Instituto Tecnológico de Costa Rica.

Administración de Tecnologías de Información.

Algoritmos y Estructuras de Datos TI-2402

Prof: Andréi Fuentes L.

TAREA PROGRAMADA I

* Kenneth Fallas. 2014137252
* Daniela Solís. 2014129963
* Jorge Mora. 2014159921

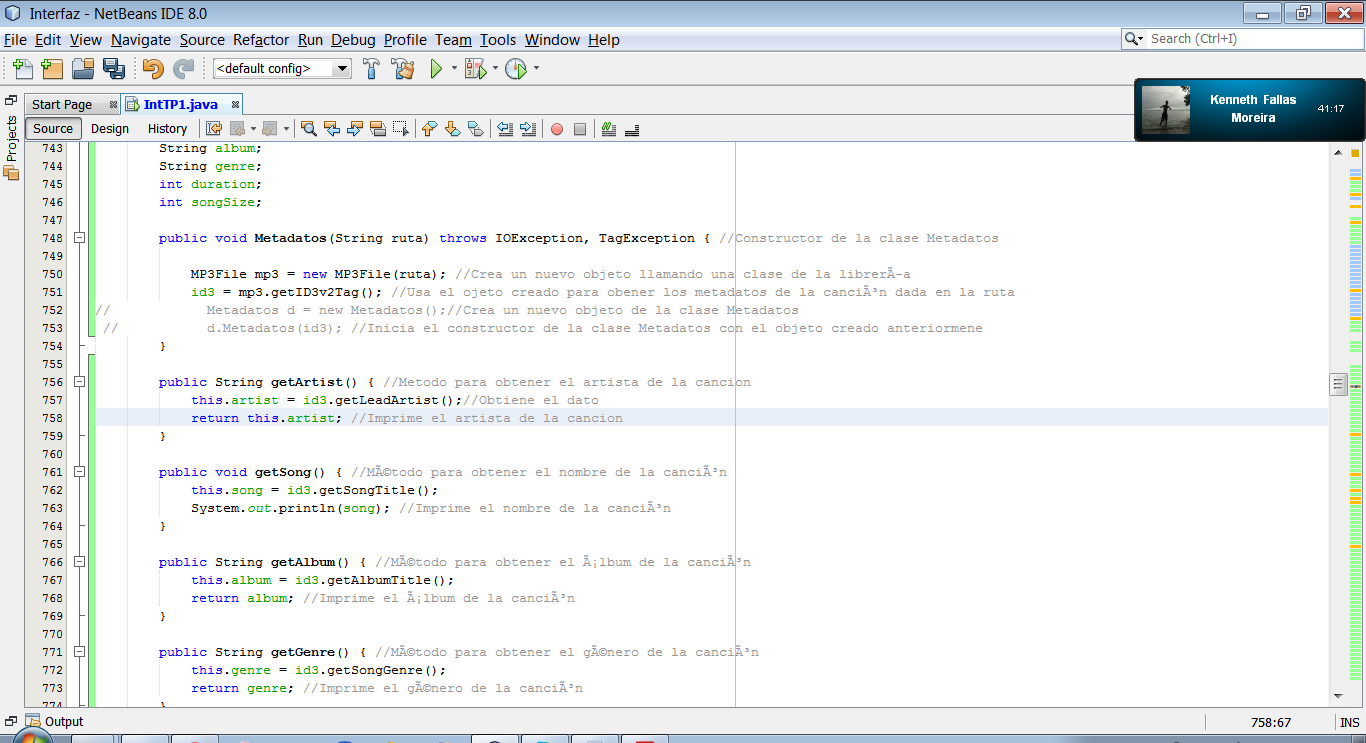
Tabla de contenidos

|  |
| --- |
| 1. Descripción del problema. |
| 1. Diseño del programa. |
| 1. Librerías utilizadas. |
| 1. Análisis de resultados |
| 1. Manual de usuario |
| 1. Conclusión personal |

