JAVA PROJECT REPORT

(Project Term January-May 2023)

“JPLAY: A User-Friendly Music Player using JFRAMES”

Submitted by

Farzan Khan

Registration Number

12112338

Course Code: CSE310

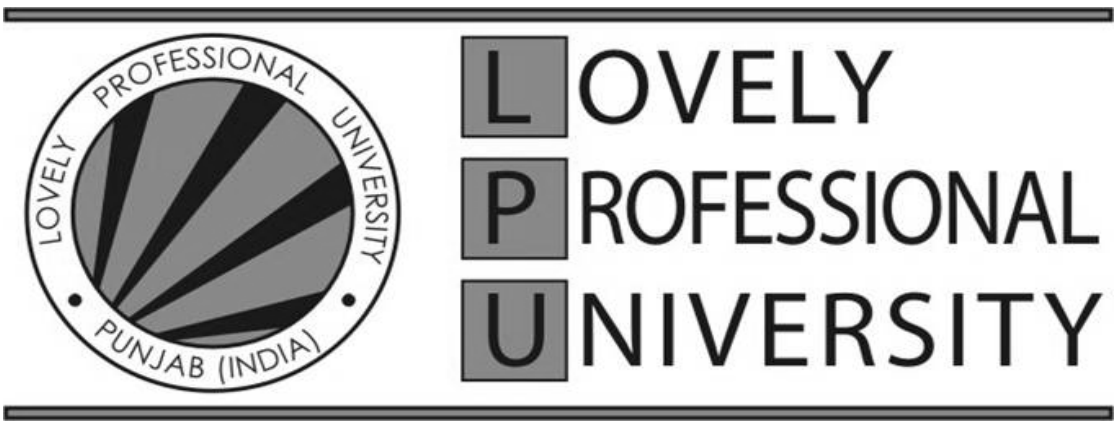
Under the Guidance of

(Dr. Ranjith Kumar A: 26108)

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab



DECLARATION

I hereby declare that the project work entitled ("JPLAY: A User-Friendly Music Player using JFRAMES") is an authentic record of our own work carried out as requirements of Java Project for the award of B.Tech degree in (Computer Science and Engineering)from Lovely Professional University, Phagwara, under the guidance of (Dr. Ranjith Kumar A), during August to November 2022. All the information furnished in this java project report is based on my own intensive work and is genuine.

Farzan Khan
12112338

Date:24-04-2023

TABLE OF CONTENTS

INTRODUCTION

JPlay is a music player application developed in Java, using JFrames for the graphical user interface. It allows users to play their favorite music tracks on their desktop or laptop, with a variety of multimedia functionalities such as play, pause, stop, previous and next.

JPlay features a user-friendly and intuitive interface, designed to provide a seamless music playback experience. With its minimalist design and easy-to-use controls, users can quickly navigate through their music library and access their favorite songs with just a few clicks.

In addition to its playback capabilities, Jplay also supports the ability to add and remove music tracks from the playlist, providing a personalized music listening experience. The application is also capable of reading and displaying metadata such as song title, artist, and album information.

JPlay's code is well-documented, with clear and concise comments explaining the purpose and functionality of each module and function. This makes it easier for other developers to understand and modify the code in the future.

Overall, Jplay is a versatile and user-friendly music player application that demonstrates the power and flexibility of Java and JFrames in developing multimedia software applications.

PROPOSED TECHNIQUE

Our music player was developed using Java programming language and a combination of frameworks and java libraries such as swing, java.awt and javazoom. Swing for the frontend user interface, java.awt for its design and visuals and javazoom for manipulating the audio file, it also supports for various audio file, playback controls, and audio visualizations.

Swing is a powerful and popular Java library that provides a set of GUI components and tools for creating modern, platform-independent graphical user interfaces.

We have used swing libraries' function such as JLabel, JFarme, JButton, eventListener in this project to build the GUI frame and created buttons, text fields with the help of these functions, and passed parameters.

