Administración de Tecnologías de Información.

Prof. Andréi Fuentes.

TI-2402 Algoritmos y Estructuras de Datos.

* Kenneth Fallas.
* Daniela Solís.
* Jorge Mora.

TAREA PROGRAMADA I

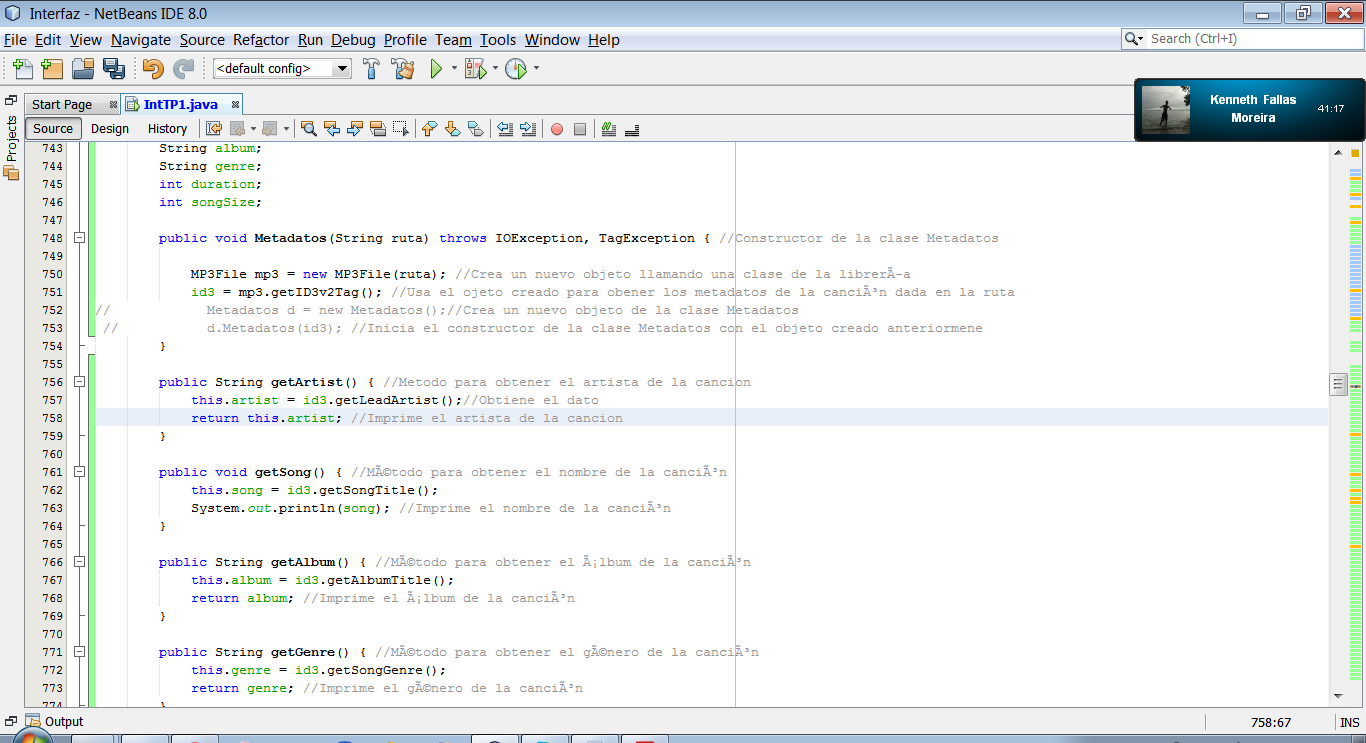
Tabla de contenidos

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