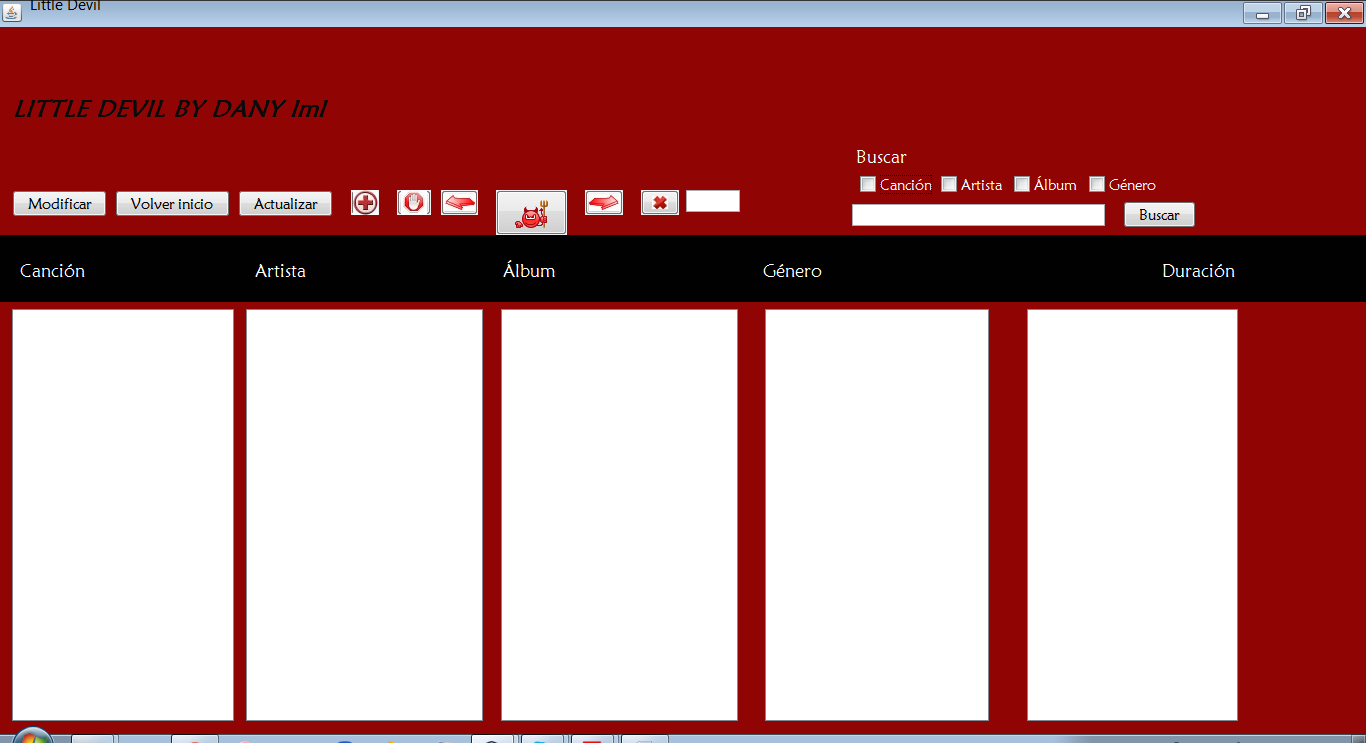
1. **Descripción del problema**

Se solicitaba un programa que fuera capaz de reproducir música en formato mp3, que permita gestionar la biblioteca; agregando o eliminando canciones. Se contaría con un espacio de búsqueda en donde el usuario pueda escribir el artista, género, etc y reproduzca dicha canción. Por otra parte se requería modificar las canciones por medio de metadatos, cambiando el nombre de la canción, álbum, género, etc. El programa debe de estar escrito en Java y funcionar en Linux.

1. **Diseño del programa**
2. Se decidió crear la Interfaz Gráfica basándose en iTunes, con varias listas en java, que especificarán nombre, artista, álbum, género y duración de las canciones.
3. Se creó la clase metadatos que nos permite acceder a los datos específicos de cada canción; nombre, artista, duración, entre otros.
4. Se creó la clase Interfaz, la cual tiene contenida a la clase LD que es una lista doblemente enlazada. LD tiene todo lo que tenga que ver con el reproductor. De igual forma contiene la clase metadatos.
5. La clase LD contiene varios métodos que son los que permiten la búsqueda y reproducción de canciones.
6. Para extraer los metadatos, se crearon los métodos de get ( getArtist, getSong, getAlbum, getGenre, getDuration). Cada uno se creaba de esta manera:



1. Para la búsqueda se decidió usar CheckBox’s los cuales van a filtrar la información dependiendo de la elección.

****

1. Estos compararían los datos introducidos en el campo de texto con los que se tengan en los métodos de los metadatos y devolvería la o las canciones que tengan coincidencias.
2. **Librerías utilizadas.**

* Jid3lib: Proporciona funcionalidad para leer información mpeg. También permite la lectura, la escritura, y la eliminación de las etiquetas ID3v1 e ID3v2. En este caso fue utilizada para los métodos get que nos devuelven los metadatos de cada canción.
* BasicPlayer3.0: está diseñada para utilizarse en cualquier aplicación que necesite características simples (reproducir, detener, pausar, reanudar, buscar) para reproducir el archivo de audio. En este caso se utilizo para crear todos los métodos de reproducción. Contiene varias librerías dentro de ella que ayudan a conseguir esas funciones y aparte tienen todo el manejo del sonido. Como las siguientes:
* Tritonus\_share.jar: ofrece las clases de base y de servicios públicos para los plug-ins de aplicación Tritonus Java Sound.
* mp3spi1.9.4.jar: es una interfaz de proveedor de servicio de Java que añade MP3 (MPEG 1/2 / 2.5 Layer 1/2/3) soporte de formatos de audio para la plataforma Java. Se basa en las bibliotecas JLayer y Tritonus Java.
* Jl1.0.jar: es una biblioteca que decodifica / reproduce / convierte MPEG 1/2 / 2.5 Layer 1/2/3 (es decir, MP3) en tiempo real para la plataforma Java.

