

Group Name: BTS(ETJLAT)

Project Name: Top 100 Billboard Tracker

Members:

- Tejas Siddaramaiah
- Eric Tang
- Jason Lei
- Aiden Tan

Applications:

1. Dashboard Application: Song Statistics about top 100 songs
2. Core Application: Basic CSS and Templates

Components:

1. Dashboard/Data Application:

- Information on how to run the website and mentioning specific features.
- Data about the song artist, title, rank, and other basic information
- Graph showing how the song has risen and fallen in the top 100

2. Search Application:

- Active search bar to find specific songs, artists, or albums

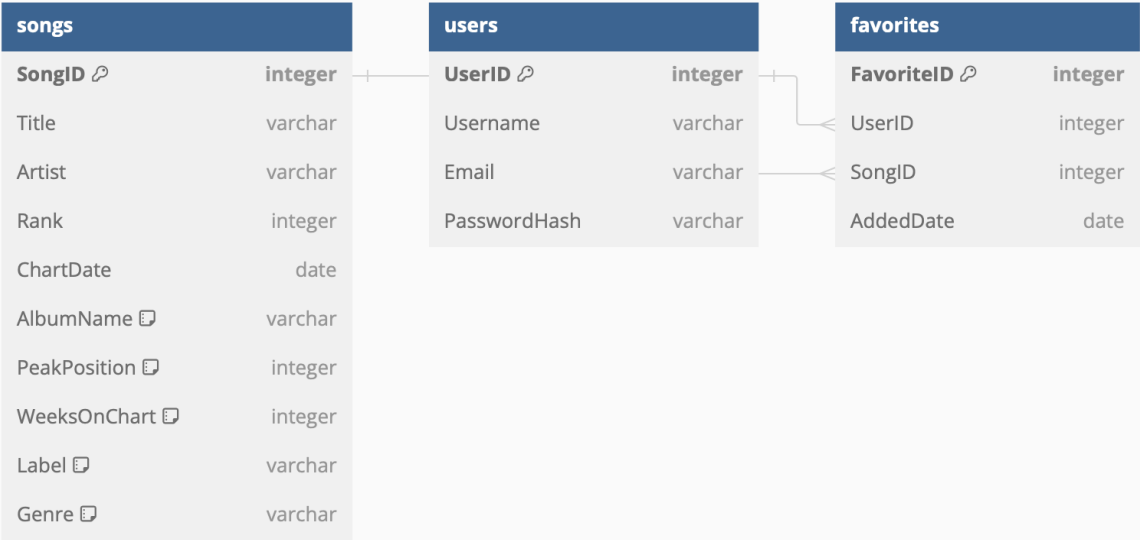
3. User Application:

- System for push notifications and email alerts on new songs on billboard.
- Playlist for user with selected songs or artists

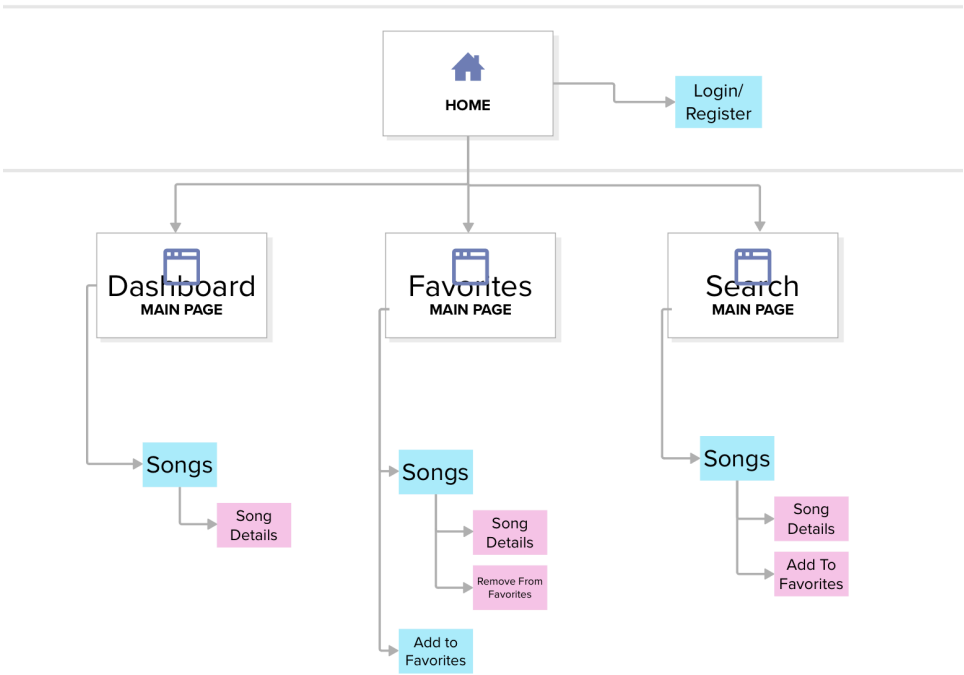
Relationships Between Components:

- Main Application interacts with the Data Application to retrieve and display songs.
- User Application uses the data from Data Application to personalize user experiences based on saved preferences.
- Search Application uses data from Data Application to actively search for similar information.

Database Design:



Site Map:



Task Breakdown:

1. Search Application: Jason and Aiden
 - Display Song Info: All
2. Data Application: Tejas, Eric, Jason, and Aiden
 - Display Song Info: Eric and Tejas
 - Adding Mechanism: Tejas and Aiden
 - Display Graphs: All
3. User Application: Tejas and Eric
 - Login/Register: Tejas
 - Connecting to Account to Songs: Eric

APIs

1.) Billboard API

- API Documentation: Billboard API
(<https://rapidapi.com/LDVIN/api/billboard-api>)
- Endpoints:
 - /charts: Fetches the current top 100 songs chart.
 - /search: Searches for songs, albums, and artists on the charts.
- Usage:
 - The /charts endpoint will be used to retrieve the list of top 100 songs for displaying on the main page.
 - The /search endpoint allows users to find specific songs or artists in the top charts.
- Data Handling:
 - The fetched data will be parsed and stored in the local database under the Songs table, which includes fields like Song ID, Title, Artist, Rank, and Date.

2.) API Documentation: Spotify API

(<https://rapidapi.com/Glavier/api/spotify23/>)

Endpoints:

- /search: Searches for songs, albums, and artists using various filters.
- /track: Retrieves detailed information about a specific track.

Usage:

- The /search endpoint allows users to search for music content by song name, artist, or album.
- The /track endpoint provides detailed data on a specific track, including the song's duration, explicit content status, popularity, and etc.

Data Handling:

- Data from the /search endpoint will be utilized to display search results to users, allowing them to explore different songs, albums, or artists based on their search criteria.
- Information retrieved from the /track endpoint can be used to present detailed song information on a webpage or app, such as lyrics, album art, and artist bio.