Review of Version 1: *Emotion Based Playlist Generator*

1. Project Overview

The **Emotion-Based Playlist Generator** is a web application designed to create music playlists based on the user's emotional state. The user provides a brief textual description of their current mood, which is analyzed using sentiment analysis (via **TextBlob**). The application then selects songs from a curated dataset of music tracks, using sentiment tags such as **valence**, **arousal**, and **dominance** to match the user's emotional state to the appropriate tracks.

Currently, the application includes the following implemented features:

- **Sentiment Analysis**: Uses TextBlob to analyze the user's mood and determine whether it is positive, negative, or neutral.
- **Musical Sentiment Dataset Integration**: The app uses a dataset from Kaggle, which includes **valence_tags**, **arousal_tags**, and **dominance_tags** to match songs with the user's emotional state.
- **Song and Artist Display**: The generated playlist displays song titles and artist names, with enhanced formatting for better readability (song names in bold and artist names in italics).
- **Navigation and User Interface**: Users can enter their mood via a form on the homepage, generate a playlist, and easily return to the homepage to generate another playlist.

2. Current State Demonstration

This section demonstrates how the application works at its current state.

Screenshots:

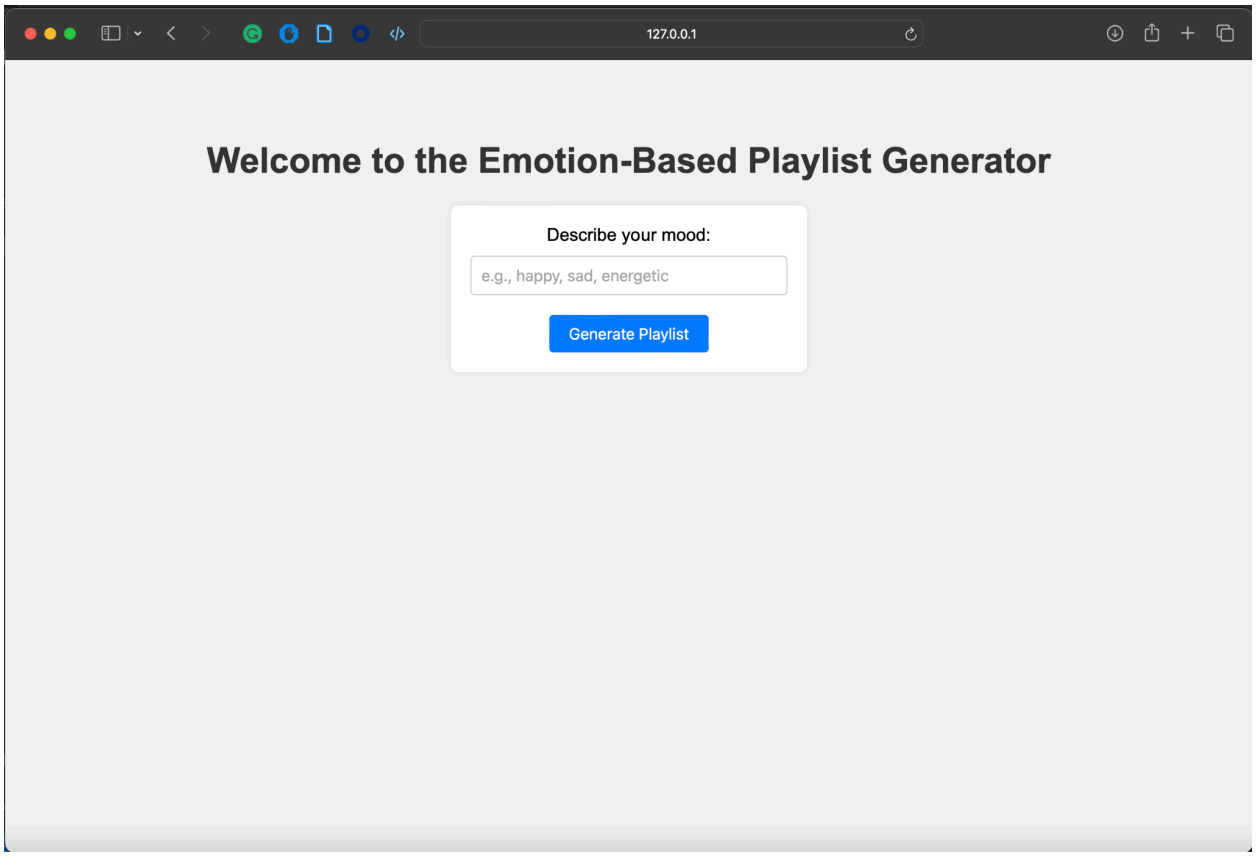

Fig 1: The homepage where the users land after opening the webpage

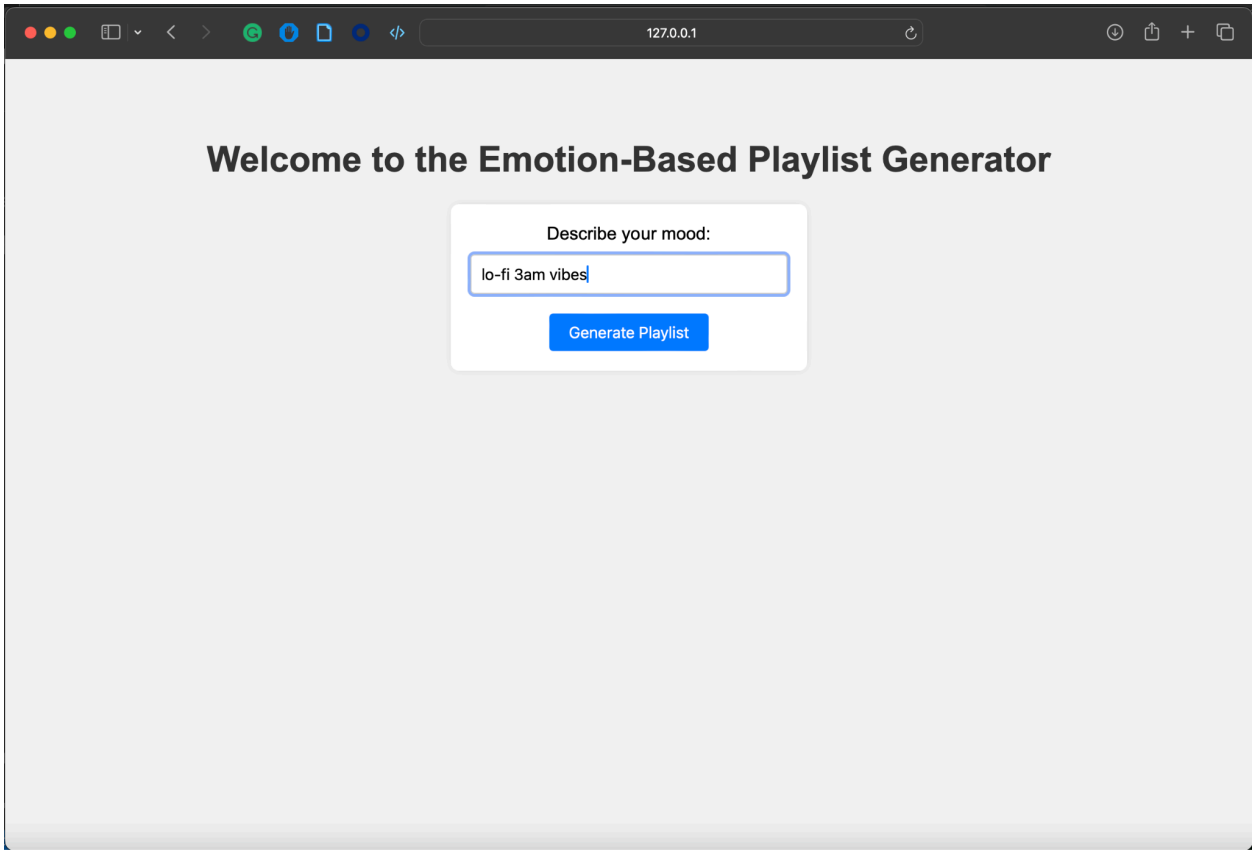

Fig 2: Users enter their mood.

