

Software Requirements Specification (SRS)

AI Music Recommender System

Snapshots 1–4

Version 4.0

Project Group 5

David Nazaryan, Eric Castellon, Gustavo Trejo, Rojina Zalzar

December 7, 2025

Contents

Version Description	3
1 Snapshot 1 – Core System	4
1.1 Introduction	4
1.1.1 Purpose	4
1.1.2 Scope	4
1.1.3 Intended Audience	4
1.1.4 Overview of the Software	4
1.2 Overall Description	4
1.2.1 User Needs	4
1.2.2 Product Perspective	5
1.3 External Interface Requirements	5
1.3.1 User Interface	5
1.3.2 Software Interfaces	5
1.3.3 Hardware Interfaces	5
1.4 Functional Requirements	5
1.4.1 Authentication and User Management	5
1.4.2 Music Catalog and Playback	6
1.4.3 Recommendation and Interaction Logging	6
1.4.4 Search	6
1.5 Non-Functional Requirements	6
1.5.1 Performance	6
1.5.2 Usability	6
1.5.3 Security	7
1.6 Legal and Ethical Considerations	7
1.6.1 Data Storage and Privacy	7
1.6.2 User Consent and Data Protection	7
2 Snapshot 2 – Mood-Based Recommendations	7
2.1 Introduction	7
2.1.1 Purpose	7
2.1.2 Scope	7
2.1.3 Intended Audience	7
2.2 New and Updated Requirements	8
2.2.1 User Interface Extensions	8
2.2.2 Mood-Based Recommendation	8
2.2.3 Interaction and Data Storage	8
2.2.4 Search and Navigation	8
2.3 Non-Functional Requirements (Snapshot 2)	8
2.3.1 Performance	8
2.3.2 Usability	9
2.4 Legal and Ethical Considerations	9

3 Snapshot 3 – Social and Collaborative Recommendations	9
3.1 Introduction	9
3.1.1 Purpose	9
3.1.2 Scope	9
3.2 New Functional Requirements	9
3.2.1 Follow System	9
3.2.2 Friends' Picks and Collaborative Recommendations	9
3.2.3 Privacy and Control	10
3.3 Non-Functional Requirements (Snapshot 3)	10
3.3.1 Performance	10
3.3.2 Usability	10
3.4 Legal and Ethical Considerations	10
4 Snapshot 4 – Final Release and Future Work	10
4.1 Introduction	10
4.1.1 Purpose	10
4.1.2 Scope	11
4.2 New Functional Requirements	11
4.2.1 Quick Mix Feature	11
4.2.2 Refinement and Quality	11
4.3 Non-Functional Requirements (Snapshot 4)	11
4.3.1 Performance and Reliability	11
4.3.2 Usability	11
4.4 Future Work	12
Glossary	12

Version Description

Version	Description	Date
1.0	Initial SRS for Snapshot 1 of the AI Music Recommender System. Core functionality: authentication, basic playback, interaction logging, and baseline recommendations.	December 7, 2025
2.0	SRS updated for Snapshot 2. Adds mood-based recommendation feature and corresponding requirements.	December 7, 2025
3.0	SRS updated for Snapshot 3. Adds social and collaborative recommendation features such as following users and friends' picks.	December 7, 2025
4.0	Final SRS for Snapshot 4. Adds Quick Mix feature, refinement requirements, and explicit future work.	December 7, 2025

1 Snapshot 1 – Core System

1.1 Introduction

1.1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the functional and non-functional requirements for Snapshot 1 of the AI Music Recommender System, which suggests songs and playlists based on users' listening history, preferences, and interactions.[file:1]

1.1.2 Scope

Snapshot 1 delivers a minimal viable product as a web-based or mobile application that allows users to register and authenticate, browse a catalog, play songs, record likes and skips, and receive simple personalized recommendations derived from interaction history stored in the project database.[file:1]

1.1.3 Intended Audience

The intended audience for this SRS includes the course instructor and teaching assistants evaluating the project, Project Group 5 members implementing the software, and future developers or maintainers who need to understand the required behavior of the system for Snapshot 1.[file:1]

1.1.4 Overview of the Software

The AI Music Recommender System in Snapshot 1 focuses on core flows: authentication, music playback, interaction logging, and baseline recommendation generation using past user behavior.[file:1]

1.2 Overall Description

1.2.1 User Needs

Users need:

- A simple way to create an account and sign in securely.
- A music player to browse and play songs from a catalog.
- Personalized recommendations aligned with previous listening behavior.
- A means to express preferences (likes and skips) that influence future recommendations.

[file:1]

1.2.2 Product Perspective

The system operates as a standalone application with a client, backend server, and database layer, where the music catalog for Snapshot 1 is predefined and stored in the project database.[file:1]

1.3 External Interface Requirements

1.3.1 User Interface

Snapshot 1 shall provide:

- A login and registration screen for user authentication.
- A home screen that displays a list of recommended songs and basic playlists.
- A search screen to search by song title, artist, or genre.
- A now-playing screen with play, pause, next, and previous controls, plus like and skip buttons.