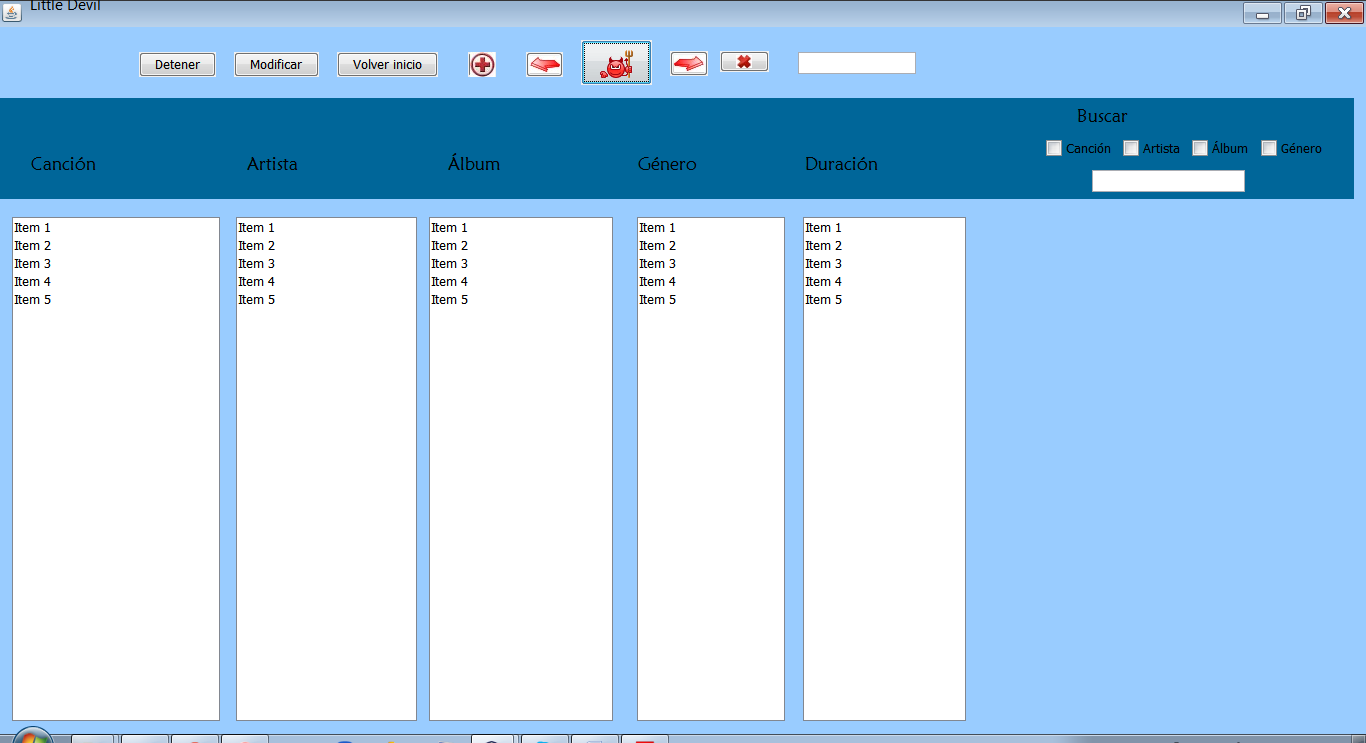
1. **Descripción del problema**

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.

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. También imprime la canción en la pantalla, especificando canción, artista, álbum, género y duración.