1. **Análisis de resultados**

**Objetivos alcanzados**.

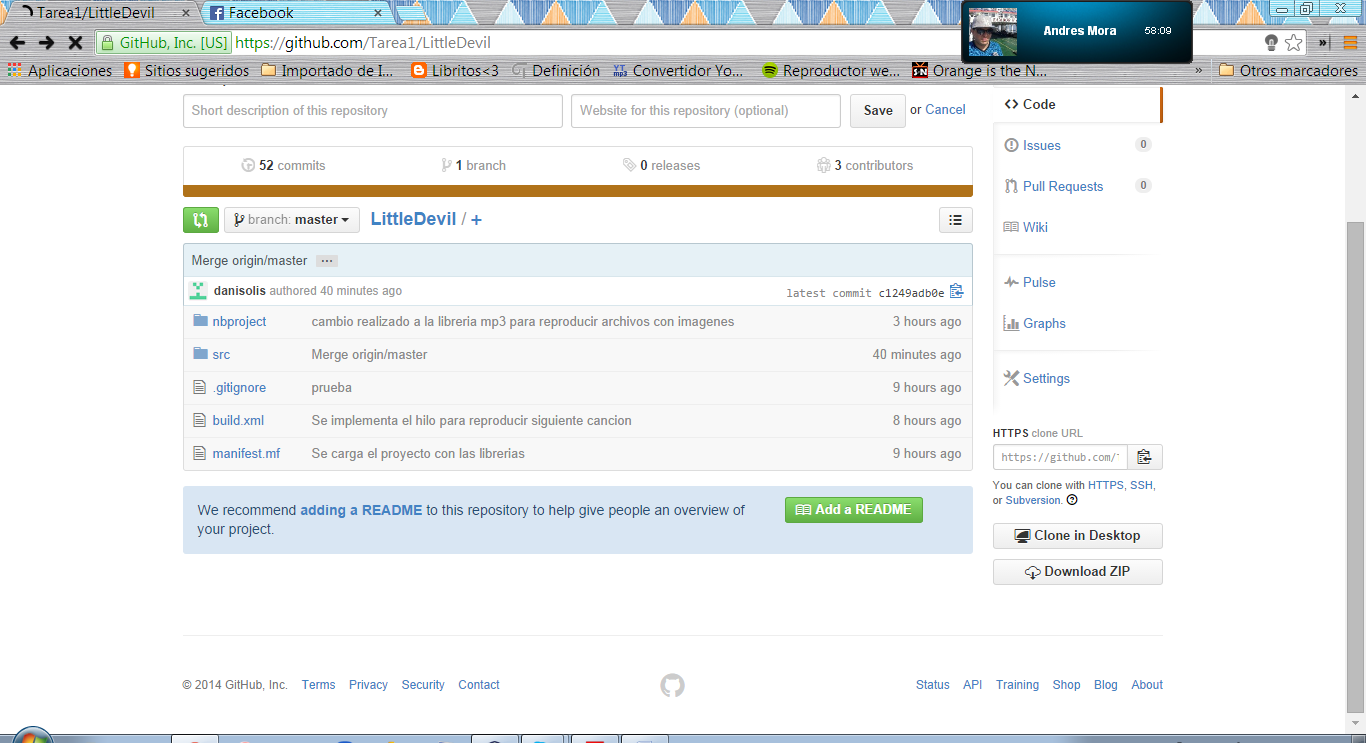
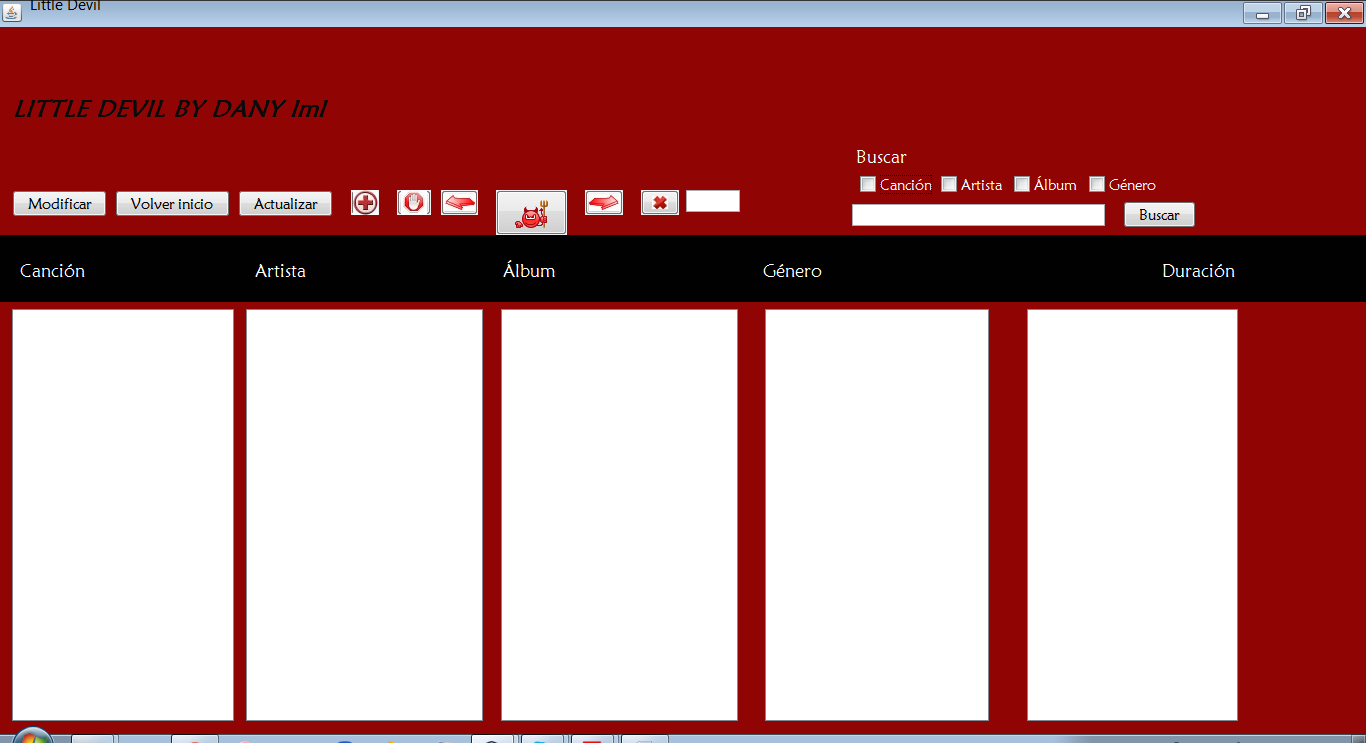
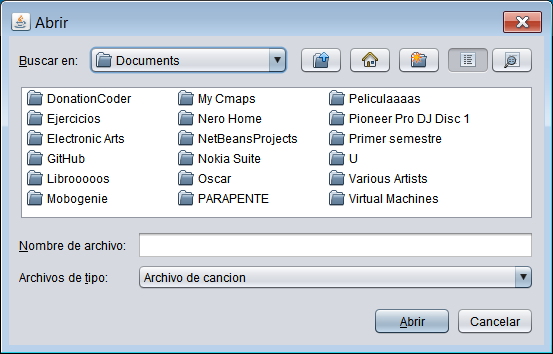
Permite agregar cualquier canción que este en formato .mp3 desde la computadora. Por otra parte se obtuvieron los metadatos de cada canción lo que nos da acceso a toda la información de cada una.

