ADVANCED JAVA (3CS503ME24)

INNOVATIVE ASSIGNMENT

Submitted by:

22BCE062: Deekshita Athreya

22BCE182: Juhi Mehta

23BCE507: Dhruvi Tanna

**SPOTIFY CLONE**



|  |  |
| --- | --- |
| Title | Page No. |
| Introduction | 3 |
| Concepts Used | 4 |
| Output | 6 |
| Conclusion | 8 |

INTRODUCTION

This Spotify clone project is a Java Swing-based application designed to give users a simple, intuitive music experience. The app includes a homepage for browsing, a search page to explore music, and a liked playlist page where users can view their favorite tracks. To provide smooth music control features like play, pause, and skip, Java threading concepts have been applied, ensuring seamless transitions and responsive functionality.

For the user interface, various Java Swing components have been utilized to create a layout similar to the popular Spotify app. Database concepts are incorporated to manage user data effectively, enhancing the app's overall experience by saving and retrieving user preferences. With a mix of UI design and background processes, this project offers a functional and engaging music app built entirely using Java Swing.

CONCEPT USED

1. **JAVA SWING**

Java Swing is used for the frontend design of this Spotify clone, helping create a responsive and visually appealing interface. Swing components like JFrame, JPanel, and JButton structure each page, including the homepage, search page, and liked playlist page. Customized elements, such as a progress bar and volume slider, mimic Spotify’s design, enhancing user experience. This layout ensures users can intuitively navigate between music controls and playlists. Swing’s flexibility allows for easy customization, making the interface attractive and user-friendly.

1. **DATABASE / JDBC (CRUD OPERATIONS)**

A database, accessed via JDBC, is the backbone for managing song data in this project, enabling CRUD operations (Create, Read, Update, Delete). When a user likes a song, an INSERT query adds it to the liked playlist, and SELECT queries retrieve songs for display. This setup allows users to save and view favorite songs, preserving their choices between sessions. By handling data reliably, the database brings consistency to the music experience. These CRUD operations make managing and displaying song data efficient and seamless.

