

Final Report - Melodic Music Application

a) Project Introduction

Melodic is a desktop music application designed to offer users a personalized music listening experience. It allows users to listen to songs stored locally, search and stream music from YouTube and Spotify, and receive music recommendations based on mood detection. The goal of this project is to create an engaging and user-friendly platform that combines media playback with smart mood-based suggestions.

b) User Stories (All Sprints)

Sprint 1:

- As a user, I want to create an account and log in securely.
- As a user, I want to play, pause, and skip music from my local library.

Sprint 2:

- As a user, I want to see song recommendations based on my mood.
- As a user, I want to use a GUI interface to interact with the app.
- As a user, I want to have a mood-detection module integrated into the GUI.

Sprint 3:

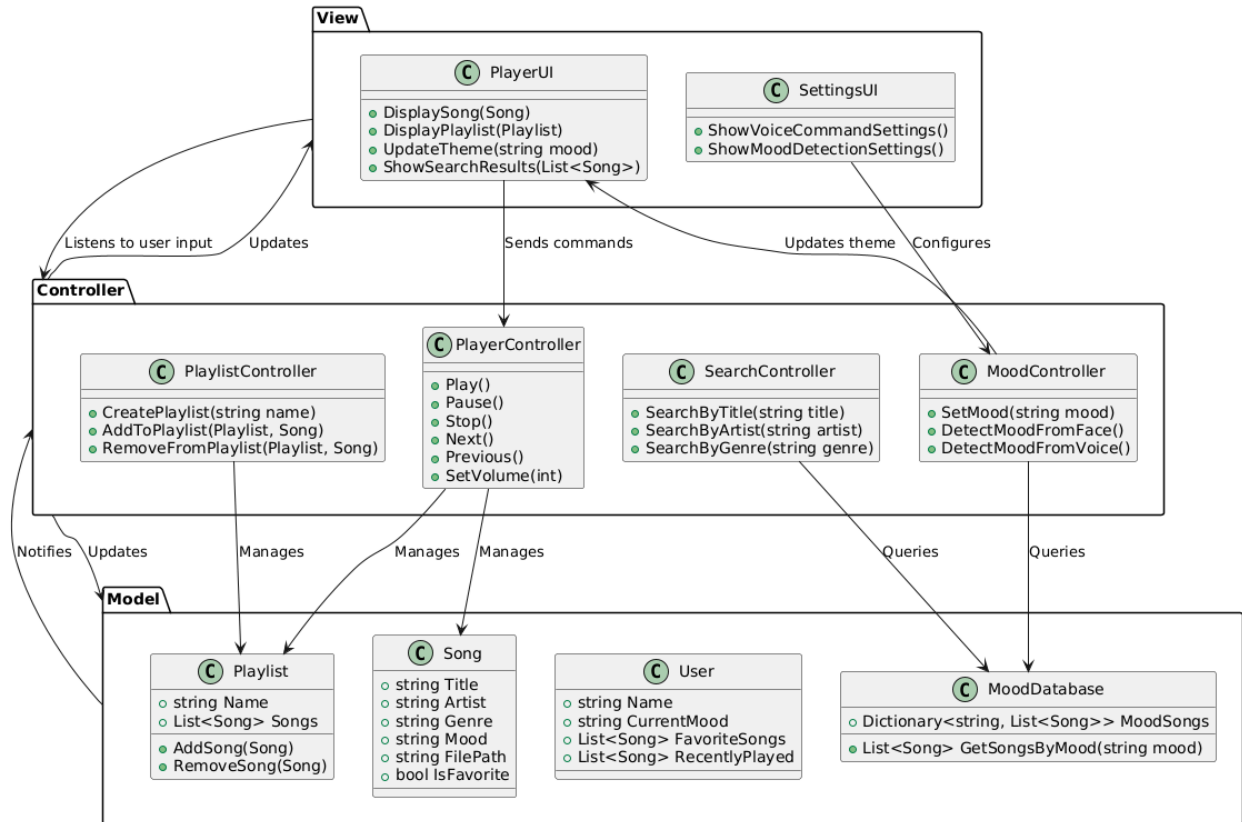
- As a user, I want to search and stream songs from YouTube and Spotify.
 - As a user, I want to receive better music suggestions based on my mood history.
-

c) Design (All Sprint 2 Items)

