Software Design Document (SDD)
AI Music Recommender System
Snapshot 4 - Final Release and Future Work

Project Group 5

December 7, 2025

Version Description

Version	Description	Date
4.0	Final SDD for Snapshot 4. Adds Quick Mix flow, performance refinements, and design notes for future work.	December 7, 2025

1 Introduction

1.1 Purpose

This document refines the design for Snapshot 4 by adding a Quick Mix feature, performance-related improvements, and architectural notes that support future enhancements.

1.2 Scope

Snapshot 4 does not introduce a major new subsystem but extends existing modules and clarifies design decisions for maintainability and future growth.

2 Architecture Updates

2.1 Quick Mix Workflow

1. The user triggers Quick Mix from the main screen.
2. The client calls a Quick Mix endpoint with the user identifier and context (e.g., current mood, recent interactions).
3. The backend collects candidate tracks from content-based, mood-based, and collaborative sources.

4. The recommendation engine combines these sources into a hybrid ranked playlist.[web:14][web:19]
5. The backend returns the playlist, and the client starts playback.

2.2 Caching and Optimization (Optional)

- Frequently requested recommendation lists (e.g., top mood playlists) may be cached briefly to reduce load.
- Database indexes may be added on key fields (userId, trackId, actionType, mood, follow relationships) to improve query performance.

3 Interface Design

3.1 API Additions

- GET /songs/quick-mix: returns a hybrid playlist combining multiple recommendation strategies.
- Optional: endpoints to expose diagnostic data for debugging recommendation issues (for internal use only).

4 Design Considerations and Future Work

4.1 Design Decisions

- The system uses a hybrid design combining content-based, mood-based, and collaborative filtering to leverage strengths of different recommendation techniques.[web:14][web:17]
- Separation of concerns is maintained through distinct modules for recommendation, mood handling, and social features.

4.2 Future Work Support

- The architecture is designed to accommodate more advanced ML models and context-aware features without major structural changes.[web:13][web:16]
- Additional services can be added (e.g., analytics, explanation service) using the same API-driven modular approach.