Additionally, with the use of Swing a consistent look and feel a user can get across different platforms. In the context of the Jplay music player application, Swing is used to create a visually appealing and user-friendly interface that allows users to easily navigate and control the music playback.

One of the key functionality our music player is the ability for users to add their favourites tracks as many as they can.

MODULES

○ HEADER:

- First, we have given the music player name along with an exit icon. In the main class named as Jplay we have used several methods to create this window and we make it responsive. We have used different functions such as JPanel, JLabel, etc.

- We have used a png file of music logo and created an object to link it with the GUI and sets some parameter of it like setImage, setBackground so it looks attractive while experiencing by the user.

- Same functions applied for the exit icon also creating an object scaling it setting some paramters.

○ MARQUEE LABEL:

- To create this window, we have first created a separate file with class name MarqueeLabel.java.

- In this java's swing function JLabel along with the component "marquee" effect is working which is making the text display in right to left order with a constant speed continuously.

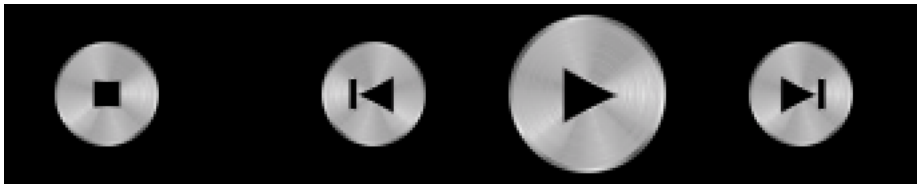
○ JPLAY LOGO:

- We have added logo png to the main java class of this project and created a method named Decoration and stated some parameters to adjust this logo and fixed in the GUI window like its boundary it should be static in center and logo appears when music is not playing.

○ BUTTONS:

- We have created buttons using swing library functions, every button is having their own method and defined various parameters like using exception handling (try and catch) creating constructor to override the run method.

- We used if and else conditions also so that it can check player is not null and give many other data to the buttons like InputStream, BufferInputStream, setting path as well.



o TRACKLIST

- First, we have created a TitledBorder named Tracks in which we have set parameters such as TitleColor, TitleJustification, TitlePosition and its border as well. We have added the specified mouse listener also to receive mouse events from the swing library component.

- We have created few more objects so that when a song start playing an animation runs at the same time by setting some parameters and scaling it to default.