- GUI Design using Python Tkinter for login/signup, player controls, and mood detection interface.
 - Integration of the mood detection module with the main player GUI.
 - Display of current song, artist name, and album cover.
-

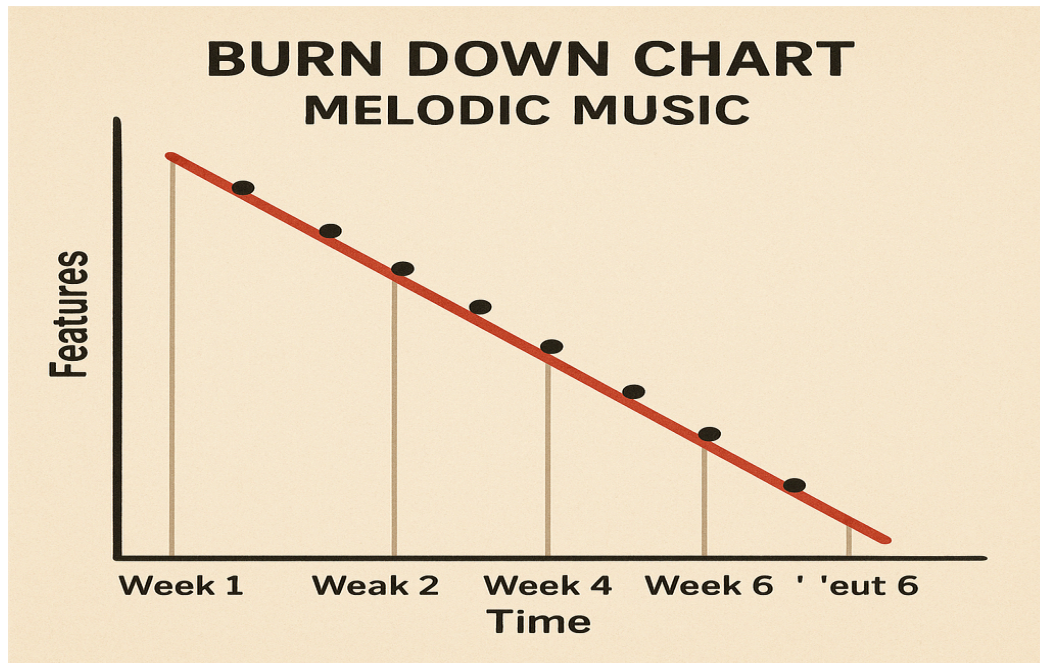
d) Architecture

The application uses the **Model-View-Controller (MVC)** architecture pattern.