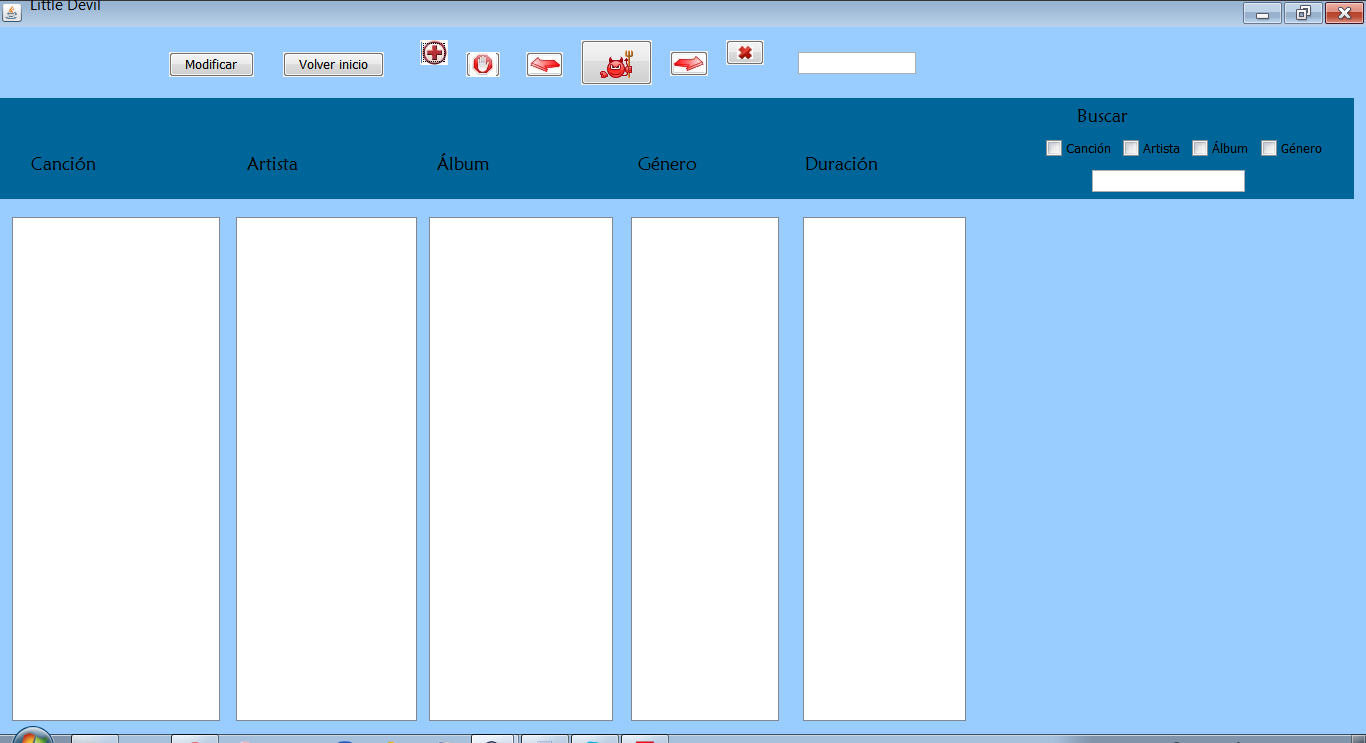
**Objetivos no alcanzados.**

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.

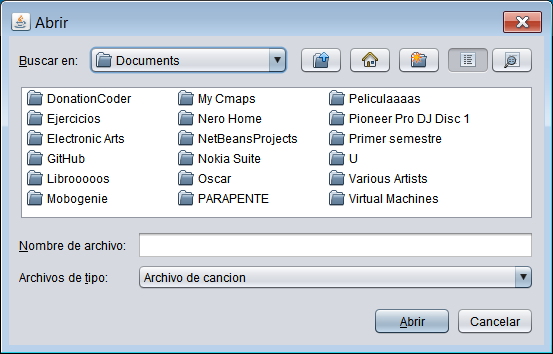
Eliminar una canción: no se encontró como adquirir un valor desde la interfaz en tiempo de ejecución.

Búsqueda avanzada: no se pudo tener esta parte porque se hizo difícil, tomar los datos a parte y compararlos.

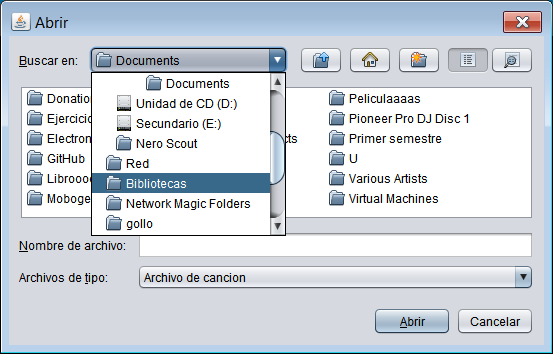
1. **Manual de usuario.**
2. Al iniciar el programa esta será la ventana principal del reproductor de música.

****

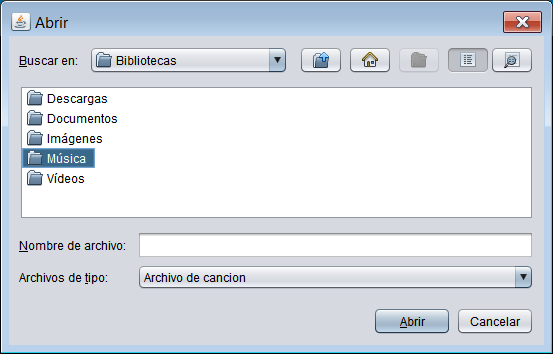
1. Si desea agregar una canción para reproducir, click el signo C:\Users\gollo\Desktop\agregar.jpg y aparecerá una ventana como esta:

****

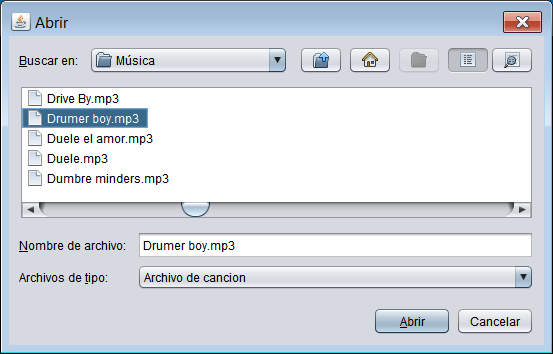
1. Le damos click en documentos y bibliotecas:

****

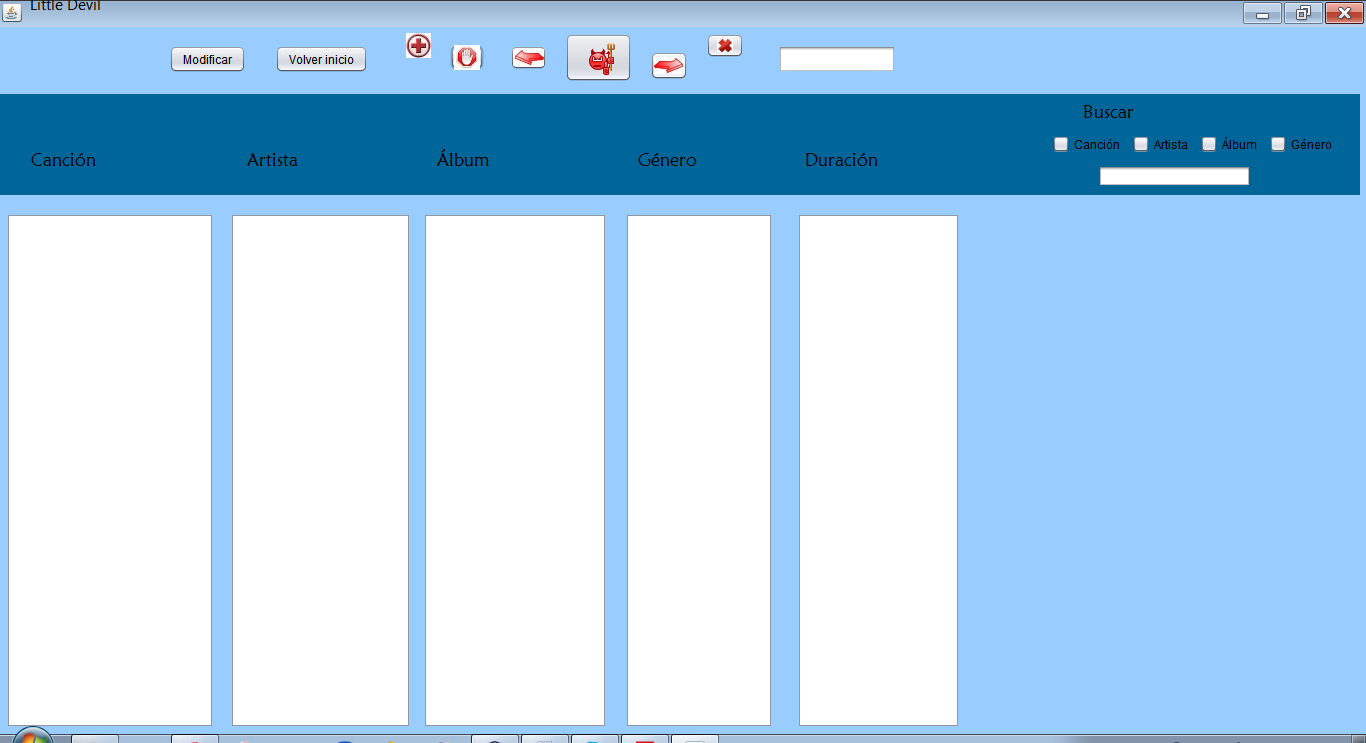
1. Y seleccionamos la carpeta música:



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 ver la siguiente canción click en C:\Users\gollo\Desktop\sig2.jpg y para ver la anterior click en C:\Users\gollo\Desktop\sig.jpg. En caso de querer detener la canción click en C:\Users\gollo\Desktop\detener.jpg y para eliminar la canción escribimos el número de la canción en el espacio en blanco a la par de laC:\Users\gollo\Desktop\borrar.jpgy luego presionamos la C:\Users\gollo\Desktop\borrar.jpg . Si queremos volver a comenzar en el inicio de la lista de reproducción click en *Volver al inicio* y para modificar click en *Modificar*.
2. **Conclusión personal**

En general la Tarea Programada no se dificulto, la parte de reproducir, reproducir siguiente, reproducir anterior, agregar, detener, etc., fue fácil de lograr. Al conectarlo a la interfaz funcionó muy bien. 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 y nos ayudo mucho. Por otro lado hubieron una serie de requisitos que no se pudieron cumplir, todo ese tipo de cosas quedan como experiencia para proyectos futuros.