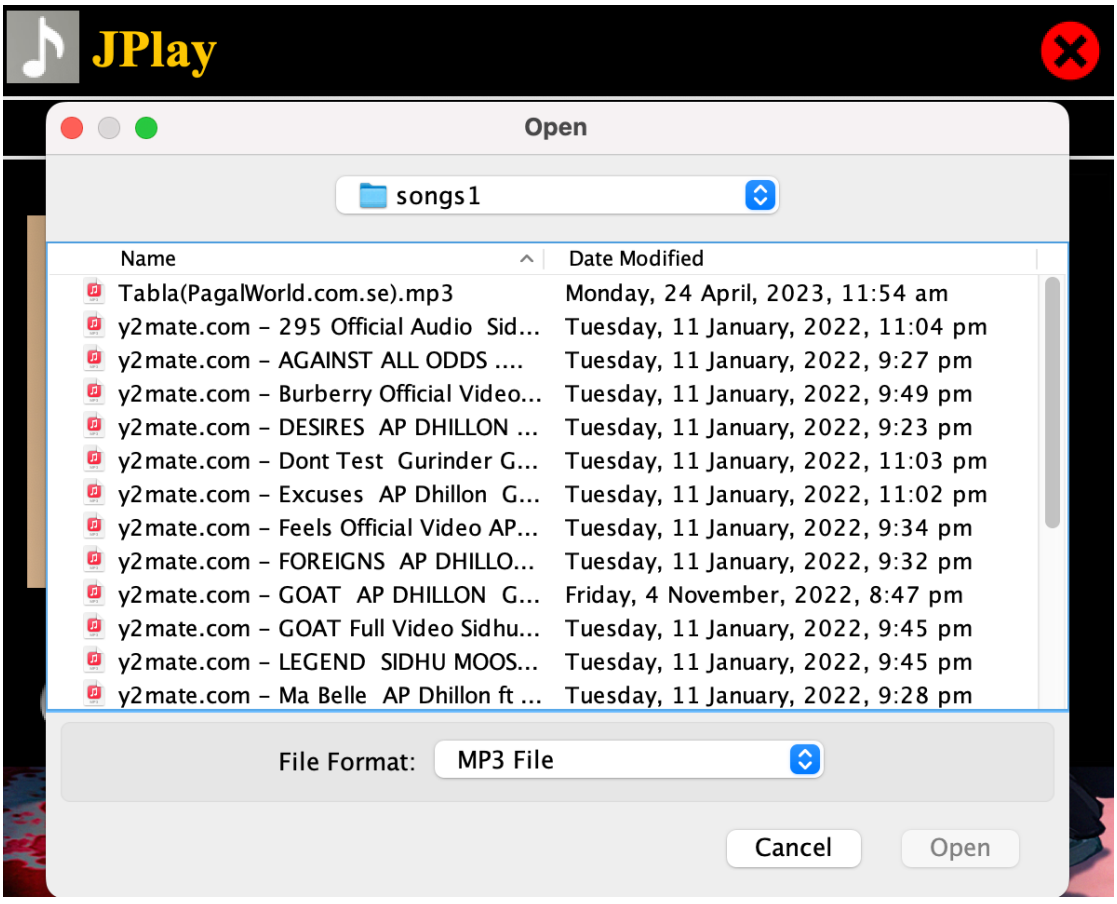
○ ABOUT:

- It is a section about the music player when we tap on the Jplay logo it appears and shared some information about the music player and the developer.

- To create this window, we have created a file named Information.java same as the class name and used swing library function such as JPanel, JLabel, Jframe and created a method inside it where we have given text input using setText function, added Jplay logo using an object and setting boundaries, font, colors, etc.

○ FILE ACCESS:

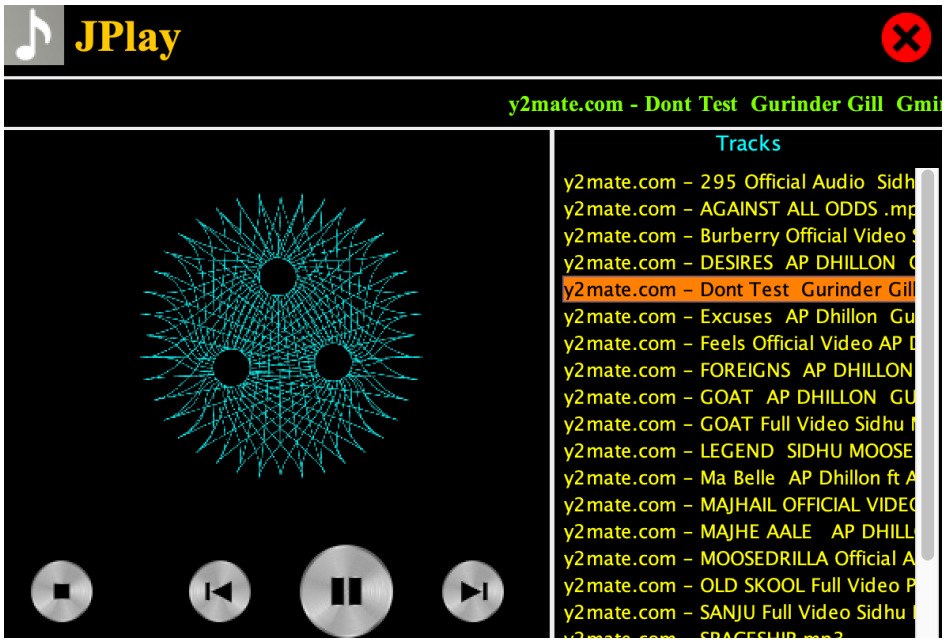
- First we have imported required libraries such as java.io and call its function InputFileStream, JFileChooser, File, etc.
- Then, we created required objects and variables to get file access files using file system and store their path to use in our program.
- We used JChooser method to provide interactive GUI for our users to choose their soundtracks from their system.