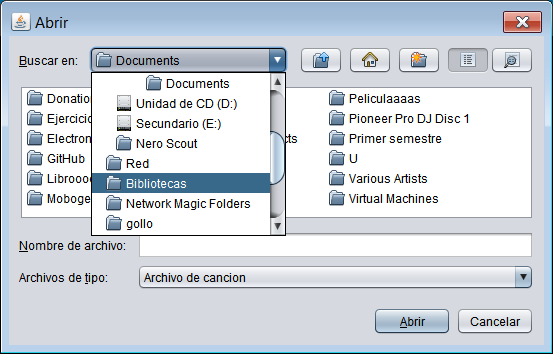
En la parte de la gestión de la biblioteca se pueden agregar y eliminar canciones .Se logró la reproducción y con ello poder avanzar a la siguiente canción o devolverse a la anterior; detenerla y volver al inicio de la lista de reproducción, y eliminarla de la lista. La lista de reproducción es continua, es decir, si llega al final de esta vuelve a reproducir la canción que estaba de primera. También imprime la canción en la pantalla, especificando canción, artista, álbum, género y duración.

Se cuenta con la búsqueda avanzada, la cual es dependiendo de la elección del usuario. Puede buscarla por canción, álbum, género etc. Esta nos imprime la canción que se ah buscado con todos los datos.

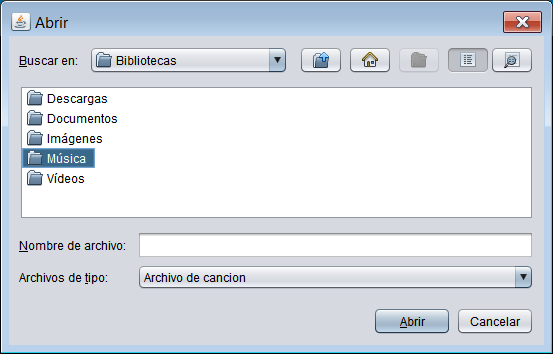
**Objetivos no alcanzados.**

Mostrar imagen de un álbum: no se puedo conseguir esa parte porque se dificultó mucho, y no se encontró una manera de hacer que funcionara.

