Gyuwon Lee (SEC 03)

Lex Lei (SEC 03)

Deo Ong (SEC 03)

Justin Woo (SEC 03)

Problem statement - describe the problem that your project is trying to solve

Music is becoming more impactful every year and with tens of thousands of songs being released everyday, people are struggling to find songs that suit their preferences. With our project, we plan to create a user interface that interacts with a master database of songs that people can use to select songs that are suited specifically to the user. Coupled with other users's playlists and ratings made across the platform, it can provide more guidance for users to create a more personalized playlist.

Solution statement - describe the solution you implemented to solve the problem

The song database aims to create a platform where songs can be stored according to each user's preferences (similar to Spotify playlists). Each song has a specific genre to them, which can help users determine which songs they want in their playlists. Users can also determine which songs the general public likes by looking at the rating. Playlists that appeal to several users can 'follow' the playlist, increasing its following count. The platform comes with a user interface that allows each user to navigate smoothly to different songs and playlists available to them.

User - describe the typical user(s) that would use your solution

There are no restrictions on the type of users that would use the database. Listening to songs is a universal activity. With the implementation of a variety of genres, several users are welcome to select and filter out the type of songs they would want to save and organize into different playlists. Each user can have multiple playlists according to their preference, which makes our database a versatile solution.

The implementations of our database can be used in different contexts as well. Regardless of the environment, work setting, parties, athletic activities, or relaxing at home, the database of songs will help the users determine which songs they would want to listen to. The personalization of each user's playlist allows each user to have their own unique preference expressed through their database.

Domain objects - describe at least two of the domain objects you implemented in your solution

‘Playlists’ is the first domain object. It has properties such as ‘title’, ‘followers’, ‘created’, and ‘updated’; each property respectively corresponds to the name given to the playlist, the number of followers the playlist has, the date at which the playlist was created, and the date at which the playlist was most recently updated. ‘Playlists’ is used to store songs that the user has selected.

‘Songs’ is the second domain object with properties such as ‘title’, ‘artist’, ‘feature’, and ‘length’. The title is the name of a specific song. The artist is the main singer of the song with ‘feature’ as the guests that the singer performs with. Length is the length of the song in seconds. Each property provides the user an avenue for sorting and filtering their choice in music.