Práctica 4

MEMORIA

Programació II

Eduardo Wass Rosado
NIUB 16447981
Universitat de Barcelona

Índice

- Introducción
  - Descripción de la práctica
  - Objetivos
- Análisis
  - Previsión y organización
  - Guión de trabajo
- Desarrollo
  - Clases implementadas
  - Reutilización de código de clases anteriores y adaptaciones
  - Problemas y soluciones
- Resultados
  - Pruebas realizadas
  - Ejemplos de Output
  - Conclusiones

Introducción

Descripción de la práctica

El objetivo de la cuarta práctica es crear una nueva vista, la cual ya será una interfaz gráfica compatible con la mayoría de sistemas operativos. Para ello aprovecharemos el editor de diseño de Netbeans que nos ayuda a generar los controles SWING para diseñar UI's que Java pone a nuestra disposición, así como el control de eventos para interactuar con los mismos.

La gracia es que podremos interactuar con el programa que hemos implementando anteriormente pero esta vez gracias a los controles, lo haremos de manera mucho más gráfica y por tanto más intuitiva y amigable de cara al usuario. Es decir, en esta práctica nos centraremos completamente en la VISTA de nuestro programa.

Objetivos

- Aprender a hacer uso de los controles SWING y el editor que nos facilita NetBeans
- Diseñar una interfaz gráfica para nuestro programa con la cual podamos llevar a cabo todas las funcionalidades que se especifican en el enunciado de la práctica
- Cumplir los requisitos que nos solicita el enunciado: usar JList, JFileChooser, etc.
- Programar todo la lógica de los controles para que se comporten como es debido

Análisis

Previsión y organización

Esta práctica es un poco diferente a las anteriores ya que toda la importancia recae sobre la vista del programa, así que esta vez no era todo pensar en clases y métodos, sino en mayor parte en nuestra interfaz así como sus controles y eventos.

A continuación especifico un poco más el proceso de trabajo seguido.

Guión de trabajo

- Diseñar UI sobre el papel
- Implementar el diseño a la vista con el asistente de diseño de Netbeans
- Renombrar todos los controles, ajustar su tamaño y demás propiedades
- Asegurarse que la estructura era lógica y que el asistente de diseño no había introducido demasiado código innecesario
- Programar los eventos y lógica que hay detrás de cada uno de los controles
- Probar y corregir el funcionamiento de cada uno de los controles

