Capstone Project Proposal

Stack Focus

1. *What tech stack will you use for your final project? We recommend that you use React and Node for this project, however if you are extremely interested in becoming a Python developer you are welcome to use Python/Flask for this project.*
-I believe I will use python/flask for this since that is what I am familiar with and have used. If you believe I can use React/Node though, I am open to the experience.

Type

2. *Is the front-end UI or the back-end going to be the focus of your project? Or are you going to make an evenly focused full-stack application?*
-I will be focusing more on the front end and trying to create a good user experience, based on the outline of this project though, I will need a strong back end to create that positive experience as well.

3. *Will this be a website? A mobile app? Something else?*
-This will be a website, I believe a mobile app would be too complex for me at this time and I would like to stay within my comfort zone a little bit for now.

Goal

4. *What goal will your project be designed to achieve?*
-I will be working at creating a positive Spotify user experience that allows for curating playlists/music based on user interest or input. Allowing the user to maximize their creativity and ability to develop their own playlists or sounds based on moods or past music interest.

Users

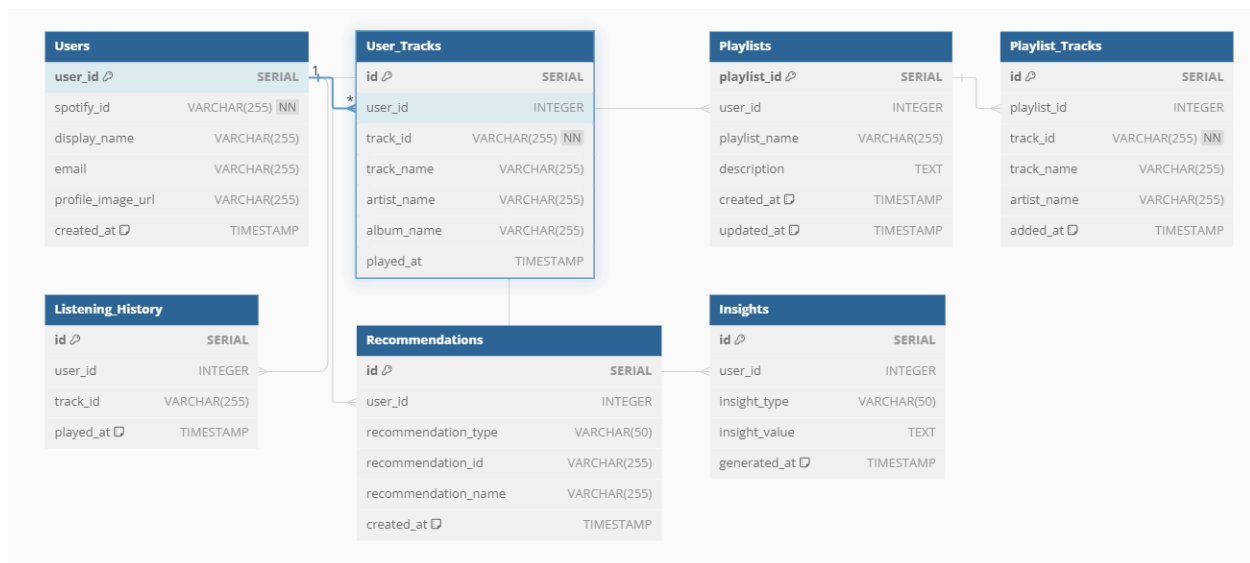5. *What kind of users will visit your app? In other words, what is the demographic of your users?*
-The app will be targeted at Spotify users to allow them to go above and beyond what spotify offers for listening options (Playlists, music suggestions, Artists)

Data

6. *What data do you plan on using? How are you planning on collecting your data? You may have not picked your actual API yet, which is fine, just outline what kind of data you would like it to contain. You are welcome to create your own API and populate it with data. If you are using a Python/Flask stack are required to create your own API.*
-I will be using the free spotify API that is available to use.

