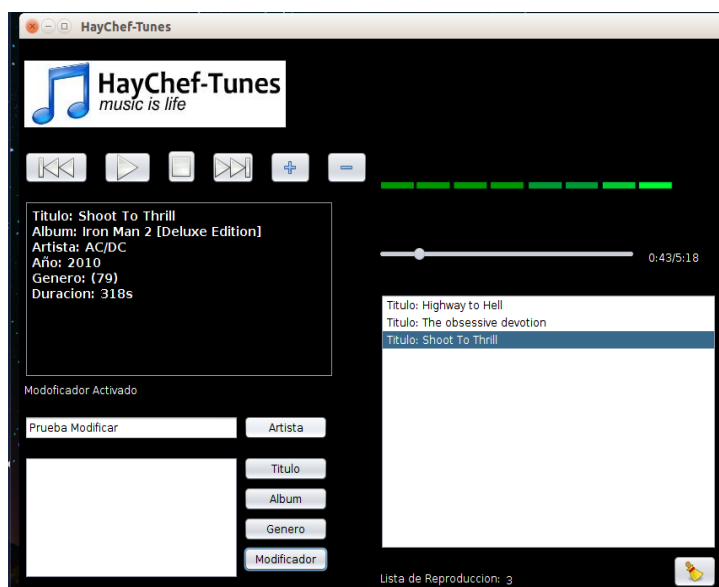
- ★ Play (▶): reproduce la canción seleccionada en la “Playlist”. Cuando esté reproduciendo la canción su símbolo cambiará al de “Pause (⏸)”.
- ★ Pause (⏸): detiene momentaneamente la canción que se está reproduciendo, cuando se de click la canción se pone en pausa y el símbolo del botón cambia al del “Play (▶)”, así después al dar click en “Play (▶)” la canción reanudará en el momento que se detuvo.
- ★ Stop (■): detiene la canción por completo, y si se da click al botón “Play (▶)”, la canción seleccionada en la “Playlist” empezará desde cero.
- ★ Next (▶▶): detiene la canción que se reproduce actualmente y pasa a reproducir la siguiente canción en la “Playlist”.
- ★ Previous (◀◀): detiene la canción que se reproduce actualmente y pasa a reproducir la anterior canción en la “Playlist”.
- ★ Agregar (+): al dar click se abre el explorador de archivos y se selecciona la canción que se desea agregar a la “Playlist” para reproducirla.
- ★ Eliminar (-): se selecciona la canción en la “Playlist” para eliminarla de la “Playlist”.
- ★ Eliminar Playlist (esquina inferior derecha, símbolo de escoba): todas las canciones de la “Playlist” son removidas.

El de cuadro de color negro con marco blanco, al lado izquierdo de la ventana del reproductor, es en donde se ubican los metadatos del mp3 que se reproduce actualmente. Abajo de este cuadro, aparece un pequeño cuadro de texto en donde se podrá buscar una canción ya sea por título, artista, género, o álbum. En la siguiente imagen se observa un ejemplo.

Se puede observar que en cuadro inferior izquierdo, aparecerá el título de la canción que posea en los metadatos, el mismo valor que se digita en el cuadro de texto. Para buscar se ingresa lo deseado en el cuadro de texto y luego se dará click en la categoría que se desea buscar (título, artista, género, o álbum).



Para modificar un metadato de una canción, se utiliza el botón “Modificador”, al hacer click se habilitan en los botones de “Título”, “Artista”, “Género”, y “Álbum”, la función para modificar los metadatos según lo deseado. Además, en la pantalla aparecerá lo siguiente “Modificador Activado”, indicando así que se encuentra en ese modo. Así que, primeramente daremos click en el botón “Modificador”, escribimos lo que deseamos cambiar en el metadato, en el mismo cuadro de texto en donde buscamos, y le daremos click al botón a la categoría correspondiente a cambiar. Y listo, el cambio aparecerá en pantalla como se muestra a continuación. Al lado izquierdo, en modo modificar y al derecho el cambio guardado.



Conclusión

Como conclusión, adquirimos conocimientos amplios en el lenguaje de programación Java, el cual es nuevo para nosotros, ya que no teníamos experiencia en ese lenguaje. Además, al utilizar el IDE de Netbeans para facilitar la parte del diseño del programa, aprendimos sobre cómo funciona el mismo y la facilidad que genera al programador con lo referente al diseño de la interfaz gráfica.

El trabajo fue un poco difícil, de igual forma, también se presentaron ciertos métodos los cuales resultaron fáciles ya que cada vez tenemos más conocimiento, conforme avanzábamos con la tarea programada. En la parte de lista de reproducción es donde principalmente se usaron los conocimientos adquiridos en el curso, se implementó una pequeña función extra de modificar metadatos específicos, fue un trabajo en equipo donde cada uno contribuyó al código y nos ayudabamos mutuamente con nuestros problemas.

También obtuvimos conocimientos sobre git, github y el uso de repositorios para proyectos, estos son herramientas muy útiles pues simplifican mucho la organización del proyecto, es mas facil salir de dudas con los compañeros de equipo y el ponerse de acuerdo respecto a las funciones de cada miembro del grupo aparte de que mantiene un orden respecto a quien subió que, cuando e incluso se pueden devolver versiones por si algo salio mal y se arruino el código.