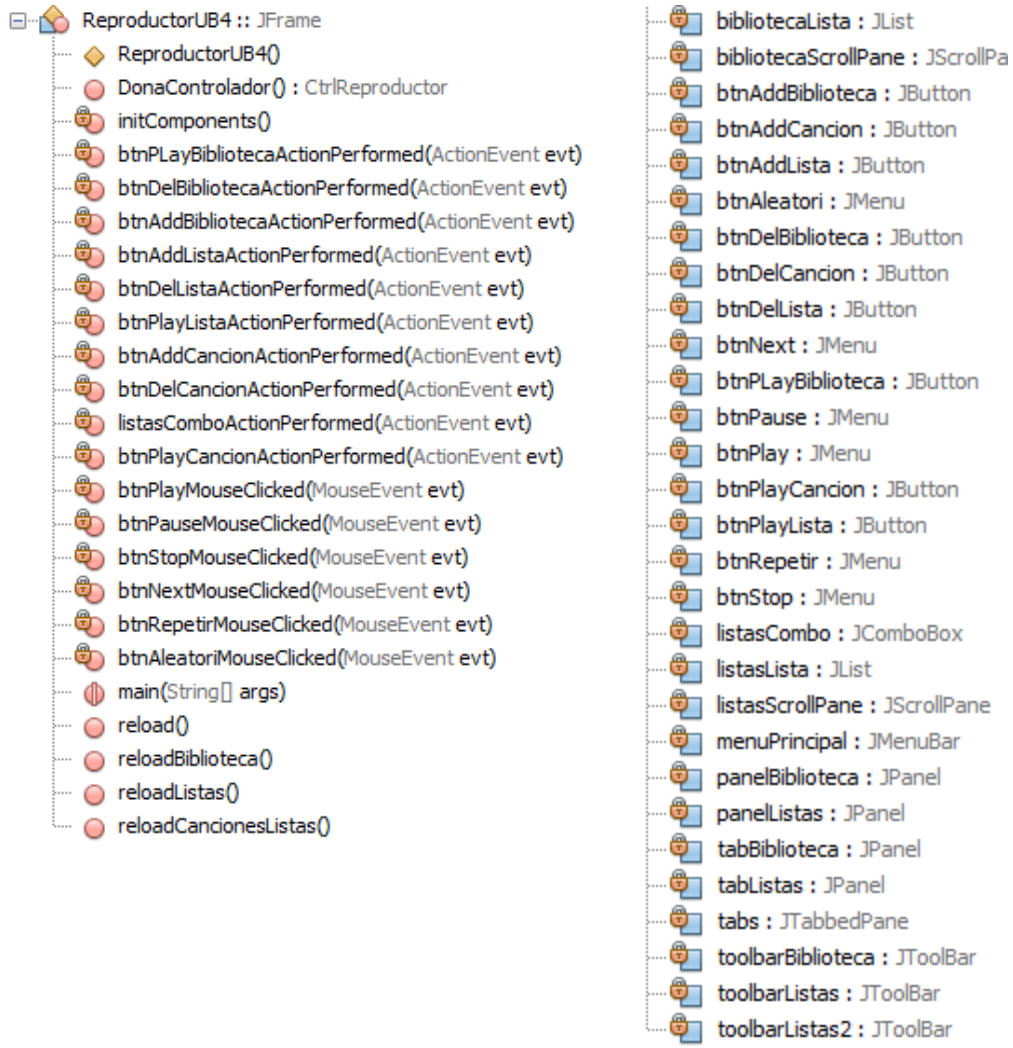
Desarrollo

Clases implementadas

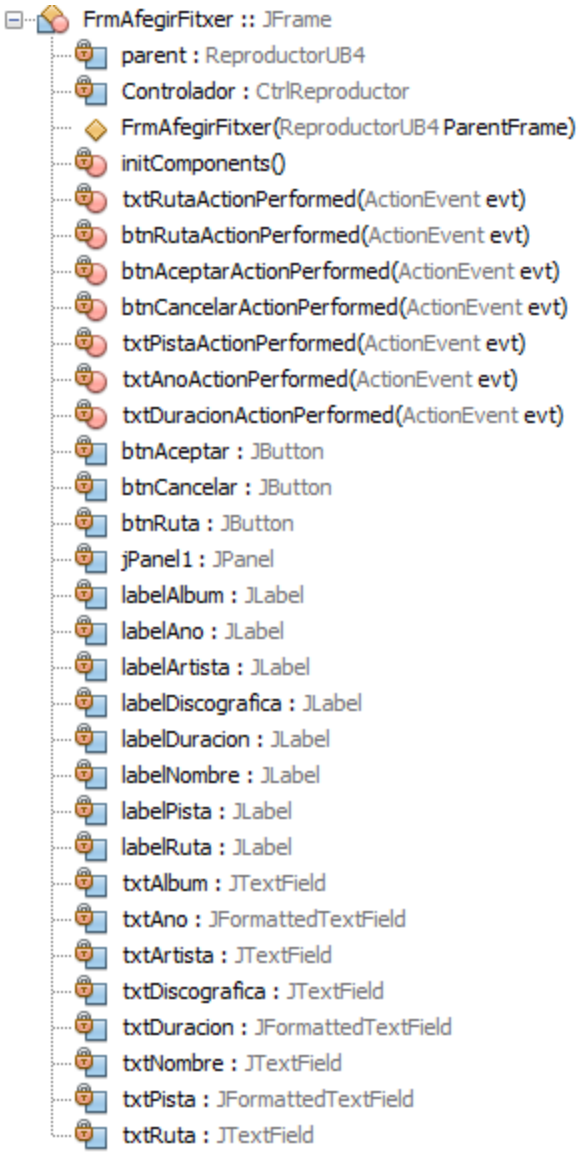
En esta práctica principalmente se han añadido dos clases, ambas pertenecientes a la parte de la vista: ReproductorUB4 (la vista principal de nuestro programa desde donde accedemos a todas las demás funcionalidades) y FrmAfegirFitxer (ventana que utilizamos para introducir la información de las canciones en el sistema).