● *What does your database schema look like?*

● *What kinds of issues might you run into with your API? This is especially important if you are creating your own API, web scraping produces notoriously messy data.*
I am not totally sure if I will run into issues with the Spotify API or if it will give me all of my relevant information I need.
● *Is there any sensitive information you need to secure?*
-Yes, User login and passwords will need to be secured, will probably use bcrypt to cover this
● *What functionality will your app include?*
-Capability for user to navigate to current "liked" music, playlists, or recommendations. Also navigate to new music based on past interest
● *What will the user flow look like?*

1. Landing Page

- **Purpose:** *Introduce the app and its features.*
- **Actions:**
    ○ *Option to log in with Spotify.*
    ○ *Overview of what the app offers (personalized recommendations, curated playlists, insights, etc.).*
    ○ *A "Get Started" button leading to the Spotify OAuth login.*

2. User Authentication (Spotify OAuth)

- **Purpose:** *Allow users to authenticate using their Spotify account.*
- **Actions:**

- Redirect to Spotify's login page.
- User grants necessary permissions (e.g., accessing listening history, creating playlists).
- Redirect back to the app with the user's access token.

3. Home/Dashboard

- **Purpose:** Provide a personalized experience based on the user's Spotify data.
- **Actions:**
  - Display a welcome message with the user's name.
  - Show a quick summary of their recent listening activity.
  - Navigation options:
    - **Discover New Music:** Leads to personalized recommendations.
    - **Your Playlists:** View and manage auto-generated and existing playlists.
    - **Listening Insights:** Visualizations and statistics about the user's listening habits.
    - **Search & Explore:** Search for new artists, albums, and tracks.

4. Discover New Music

- **Purpose:** Present personalized song, artist, and playlist recommendations.
- **Actions:**
  - Display a list of recommended tracks, artists, and playlists.
  - Options to:
    - Play a preview of a track.
    - Add a track to a playlist.
    - Save a recommended playlist to their Spotify account.
  - Option to refresh recommendations based on different criteria (e.g., mood, genre).

5. Your Playlists

- **Purpose:** View, manage, and create playlists.
- **Actions:**
  - Display auto-generated playlists based on listening history, mood, or other criteria.
  - Allow users to create a new playlist from scratch.
  - Edit existing playlists (add/remove tracks).
  - Sync changes with their Spotify account.

7. Search & Explore

- ***Purpose:*** *Allow users to search for new music beyond recommendations.*
- ***Actions:***
  - *Search bar to find tracks, albums, artists, and playlists.*
  - *Display search results with options to:*
    - *Play a preview.*
    - *Add to a playlist.*
    - *Follow artists or save albums to their Spotify account.*

8. Social Sharing

- ***Purpose:*** *Enable users to share their favorite music and insights.*
- ***Actions:***
  - *Share options for:*
    - *Playlists.*
    - *Favorite tracks.*
    - *Listening insights.*
  - *Integration with social media platforms (e.g., Twitter, Instagram) to share links or images.*

9. User Settings/Profile

- ***Purpose:*** *Manage account details and preferences.*
- ***Actions:***
  - *View and update user profile (e.g., display name, profile picture).*
  - *Manage app settings (e.g., notification preferences, theme).*
  - *Option to log out or disconnect Spotify account.*

10. Logout

- ***Purpose:*** *End the user session.*
- ***Actions:***
  - *Log the user out of the app.*
  - *Redirect to the landing page with an option to log in again.*

● *What features make your site more than a CRUD app? What are your stretch goals?*
User Authentication via Spotify OAuth (Basic user login and integration with Spotify).
Personalized Recommendations (Use Spotify's recommendation API to suggest tracks or artists).
Basic Playlist Management (Allow users to view and save playlists).

Basic Listening Insights (Provide simple stats like top artists or tracks).
Search Functionality (Enable users to search for tracks or artists).
Goals:

Collaborative Playlists:

- **Goal:** Allow multiple users to collaborate on playlists in real-time, with features like voting on tracks, adding comments, and merging playlists from different users.

User-Generated Content:

- **Goal:** Allow users to create and share their own content, such as reviews, playlists with custom descriptions, or music blog posts. You could even include a community-driven aspect where users can follow each other and see what their friends are listening to.

While I believe this is a large project, I do believe I will be able to work through it. I may need to scale it down to achieve it in a necessary amount of time but I would like to aim for at least one feature that sets it apart from being a basic application.