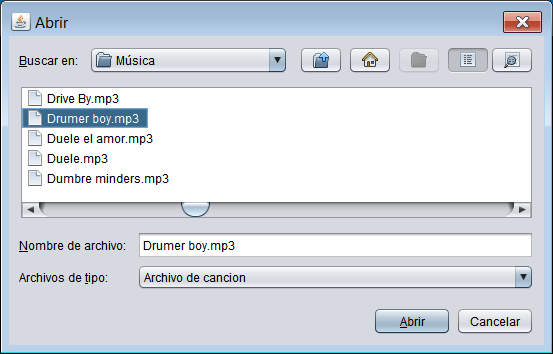
1. **Manual de usuario.**
2. En el link <https://github.com/Tarea1/LittleDevil.git> se encuentra el reproductor completo para descargar.
3. En el botón Download zip ubicado en la parte inferior de la pantalla se descarga el zip con todo lo necesario para usar el reproductor. ****}
4. Al presionar Download zip Se creará un clon del repositorio en el equipo el cual se procede a acceder a la carpeta “build.xml”, en el cual se compila el ANT.
5. Al iniciar el programa esta será la ventana principal del reproductor de música.
6. Si desea agregar una canción para reproducir, click el signo C:\Users\gollo\Desktop\agregar.jpg y aparecerá una ventana como esta:
7. Le damos click en documentos y bibliotecas:

****

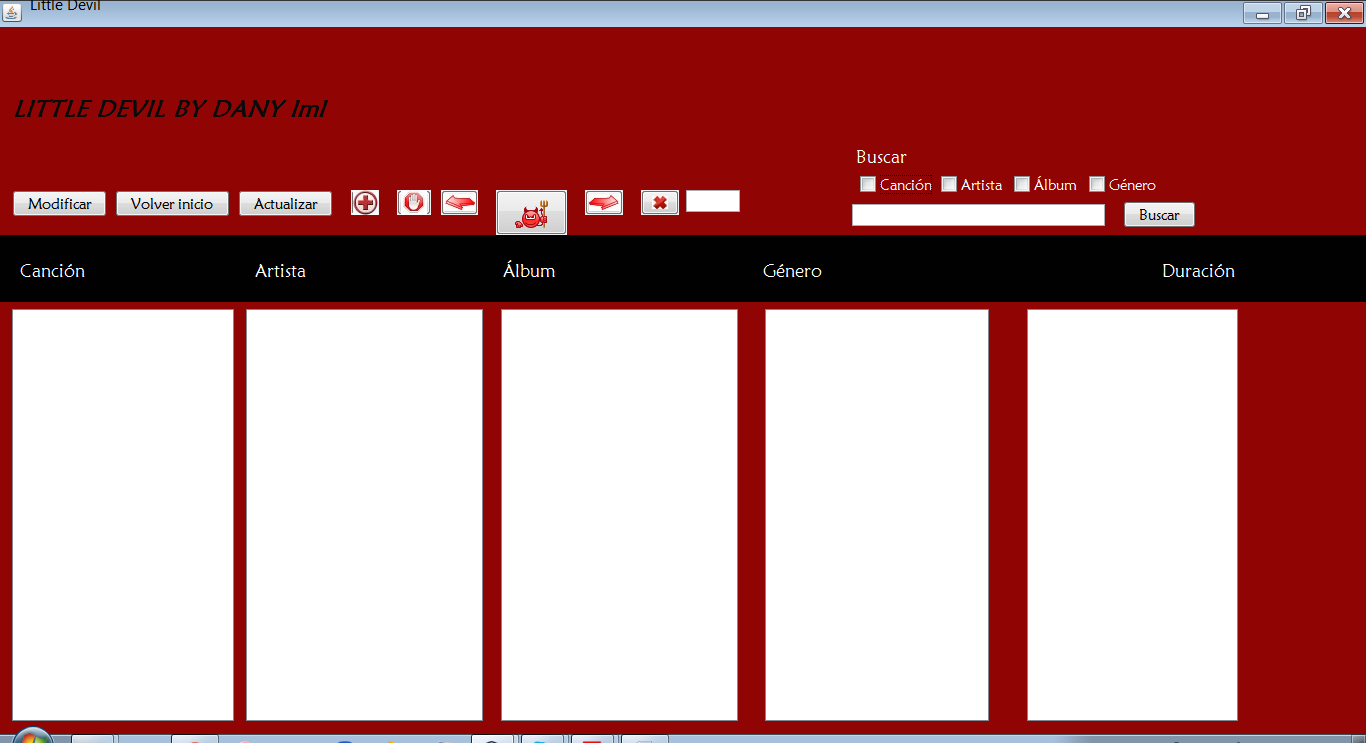
1. Y seleccionamos la carpeta música:

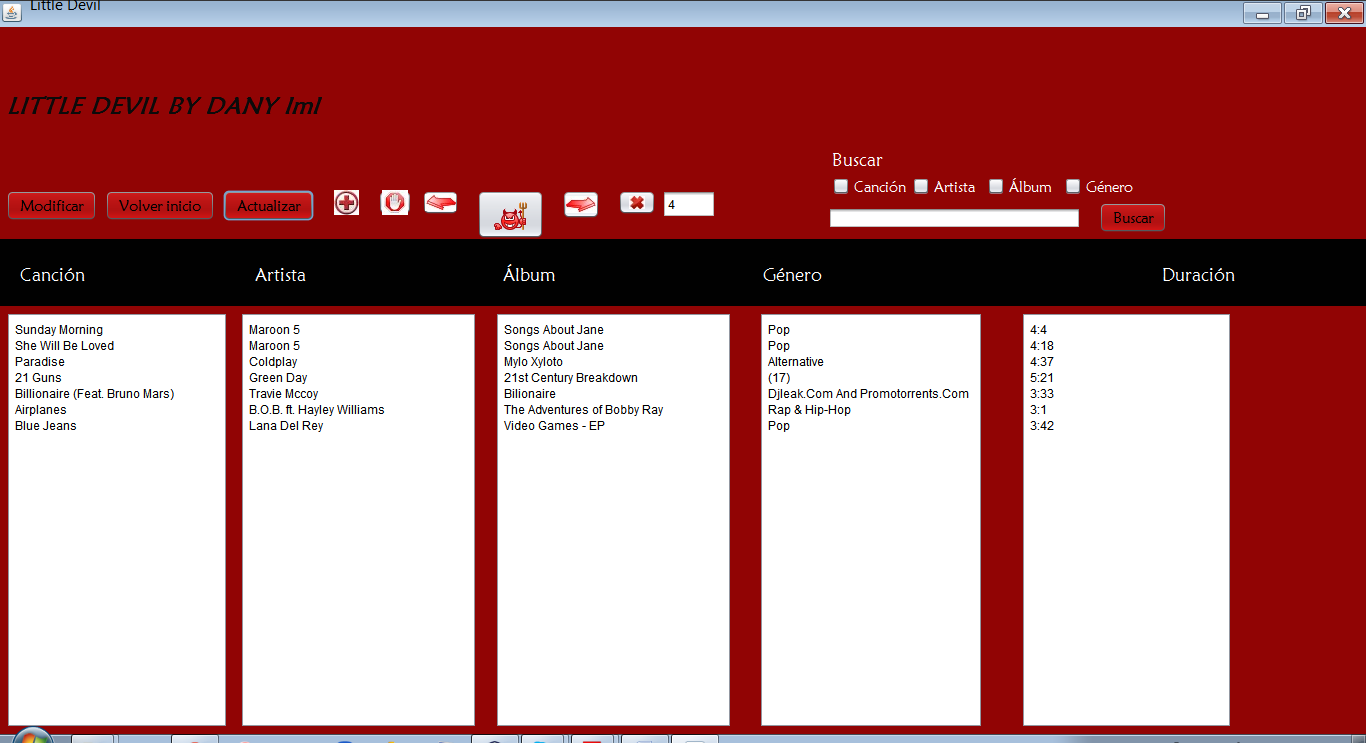


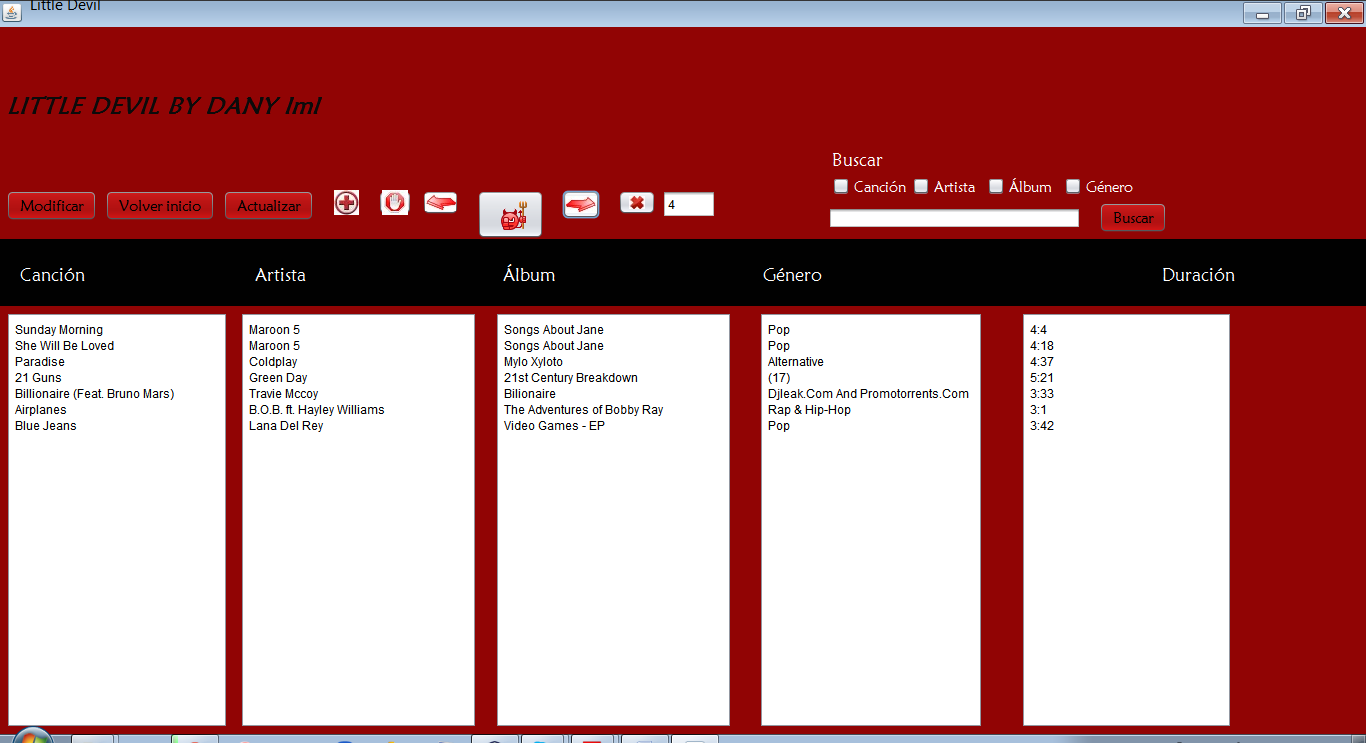