○ Animation:

- For animation while playing the song, we have created a file named Animation.java same as class name in which swing library and java.awt is used to create icon setting its width and height and to animate this we have created a method named animate.

- In that method, we have used many functions of the listed packages like getting the image, adding the animated gif file to the GUI and used exception handling concept.

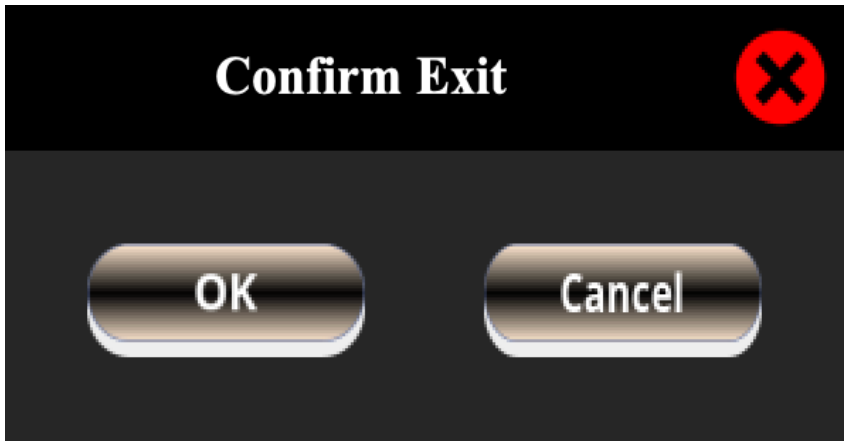


- **Exit:**

- For Exit, we have created a file named exit frame followed by class name “ExitFrame” and used java packages like swing library, java.awt.

- In that class, we created functions using JFrame, JButton, JLabel etc. and inside this class we have assigned a method in which functions are working and its parameters are assigned like setting Image, Layouts, Bounds and others.

- By the help of JButton function, ‘OK’ and ‘cancel’ buttons are created and using an object we set the parameters of the buttons and assigning its functionality.



CONCLUSION

- After developing this project music player using Java and JFrames has proven us to be an exciting and challenging work. The use of various modules and libraries such as Java Swing and Java.awt Frameworks have enabled the creation of a user-friendly GUI and the incorporation of multimedia functionalities.

- The project has also provided an opportunity to apply key software development concepts, such as modular programming and exception handling, in a practical setting. Additionally, it has allowed us for the development of valuable skills in areas such as object-oriented programming and event-driven programming.

- Overall, this music player project has been an excellent learning experience for our group members in Java and has demonstrated the versatility and power of this programming language. With further development, the music player can be enhanced with additional features and functionalities, making it a valuable tool for music lover and professionals alike.

FUTURE ENHANCEMENTS

Here are some potential future enhancements for the music player JPlay:

1. Seek: Implementing seek in our music player in order to adjust the timestamp of our sound tracks.

2. Favourites: We can allow our users to save their favourite sound tracks in our music player so there will be no hassle to pick the song.

3. Playlist: We will be allowing our users to create their own favourite playlist where they can add songs of their favourite artists.

4. Shuffle: We will be allowing shuffling option to the user so they can experience different types of tracks in one tap.

5. Sharing: We will allow our users to share their favourite sound tracks to their family, friends, and colleagues in just one go.

6. Lyrics: Allowing users to follow the music with the lyrics in-hand of their favourite song tracks.