1. **THREADING**

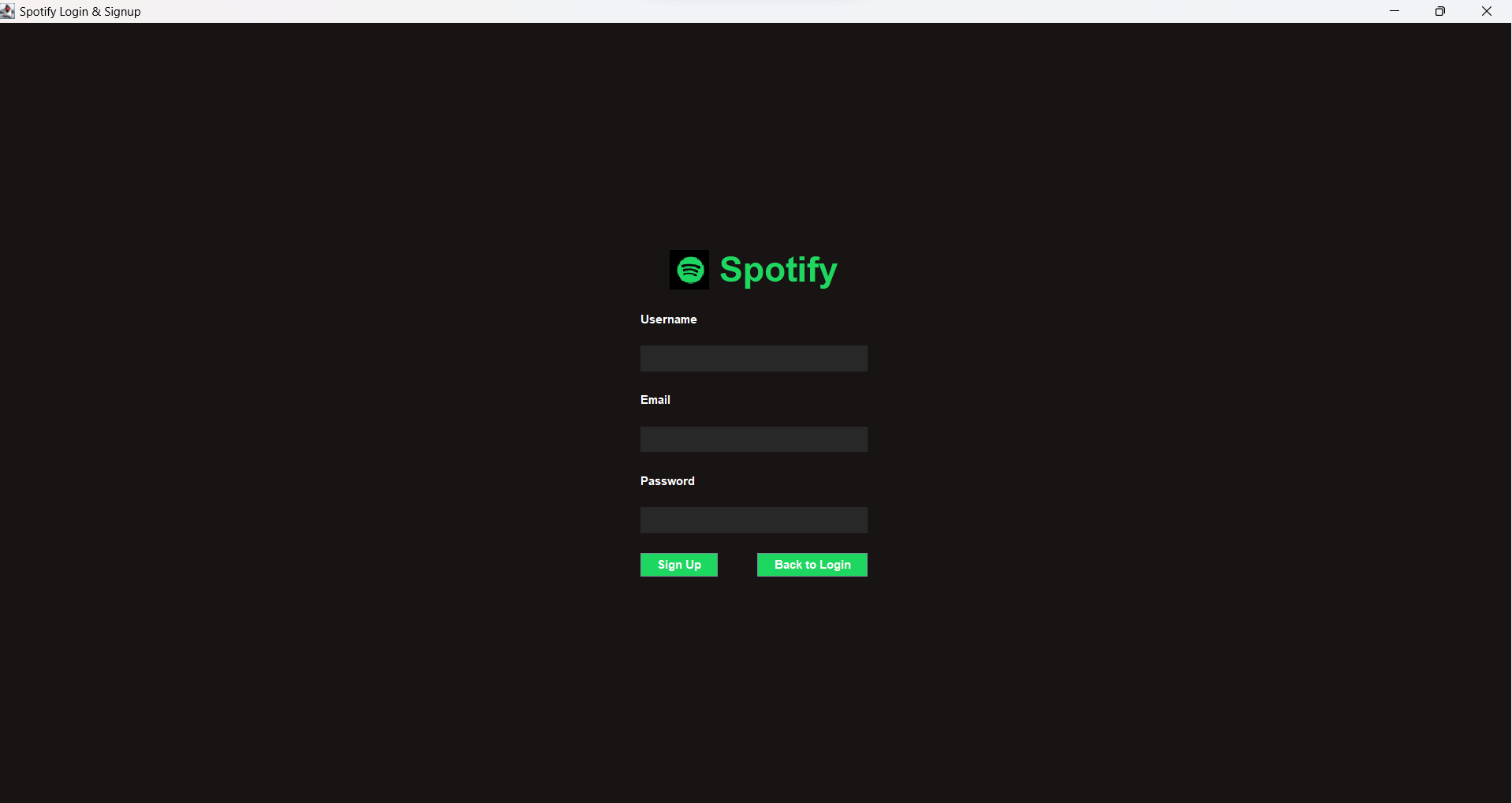
Threading is essential for smooth music playback, letting users enjoy music in the background while navigating the app. By creating a separate thread for playback, the project allows users to control play, pause, and skip functions independently from the main interface. This asynchronous setup ensures that controls respond quickly without affecting other app functions. Threading adds realism to the app, making it feel similar to real streaming services. It improves user experience by preventing lags or interruptions during music play.

1. **EXCEPTIONS**

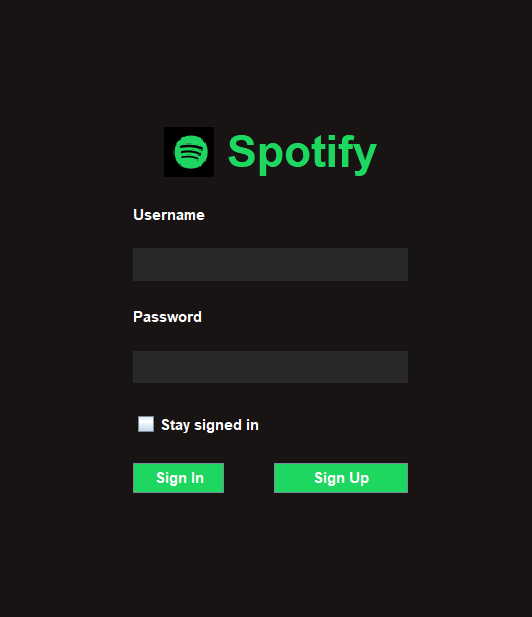
Exception handling is crucial in this Spotify clone to ensure smooth operation and prevent unexpected crashes. By managing potential issues, such as database connection errors or playback issues, the app provides a reliable experience for users. Specific exception handling for each function—like data retrieval or file loading—ensures stability and quick error recovery. This approach keeps the app running smoothly even when unexpected issues arise. Proper exception handling helps build a resilient, user-friendly music streaming experience.

