

Software Design Document (SDD)

AI Music Recommender System

Snapshot 2 - Mood-Based Recommendations

Project Group 5

December 7, 2025

Version Description

Version	Description	Date
2.0	SDD updated for Snapshot 2. Extends architecture and database design for mood-based recommendations.	December 7, 2025

1 Introduction

1.1 Purpose

This document updates the Snapshot 1 design to incorporate mood-based recommendations, including new UI elements, backend logic, and database structures.[web:16][web:18]

1.2 Scope

Snapshot 2 modifies the recommendation workflow to accept a mood parameter, adds storage for mood preferences, and updates the client to support mood selection.

2 Architecture Changes

2.1 Updated Workflow

1. The user logs in and views the home screen.
2. The user selects a mood (e.g., Happy, Chill, Focus).
3. The client calls the recommendation API with the user identifier and selected mood.

4. The backend queries song features or tags associated with that mood and the user's interaction history.
5. The recommendation engine ranks candidate songs using a mood-aware scoring function and returns a playlist.
6. The user plays songs and provides like/skip feedback, which is stored with the active mood label.

2.2 New/Extended Components

2.2.1 Client Application

- Adds a mood selector UI element on the home screen.
- Displays a “Mood Playlist” section populated by mood-aware recommendations.

2.2.2 Backend Services

- Extends the recommendation API to accept an optional mood parameter.
- Adds a mood preference handler that records the user's selected mood.

2.2.3 Recommendation Engine

- Incorporates mood metadata (e.g., mood labels, energy, valence) into the ranking algorithm.[web:13][web:18]
- Applies filters and boosts for songs whose mood attributes align with the selected mood.

3 Database Design Updates

3.1 New/Modified Tables

- MoodPreference(userId, mood, selectedAt).
- Optional mood-related attributes added to Track, such as moodLabel or energy/valence fields.
- Interaction table extended with an optional mood column to record the active mood at the time of interaction.

3.2 Example Relationships

- One User to many MoodPreference records.
- One Track may be associated with one or more mood labels via attributes or auxiliary tables.

4 Interface Design

4.1 API Changes

- GET /songs/recommended?mood={mood}: returns a mood-aware recommendation list.
- POST /moods/select: stores the user's selected mood for the current session.

5 Design Considerations

- Mood-based logic is designed to be modular, allowing additional moods or a more advanced mood detection model in future snapshots.[web:16]
- Default behavior falls back to Snapshot 1 recommendations when no mood is selected.