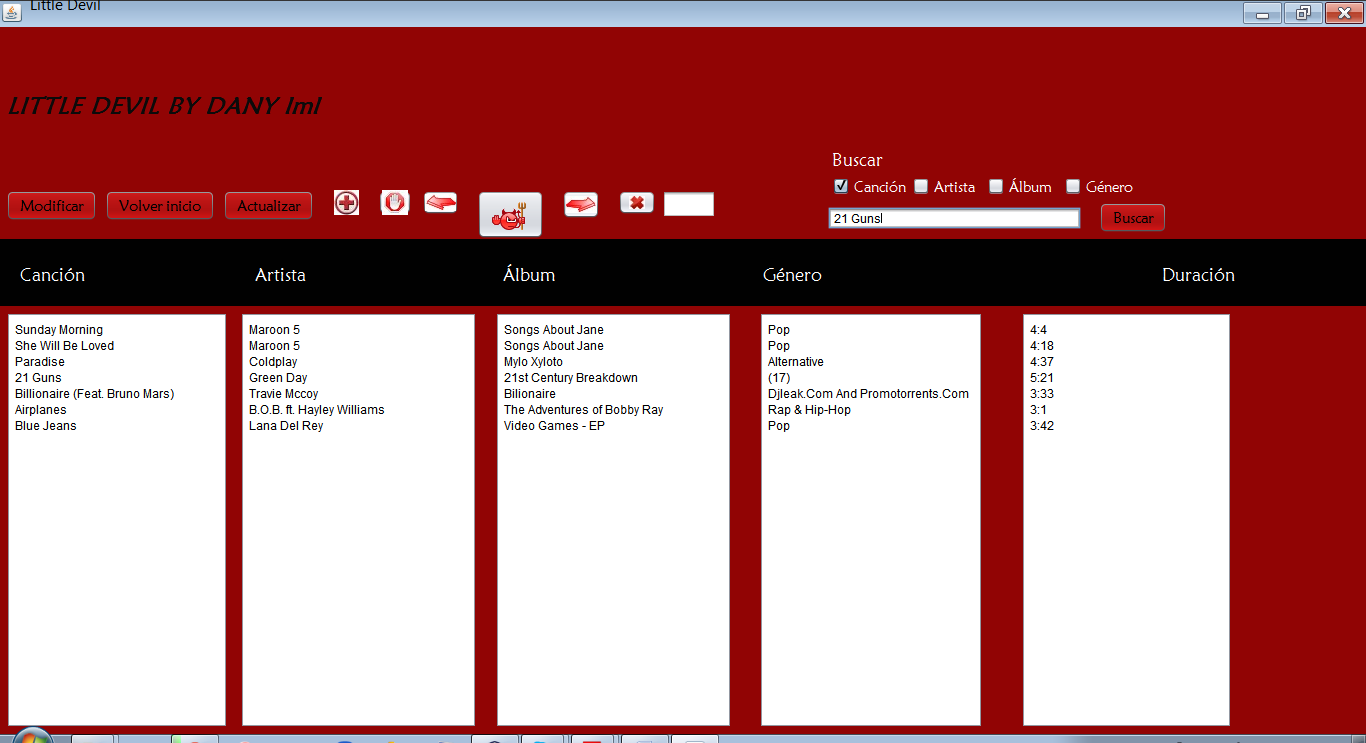
1. Luego escogemos la canción de nuestra preferencia:

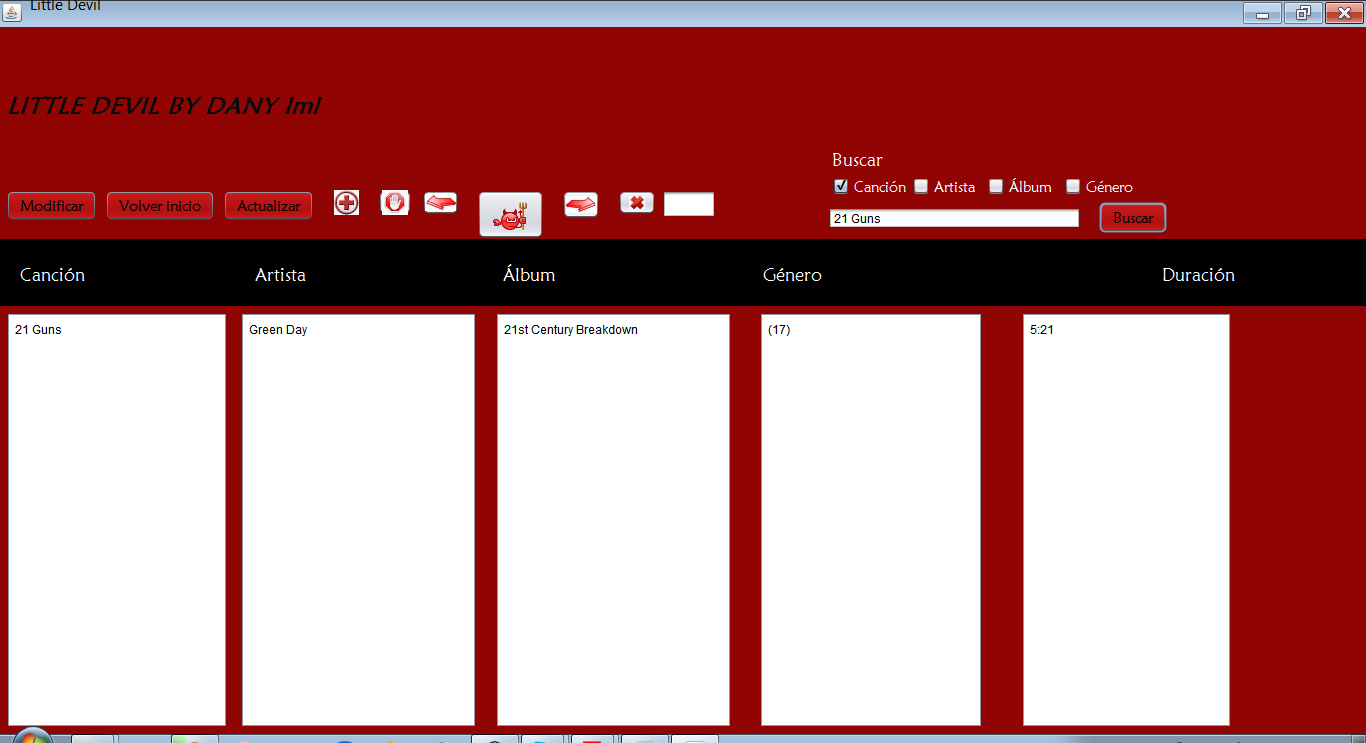


1. Ahora de esa manera se pueden agregar todas las canciones que queramos. Cuando ya tengamos listas todas las canciones que deseemos agregar le podemos dar reproducirC:\Users\gollo\Desktop\diablo.gif:



1. **** Para visualizar la lista de reproducción presionamos el botón **actualizar** el cual nos da los datos de la lista de canciones que habíamos seleccionado.
2. Para ver la siguiente canción click en C:\Users\gollo\Desktop\sig2.jpg y para ver la anterior click enC:\Users\gollo\Desktop\sig.jpg. En caso de querer detenerla click en C:\Users\gollo\Desktop\detener.jpg y para eliminar la canción escribimos el número de esta en el espacio en blanco a la par de laC:\Users\gollo\Desktop\borrar.jpg, presionamos la C:\Users\gollo\Desktop\borrar.jpg y luego actualizar. Si queremos volver a comenzar la lista de reproducción click en *Volver al inicio*.
3. Para hacer una búsqueda primero tenemos que seleccionar el filtro, es decir si queremos buscar por canción, álbum, artista o genero.



Al darle click en buscar, si la canción existe en la lista de reproducción nos la devolvería, de igual forma con todos sus datos. Funciona igualmente si seleccionamos buscar por artista, álbum o género.

1. **Conclusión personal**

En conclusión con la Tarea Programada hemos obtenido más conocimiento sobre Java que es un lenguaje bastante complejo cuando se comienza, adquirimos conocimientos sobre como importar librerías, utilizar una interfaz, ANT, github.com entre otras.

Se aprendió bastante sobre el manejo de listas y sobre como implementarlas en casos más complejos como lo fue en esta tarea con las canciones. Se uso bastante del contenido visto en clase el cual nos fue de bastante ayuda.

En general no se dificultó tanto la parte de reproducir, ir a la siguiente o anterior, agregar… Sin embargo en los métodos más avanzados como la búsqueda, la impresión de los datos en cuadros de texto separados en la pantalla nos dio más problemas, al final todos se pudieron resolver excepto mostrar la imagen de un álbum, de ahí en fuera la tarea esta completa.

Por otra parte github nos creó una serie de problemas, al eliminar ciertos archivos, y no permitirnos subir los aportes de cada uno, desde que empezamos el reproductor. Después de varios días de tener problemas logramos subir cada uno los datos pero creando un nuevo repositorio. Adquirimos mucho conocimiento sobre el uso de esta herramienta, que aunque se nos dificultó al principio resulta ser bastante útil.