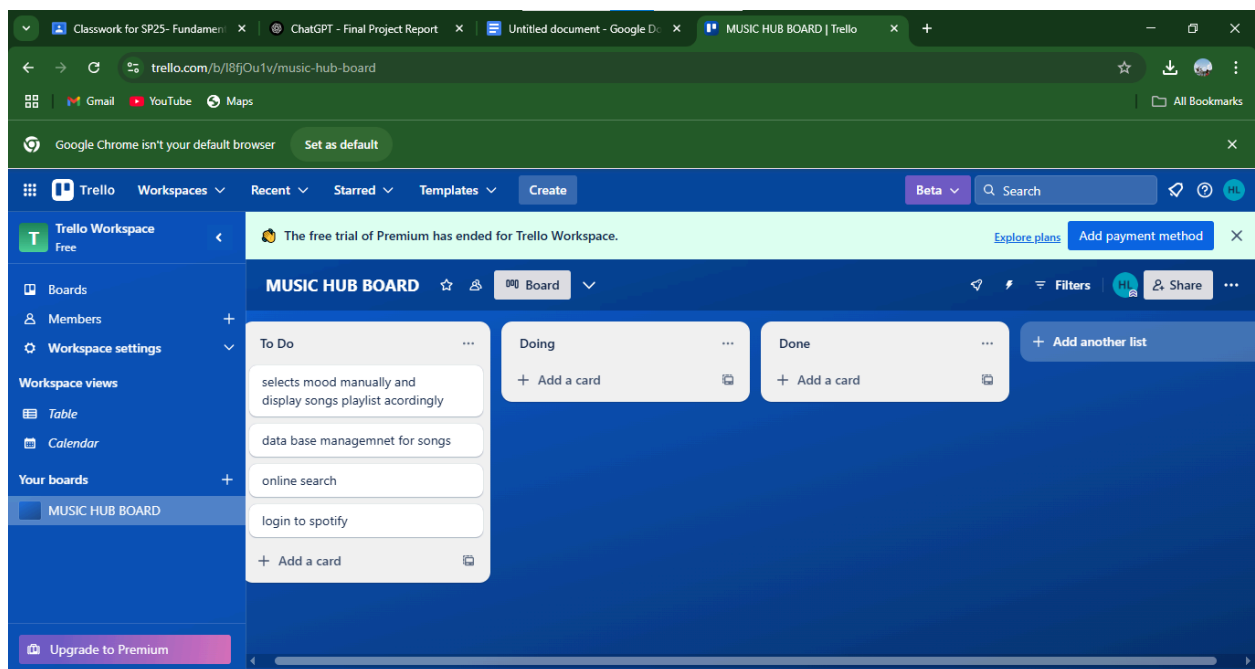


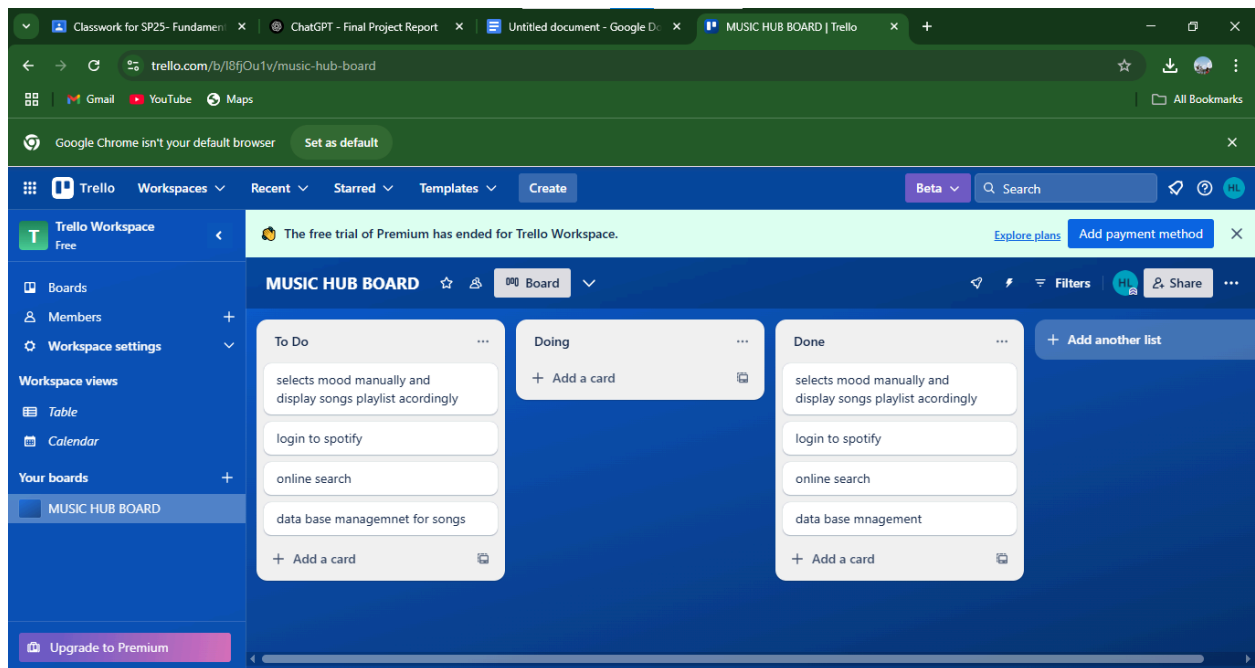
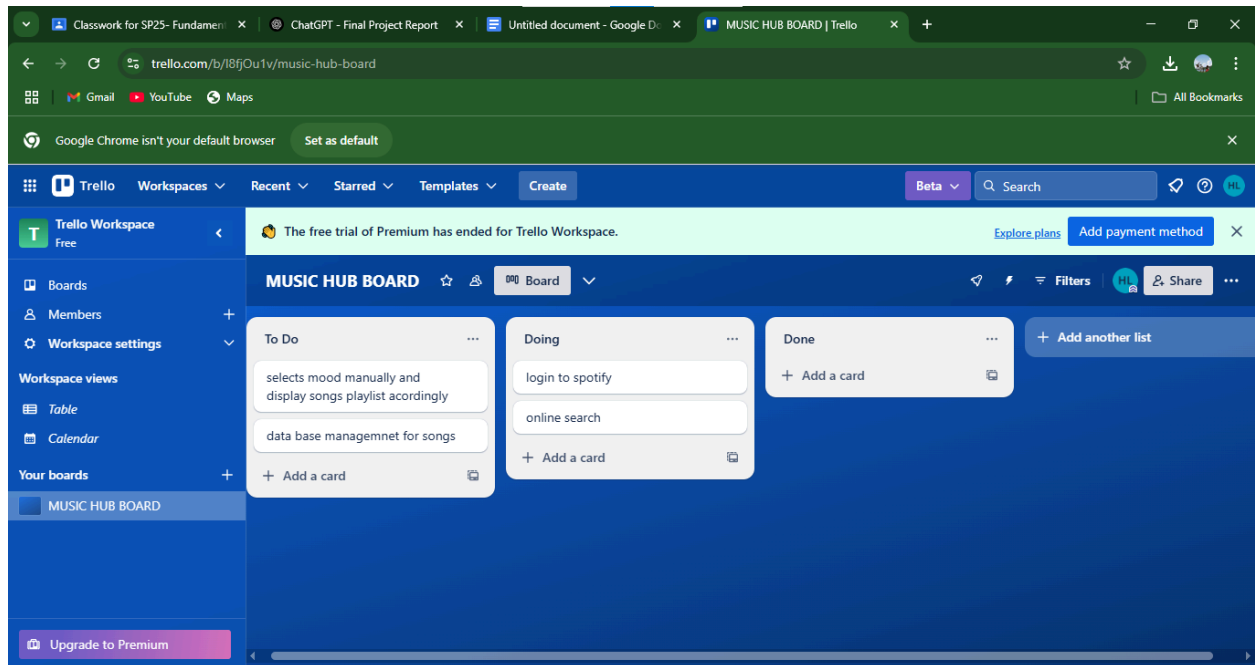
e) Actual Implementation Screenshots

f) Product Burn Down Chart



g) Trello Board Screenshots





i) Work Division Between Group Members

Member Name Tasks Assigned

Member 1 GUI design, Mood detection integration

Member 2	Local music player, Spotify API integration
Member 1	YouTube streaming module, GUI refinement
Member 2	Testing, Trello management, Documentation

j) Lessons Learnt By Group

- Importance of planning each sprint in detail before starting.
 - Realized the value of version control (GitHub) for collaboration.
 - Learned to integrate multiple APIs (Spotify, YouTube) efficiently.
 - Understood how to follow an architecture pattern (MVC) for better organization.
 - Improved communication skills and task distribution using Trello.
-

Implementation screen shots