El aspecto que queda al final es el siguiente:

ReproductorUB4:



FrmAfegirFitxer:



Reutilización de código de clases anteriores y adaptaciones

La mayoría de código del Controlador se ha podido reutilizar de las prácticas anteriores.
Las adaptaciones que he tenido que hacer han sido mínimas y mayoritariamente relacionadas con el formato de los parámetros que me llegaban o del paso de parámetros a las funciones.

A continuación enumero todos los cambios que tuve que hacer sobre el código anterior:

- afegirFitxer pedía introducir la información por teclado, para evitar esto, decidí añadir una nueva función afegitFitxerSimple que no pedía esta información
- Implementar setters para todos los parámetros de la clase FitxerAudio, problema derivado del punto anterior, necesitaba asignar los textos de los controles a los atributos de la clase de FitxerAudio
- Eliminar fichero dado su índice, lo hice haciendo sobrecarga sobre la función que ya existía eliminarFitxer, pero cambiando el tipo de parámetro por el índice
- Función donaLimit() que devuelve el tamaño máximo de elementos que puede tener una lista de reproducción, me hacía falta para controlar el límite de archivos que podía añadir a la lista

Problemas y soluciones

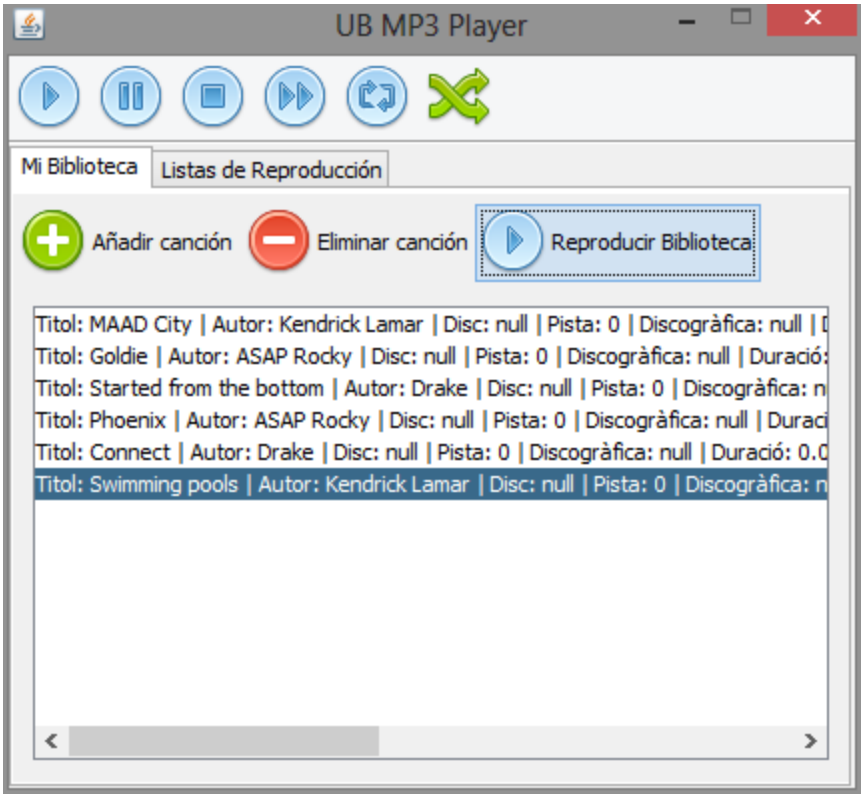
- **Aprender a usar controles y events**
  Tuve algún problema para comprender el funcionamiento de los controles y demás, conseguí solventar mis dudas haciendo con Google y stackoverflow
- **Problemas de adaptación**
  Los problemas que me encontré en el proceso de adaptación están explicados en el apartado anterior.

