

Groovy

Work Summary & Member Contribution

September 16, 2018

Team Members

Nasser Al Suhaimi 012498289

Xinyi Chen 012963494

Patricia Echual 013470806

Jennifer Luong 013354235

Caitlin Rubia 013433353

Table of Contents

Table of Contents	2
Compilation Instructions	3
Work Completed	3
Data Store and Exchange	3
Login	3
Playlist Creation and Deletion	3
Search Feature	3
Media Player	3
Member Contributions	4
Nasser Al Suhaimi	4
Xinyi Chen	4
Patricia “Trisha” Echual	4
Jennifer Luong	4
Caitlin Rubia	4

Compilation Instructions

In order to compile the program, open the GROOVY Github repository ([Link](#)). Clone the repository onto your computer. Navigate to the “executable” folder within the repository. Open the “GroovyTunes” .jar file by double-clicking it. Once the program has been executed, the login window will display.

Work Completed

Data Store and Exchange

Data, such as user information and song library, is stored as JSON (JavaScript Object Notation) objects. In order to read the JSON objects, the data is deserialized into Data Transfer Objects (DTOs) that have been defined within the program. GSON is an open source Java library that is used to help with the serialization and deserialization of JSON objects.

Login

The system requires the user to enter correct credentials in order to use the system. For this milestone, we have hard-coded credentials for anyone to enter upon login. When a user enters incorrect credentials, an error message will be displayed.

Playlist Creation and Deletion

A playlist is a list of recorded songs. A registered user can create and delete playlists from their profile by pressing the + button on their homepage.

Search Feature

A user can type in any keyword to search for a song, album, or artist. The user must indicate what he/she wants to search for a song, album, or artist by selecting the *Song*, *Album*, or *Artist* button. A table of results will display based on the given keyword.

Media Player

A song can be played using the media player. The features include play, pause, previous song, and next song. If a user wishes to play songs from a playlist, the media player will continue to play the songs until it has reached the end of the playlist.

Member Contributions

Nasser Al Suhaimi

Nasser worked on the front-end. Designed the prototype and implemented the GUI "FXML". Worked on both pages Long in and main app. also create the Media functions and the search bar. Helped in connecting the back-end with the front-end.

Xinyi Chen

Xinyi implemented the front-end functions that display table contains playlist, and table contains search result and the songs in the playlist. She also implemented the functions that take care of adding playlist, deleting the playlist, and add song to the playlist in the front end. She also implemented the play song function when user double-click the song.

Patricia "Trisha" Echual

Trisha set up data seeding and created models to encapsulate object types used throughout the project. She also implemented the login and search features in the back end.

Jennifer Luong

Jennifer set up the structure of the program. She implemented the functions to create and delete playlists as well as the functions to use the media player.

Caitlin Rubia

Caitlin implemented the getter and setter methods for all the models. She also is working on a function to create a library that is serialized and deserialized json files.