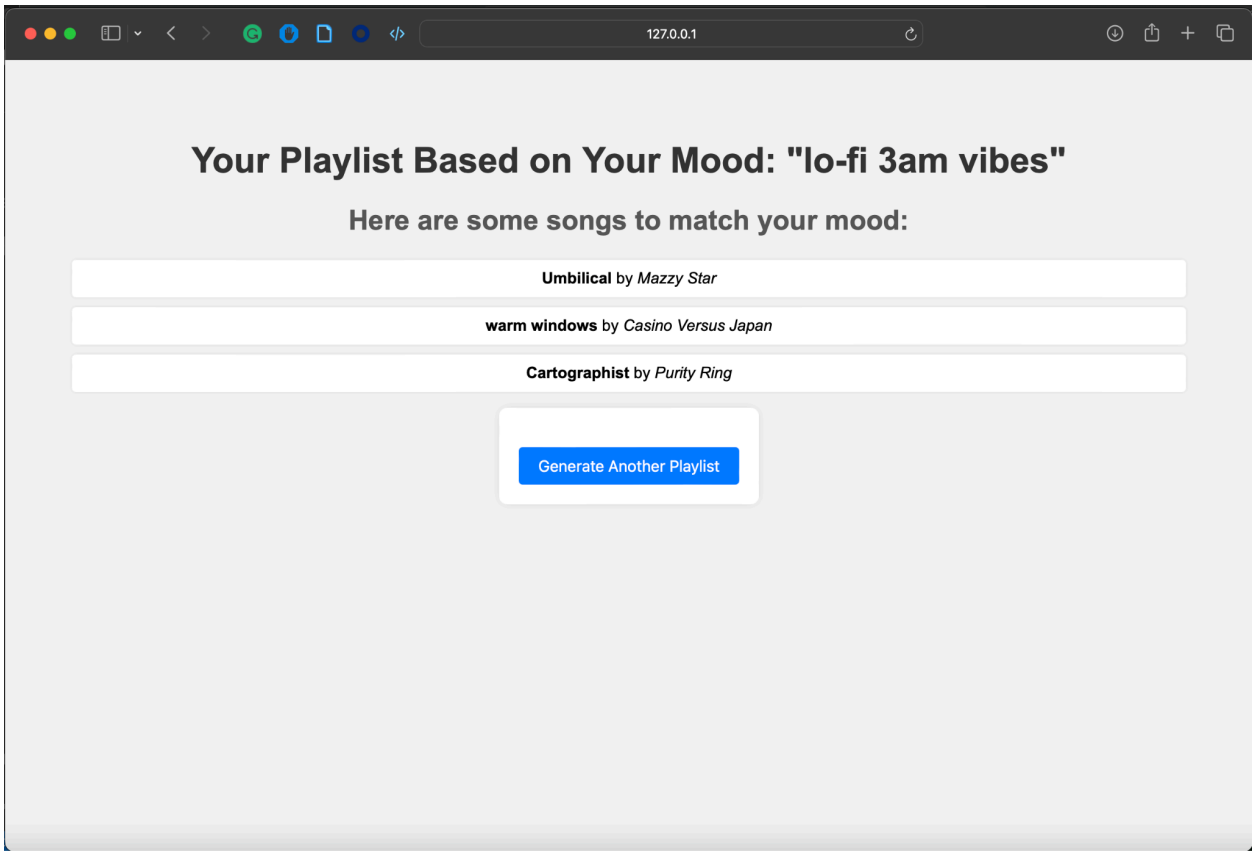

Fig 3: The generated playlist showing song titles and artist names on the results page

3. Issues & Solutions

Issues Encountered:

- **Issue 1**: The initial challenge was integrating the **Musical Sentiment Dataset** into the Flask application. The dataset contained multiple columns, and I needed to correctly map the sentiment analysis results (polarity) to the **valence_tags**.
    - **Solution**: I successfully implemented the dataset integration by using `pandas` to load the dataset and match the user's mood with appropriate songs based on the **valence_tags** column.

- **Issue 2**: Formatting the playlist display in a visually clear way was another challenge, as the initial output was just a plain list of song names.

    - **Solution**: To improve readability, I used HTML formatting to display the song names in bold and the artist names in italics. This simple change made the playlist easier to read/scan
      .

Unresolved Issues:

- **Unresolved Issue 1**: We haven't yet incorporated the **arousal_tags** and **dominance_tags** into the playlist generation logic.
    - **Proposed Solution**: In the next version, I plan to expand the playlist generation by using a combination of **valence_tags**, **arousal_tags**, and **dominance_tags** for more accurate playlist recommendations.

4.. Milestones for the Next Weeks

Milestone 1: Full Sentiment Tag Integration (Version 2 Progress Report A)

- **Description**: I will integrate the remaining sentiment tags (**arousal_tags** and **dominance_tags**) into the playlist generation process for a more refined and accurate song selection based on the user's mood. This will involve updating the backend logic to consider all three sentiment dimensions when selecting songs.

Milestone 2: Implementing Error Handling and Optimizing Playlist Generation (Version 2 Progress Report B)

- **Description**: I will add robust error handling for scenarios such as invalid mood input or when no songs match the mood analysis. Additionally, I will optimize the playlist generation algorithm to ensure a more efficient selection of songs, ensuring edge cases are handled properly.

Milestone 3: Final UI and Minor Backend Adjustments (Version 2 Progress Report C)

- **Description**: I will focus on refining the user interface by improving the overall design and layout. Minor backend adjustments, such as ensuring data consistency and smooth user flow, will be made. The majority of this milestone will center around polishing the front-end, enhancing user experience, and adding any final features like playlist formatting, export options or API integration.

5. Self-Reflection

So far, I am satisfied with the progress made in implementing core features such as sentiment analysis and playlist generation. Initially, I anticipated that integrating the dataset with the sentiment analysis might be complex, but it turned out to be manageable with the help of the **pandas** library.

However, I underestimated the complexity of working with multiple sentiment tags (**valence**, **arousal**, and **dominance**), and this has proven to be more challenging than expected. I expect that integrating these tags in future iterations will require more effort, but I am confident that it can be done.

In terms of project completion, I believe the project is on track, and I should be able to finish within the given timeframe, provided I manage my milestones effectively.