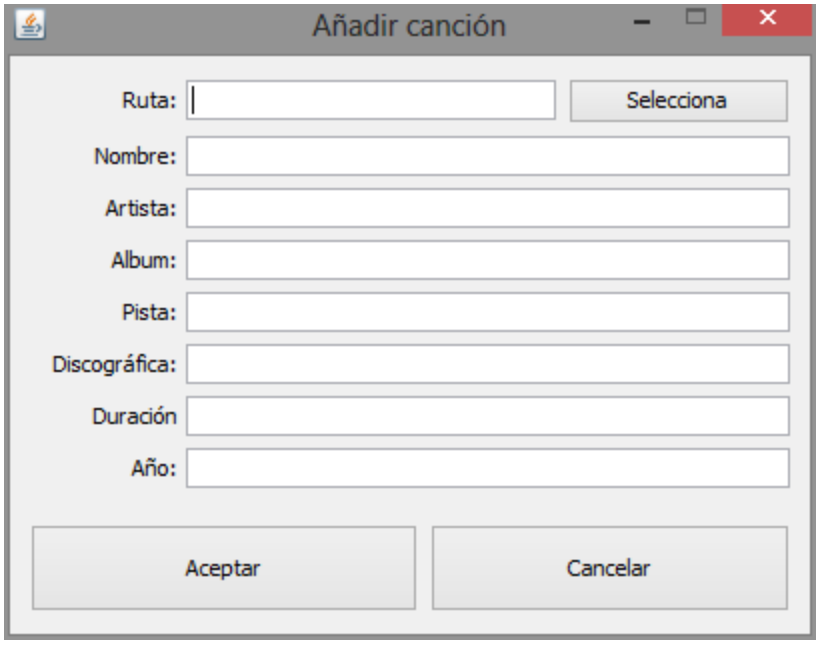
Resultados

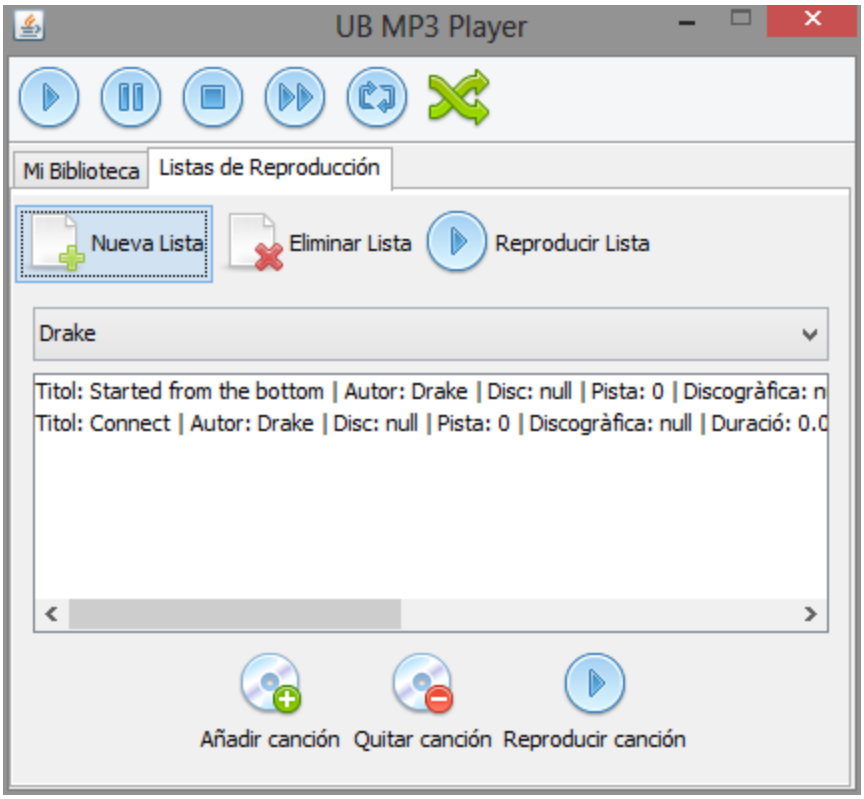
Pruebas realizadas

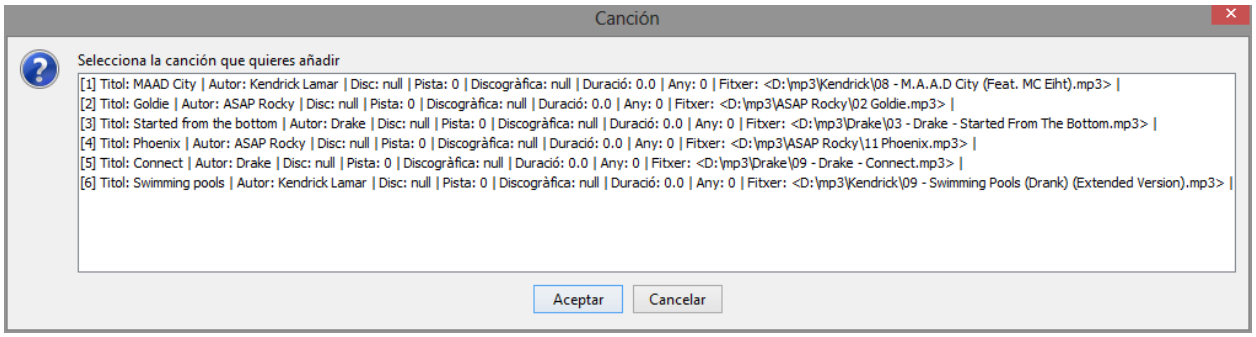
Esta vez no puede utilizar archivos de test dado que ahora la interacción con el usuario es gráfica me era más complicado usar este mecanismo. Al final la mayoría de pruebas se han hecho manualmente y a base de prueba y error.