OUTPUT

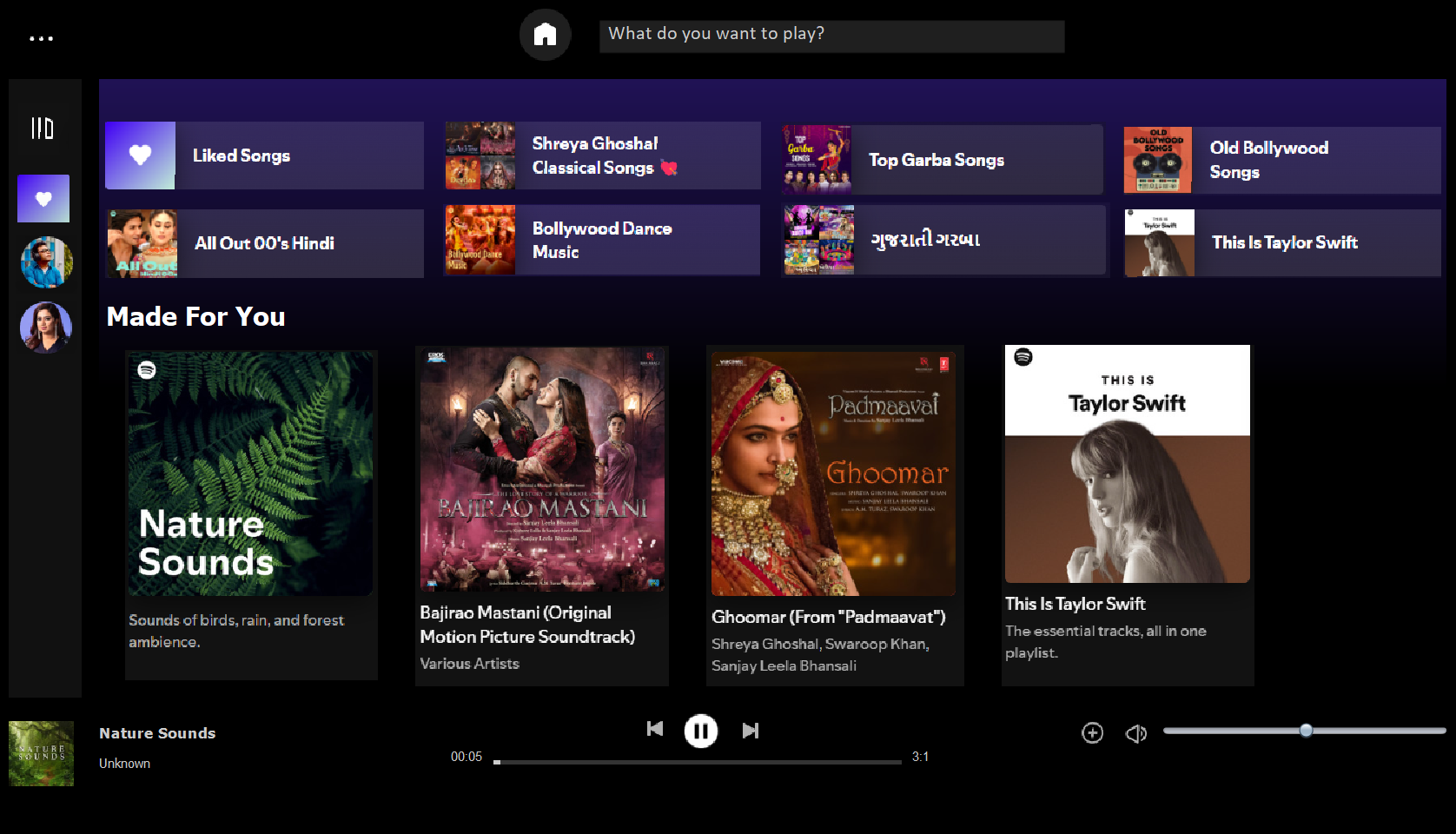
* SIGNUP PAGE:



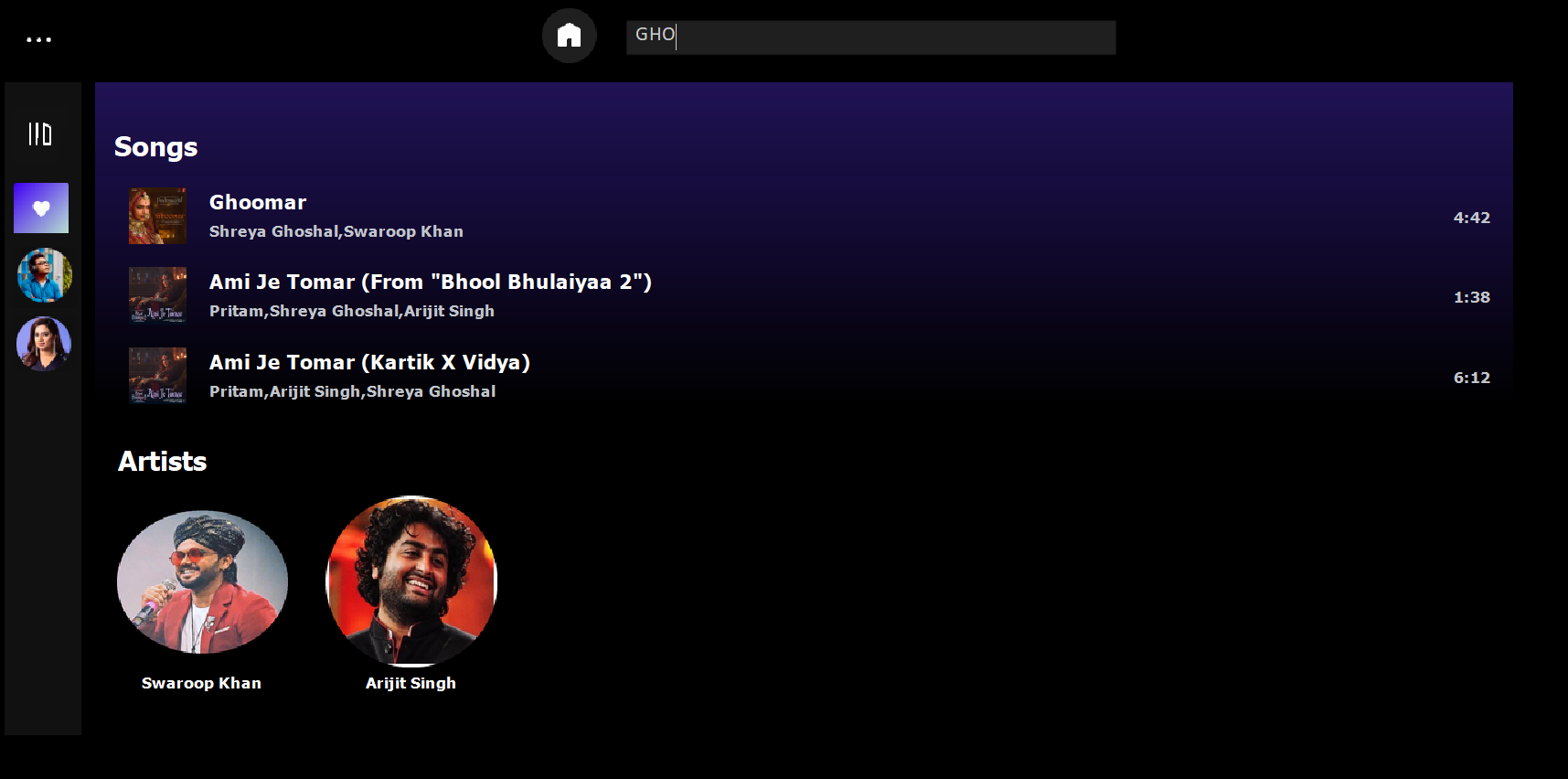
* LOGIN PAGE



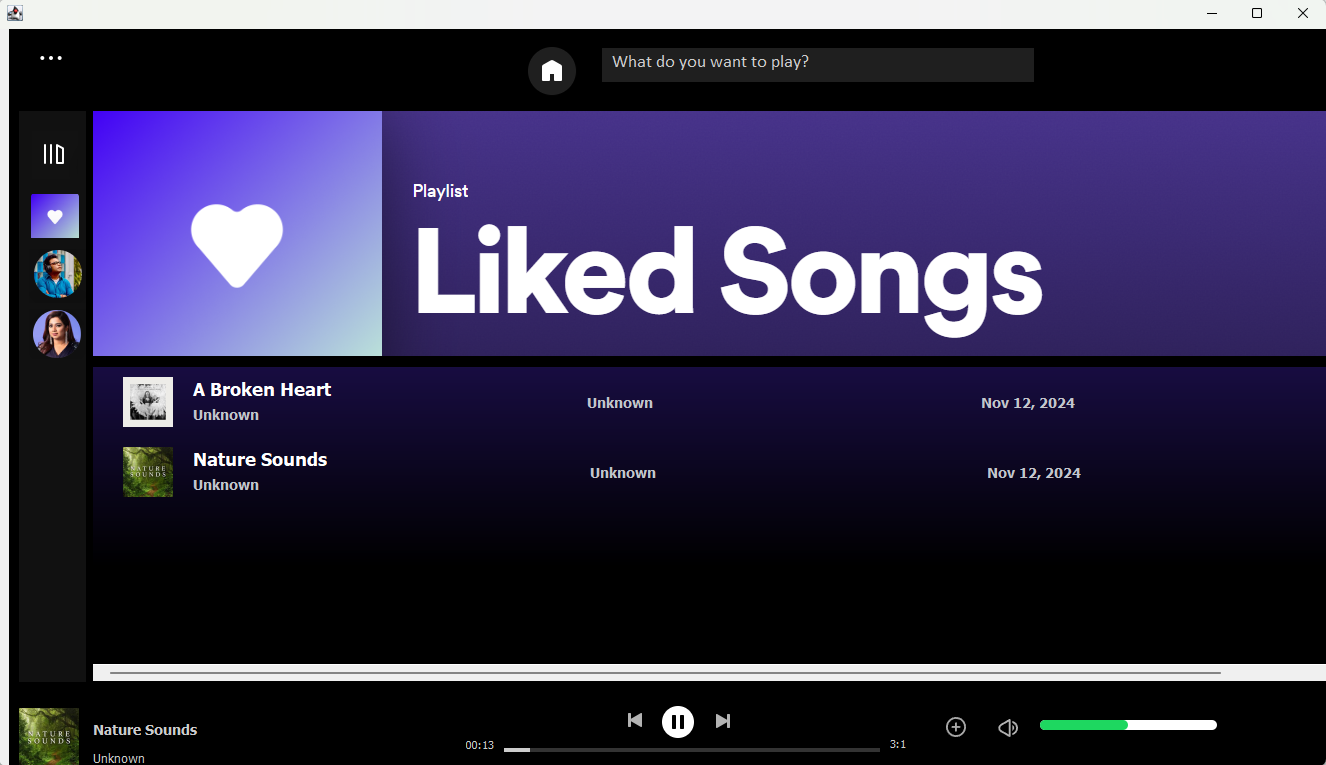
* HOME PAGE



* SEARCH PAGE



* LIKE PLAYLIST PAGE



CONCLUSION

In conclusion, this Spotify clone project combines Java Swing, database management, threading, and exception handling to create a user-friendly and reliable music streaming experience. The intuitive interface, backed by smooth music playback and secure data storage, ensures that users can easily navigate, play, and save their favorite songs. Each component works together to provide a seamless, realistic experience similar to popular streaming apps. This project showcases essential Java skills and delivers an engaging application. Overall, it effectively replicates key features of a professional music player.