

Features List (In Order of Priority):

- Display top 5 songs/artists
 - Users will be able to see their top 5 songs and artists as well as their friends' top 5 songs/artists. These will be displayed side by side so that the user can compare their listening patterns with those of their friends.
 - If we cannot get the friends feature to work, we will just display the user's top 5 songs/artists.
- Save timestamps from songs/podcasts
 - Users will be able to save time stamps from songs and/or podcasts that they are listening to.
- Show spotify song history
 - Users will be able to view their most recently played songs(~in the last month), which will be pulled from the Spotify API.
- Search Timestamps (backup plan if we can't get the friends features to work)
 - Users will be able to search through their saved timestamps.
- Add friends from spotify (tentative because we realized it may not be possible with the Spotify API)
 - Users will be able to access friends from their Spotify account and add them within our website. They will also be able to add new friends from our application.
- View friends from spotify (tentative because we realized it may not be possible with the Spotify API)
 - Users will be able to view all of their current friends within our website.

Front End Design:

https://github.com/CSCI-3308-CU-Boulder/3308SP21_section012_2/blob/main/SpotiTRYwireframes.pdf

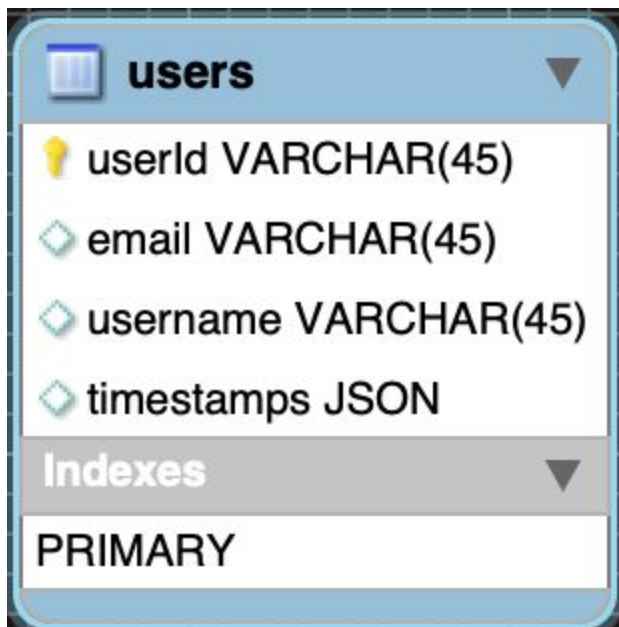
Web Service Design:

- Spotify Web API: <https://developer.spotify.com/documentation/web-api/>
 - Based on simple REST principles, the Spotify Web API endpoints return JSON metadata about music artists, albums, and tracks, directly from the Spotify Data Catalogue.
 - Web API also provides access to user related data, like playlists and music that the user saves in the Your Music library. Such access is enabled through selective authorization, by the user.
- get requests
 - authorization to user's Spotify data on login page
 - user's top 5 songs (song title, artist, album name, album cover)
 - user's top 5 artists
 - user's listening history

- user's friends list
- post requests
 - adding Spotify friends within our external website (tentative)

Database Design:

- The database that our application will be using is called Firebase. Firebase is a NoSQL database created by Google. In our application we use Firebase to store the following: userid, email, username, and an array of the user's time stamps. Because of the simplicity of our local database we only have one table for it. The ERD diagram for the table looks like this:



Challenges:

- Challenge 1: making our features unique from what Spotify already has available
 - Backup Plan: Have at least one unique feature to display such as top 5 songs.
- Challenge 2: Adding friends might not be possible
 - Backup Plan: Add a search functionality instead
- Challenge 3: making sure that the song history feature updates to reflect the most recent listening patterns of the user
 - Backup Plan: Display song history without updates or possibly just update whenever the user logs in again.

Individual Contributions:

Kyra - I put together this milestone document and worked on the features list, web service design, and challenges. I also worked with John to try to understand how we can pull the user's top 5 songs and artists.

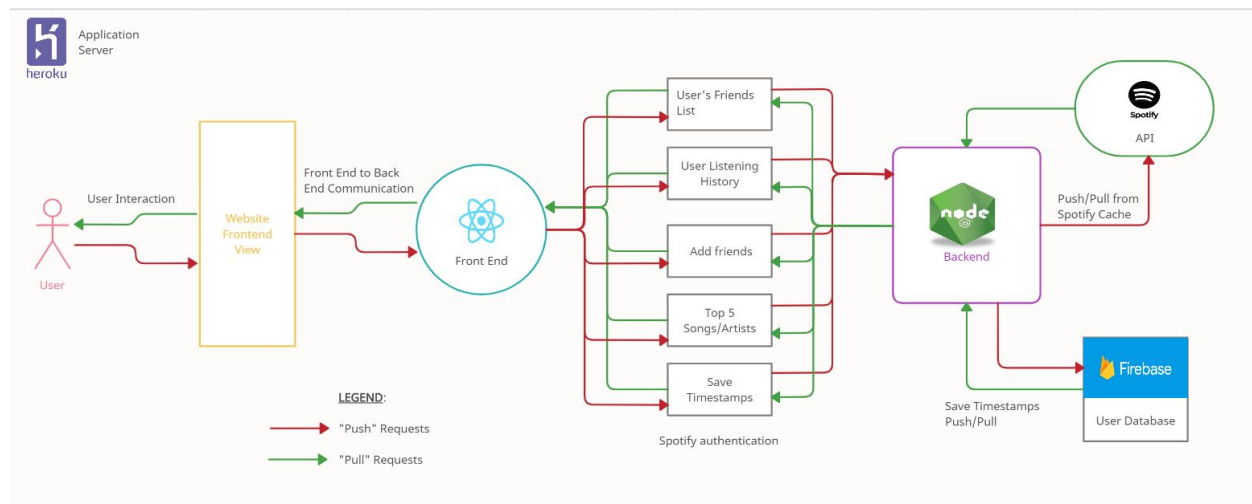
John - Worked with Kyra to try and understand how we could pull the user's top 5 songs and artists. I also worked on the homepage html and basic web design and layout while David worked on the data aspect of the page.

Ethan - Assisted David in choosing and implementing the database design. Worked on creating and styling cards in the home screen where we will be populating the data that we receive from Spotify API.

David - Implementing Firebase with our app and setting up the authentication for new users and existing users.

Sandy - Worked on timekeeping and updates to git, managing the JIRA board, along with working with Aditya on the architectural diagram and reviewing front-end design. Going to help more on the code end for the next sprint/milestone.

Aditya - Finished the architectural diagram and was able to work with Sandy on the Front End Design. I also tried to fix my issue of not being able to run react on my OS but that hasn't been resolved yet and will try again and hopefully get it up and running soon. The finalized architectural diagram is below:



Architectural Diagram Link: <https://app.creately.com/diagram/m9LH7OhH0g5/edit>