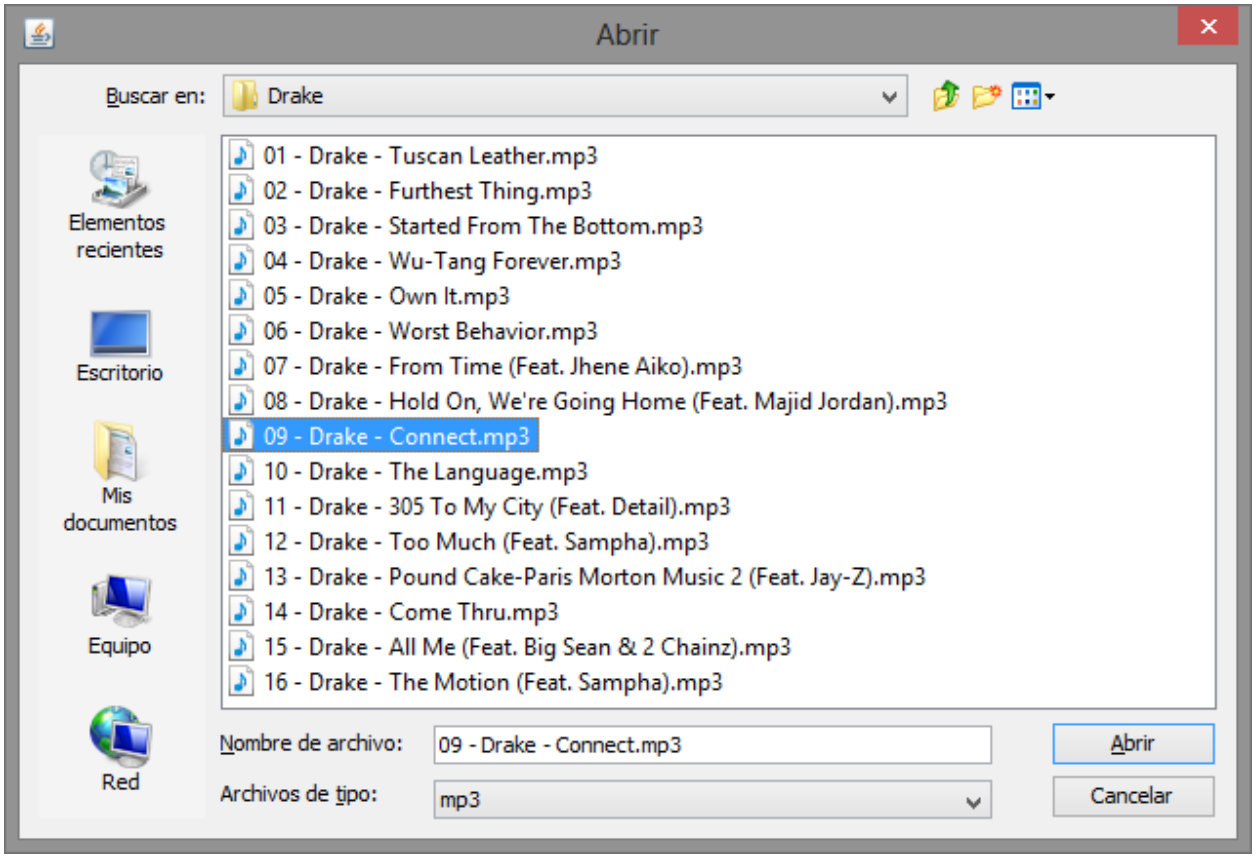
Ejemplos de Output






UB MP3 Player
Mi Biblioteca
Listas de Reproducción
Nueva Lista
Eliminar Lista
Reproducir Lista
Drake
Titol: Started from the bottom | Autor: Drake | Disc: null | Pista: 0 | Discogràfica: n
Titol: Connect | Autor: Drake | Disc: null | Pista: 0 | Discogràfica: null | Duració: 0.0
Añadir canción
Quitar canción
Reproducir canción


Canción
Selecciona la canción que quieres añadir
[1] Titol: MAAD City | Autor: Kendrick Lamar | Disc: null | Pista: 0 | Discogràfica: null | Duració: 0.0 | Any: 0 | Fitxer: <D:\mp3\Kendrick\08 - M.A.A.D City (Feat. MC Eiht).mp3> |
[2] Titol: Goldie | Autor: ASAP Rocky | Disc: null | Pista: 0 | Discogràfica: null | Duració: 0.0 | Any: 0 | Fitxer: <D:\mp3\ASAP Rocky\02 Goldie.mp3> |
[3] Titol: Started from the bottom | Autor: Drake | Disc: null | Pista: 0 | Discogràfica: null | Duració: 0.0 | Any: 0 | Fitxer: <D:\mp3\Drake\03 - Drake - Started From The Bottom.mp3> |
[4] Titol: Phoenix | Autor: ASAP Rocky | Disc: null | Pista: 0 | Discogràfica: null | Duració: 0.0 | Any: 0 | Fitxer: <D:\mp3\ASAP Rocky\11 Phoenix.mp3> |
[5] Titol: Connect | Autor: Drake | Disc: null | Pista: 0 | Discogràfica: null | Duració: 0.0 | Any: 0 | Fitxer: <D:\mp3\Drake\09 - Drake - Connect.mp3> |
[6] Titol: Swimming pools | Autor: Kendrick Lamar | Disc: null | Pista: 0 | Discogràfica: null | Duració: 0.0 | Any: 0 | Fitxer: <D:\mp3\Kendrick\09 - Swimming Pools (Drank) (Extended Version).mp3> |
Aceptar
Cancelar


Abrir
Buscar en:
Drake
