Music Recommendation Engine

Goal
The goal of the web app is to help music lovers discover new music that they will enjoy. By creating a music recommendation engine, the aim is to provide personalized recommendations to our users based on their musical preferences and listening habits.

Users
Target users are music enthusiasts of all ages who are looking for new music recommendations. The demographic of our users is likely to be broad and diverse, as music is enjoyed by people of all backgrounds and interests.

Data
The plan is to use the Spotify API to access music data such as artist names, genres, and song titles.
https://developer.spotify.com/

Approach

Schema
- app.py (main application file)
- requirements.txt (list of Python packages required for the application)
- templates (folder for HTML templates)
- base.html (base template that other templates extend)
- index.html (home page template)
- recommendations.html (template for displaying music recommendations)
- register.html (template for user registration)
- login.html (template for user login)
- static (folder for static files like CSS and JS)
- styles.css (CSS file for styling the HTML templates)
- utils.py (helper functions for processing API data and generating recommendations)
- models.py (database models for user accounts)

API Issues
- I may encounter rate limits or authentication issues when accessing the Spotify API. The API availability needs to be considered.

Sensitive Information
- I will need to secure user login information and any personal data collected from users for security.

Functionality
- The app will include user authentication, a personalized dashboard for users to view their listening habits and recommendations, a search function to find new artists and songs, and a feedback system to improve recommendation accuracy.

User Flow
- The user flow will begin with user authentication, followed by a series of questions to determine their musical preferences. Based on the responses, the app will generate a personalized dashboard with recommended artists and songs.

Features
- The site will be more than just CRUD by incorporating personalized recommendation algorithms, a feedback system, and a user dashboard to view listening habits and recommendations. The stretch goals include integrating with other music APIs and developing a mobile app for on-the-go music discovery.