Capstone Two Project Proposal

Description

Stack Focus: This will be an evenly focused full-stack application using React, Node.js, Express, JavaScript, HTML, and CSS. The project will balance the front-end and back-end development, with an emphasis on creating an interactive, dynamic user experience.

Type: The project will be a website designed for creating collaborative playlists, with password protection for exclusive access. It will cater to various events where users can contribute songs to a shared playlist.

Goal: The project aims to solve the problem of organizing group music selections for events by allowing users to create password-protected collaborative playlists. Using the Spotify API, users can invite others to contribute songs to a shared playlist for events like weddings, weekend getaways, or parties. The platform will allow users to securely access and add to playlists with a password, ensuring only invited participants can contribute, making the music selection process fun and interactive.

Users

The app's target demographic includes individuals planning events (e.g., weddings, parties, getaways) who want to collaboratively build playlists. This includes hosts who want to invite guests to add songs to a playlist, as well as guests who want to contribute their favorite tracks to the event. The app will primarily appeal to people who enjoy music and want to make the music selection process more interactive and social.

Data

The app will primarily use the **Spotify API** to pull in song, artist, and album data. Users will create collaborative playlists, and Spotify's catalog will provide the necessary information for music selection. The app will collect and manage the following data:

- Playlist information (e.g., name, description, visibility settings)
- User information (e.g., username, email, password for access control)
- Songs and artists added to playlists (e.g., song name, artist, album)

The application will also feature password-protected playlist creation, ensuring that only invited users can contribute to the playlist. The backend will handle user authentication and secure access to the playlists using token-based authentication or other appropriate methods.