Elementos recientes
Escritorio
Mis documentos
Equipo
Red
01 - Drake - Tuscan Leather.mp3
02 - Drake - Furthest Thing.mp3
03 - Drake - Started From The Bottom.mp3
04 - Drake - Wu-Tang Forever.mp3
05 - Drake - Own It.mp3
06 - Drake - Worst Behavior.mp3
07 - Drake - From Time (Feat. Jhene Aiko).mp3
08 - Drake - Hold On, We're Going Home (Feat. Majid Jordan).mp3
09 - Drake - Connect.mp3
10 - Drake - The Language.mp3
11 - Drake - 305 To My City (Feat. Detail).mp3
12 - Drake - Too Much (Feat. Sampha).mp3
13 - Drake - Pound Cake-Paris Morton Music 2 (Feat. Jay-Z).mp3
14 - Drake - Come Thru.mp3
15 - Drake - All Me (Feat. Big Sean & 2 Chainz).mp3
16 - Drake - The Motion (Feat. Sampha).mp3
Nombre de archivo:
09 - Drake - Connect.mp3
Archivos de tipo:
mp3
Abrir
Cancelar

Conclusiones

Creo que con esta última práctica se han visto claramente los beneficios del uso del patrón MVC a la hora de programar.

Hemos diseñado la vista de nuestro programa sin preocuparnos por cómo está implementando el controlador o el modelo, simplemente aprovechando las herramientas (métodos/clases) que teníamos a nuestra disposición.

Creo que es la lección más importante que llevarnos de esta práctica ya que se ha visto muy claramente que los cambios que hemos tenido que hacer han sido mínimos. Es importante ver cómo esto nos podría beneficiar en la vida real en un equipo de desarrollo, ya que la aplicación estará bien definida, modular y por tanto mucho más fácil y seguro a la hora de desarrollar a la vez con varias personas.