[file:1]

1.3.2 Software Interfaces

The client communicates with:

- A backend API for registration, login, recommendation retrieval, catalog browsing, and interaction logging.
- A database storing user accounts, songs, artists, and interaction logs.

[file:1]

1.3.3 Hardware Interfaces

The system targets standard desktop or mobile devices with network connectivity and does not require special hardware.[file:1]

1.4 Functional Requirements

1.4.1 Authentication and User Management

- FR-1: The system shall allow a new user to create an account using email and password.
- FR-2: The system shall allow an existing user to log in with registered credentials.
- FR-3: The system shall validate credentials and reject invalid login attempts.

[file:1]

1.4.2 Music Catalog and Playback

- FR-4: The system shall display a list of available songs with title, artist, and basic metadata.
- FR-5: The system shall allow a user to start playback of a selected song.
- FR-6: The system shall provide playback controls for play, pause, next, and previous.

[file:1]

1.4.3 Recommendation and Interaction Logging

- FR-7: The system shall display a list of recommended songs on the home screen for the logged-in user.
- FR-8: The system shall allow users to like a currently playing song.
- FR-9: The system shall allow users to skip a song and automatically move to the next track.
- FR-10: The system shall log interactions, including plays, likes, and skips, associated with the user.
- FR-11: The system shall use the logged interactions to adjust future recommendations.

[file:1]

1.4.4 Search

- FR-12: The system shall allow users to search for songs by title, artist, or genre.
- FR-13: The system shall return a list of matching songs that can be played.

[file:1]

1.5 Non-Functional Requirements

1.5.1 Performance

- NFR-1: The system should load the home screen and recommended songs within a few seconds under normal network conditions.

[file:1]

1.5.2 Usability

- NFR-2: Navigation among login, home, search, and now-playing screens shall be intuitive and consistent.

[file:1]

1.5.3 Security

- NFR-3: Passwords shall be stored using secure hashing rather than plain text.
- NFR-4: Only authenticated users shall access personalized recommendations and interaction history.

[file:1]

1.6 Legal and Ethical Considerations

1.6.1 Data Storage and Privacy

The system stores user account information and listening history solely to provide and improve recommendations, limiting access to authorized components and ensuring passwords are stored in hashed form.[file:1]

1.6.2 User Consent and Data Protection

Users are informed that their interaction data (likes, skips, and plays) is collected and used to personalize recommendations, and that the project avoids sharing personal or identifiable user data with third parties.[file:1]

2 Snapshot 2 – Mood-Based Recommendations

2.1 Introduction

2.1.1 Purpose

This snapshot extends Snapshot 1 by specifying requirements for mood-based recommendations, allowing users to select a mood and receive playlists tailored to that mood using predefined or learned song features.[web:8]

2.1.2 Scope

Snapshot 2 adds mood selection to the user interface, mood-aware recommendation logic, and storage of user mood choices, while preserving all Snapshot 1 functionality.[web:8]

2.1.3 Intended Audience

The audience includes the course instructor, teaching assistants, and project team members implementing and testing the mood-based feature.[web:8]

2.2 New and Updated Requirements

2.2.1 User Interface Extensions

- FR-M1: The system shall provide a mood selector on the home screen (e.g., Happy, Chill, Focus, Energetic).
- FR-M2: The system shall display a “Mood Playlist” section showing songs tailored to the selected mood.

[web:8]

2.2.2 Mood-Based Recommendation

- FR-M3: The system shall allow a user to change the current mood at any time from the home screen.
- FR-M4: When the user selects a mood, the system shall generate a playlist of songs associated with that mood using stored mood features or tags.
- FR-M5: The system shall use both mood and the user’s interaction history when ordering songs in the mood playlist.

[web:8][web:12]

2.2.3 Interaction and Data Storage

- FR-M6: The system shall store the user’s most recent mood selections for future recommendation tuning.
- FR-M7: The system shall log plays, likes, and skips that occur while a mood is active, including the active mood label.

[web:8]

2.2.4 Search and Navigation

- FR-M8: The user shall be able to navigate between the regular recommendation list and the mood playlist.
- FR-M9: The user shall still be able to use search independent of mood selection.

[web:8]

2.3 Non-Functional Requirements (Snapshot 2)

2.3.1 Performance

- NFR-M1: Updating the mood playlist after changing the selected mood should complete within a few seconds under normal conditions.

[web:8]

2.3.2 Usability

- NFR-M2: The mood selector shall be clearly visible and easy to understand (e.g., labeled buttons or icons).

[web:8]

2.4 Legal and Ethical Considerations

Mood data is treated like other interaction data and is used only to personalize recommendations, following the same privacy principles defined for Snapshot 1 and informing users that mood selections are stored for personalization.[web:8]

3 Snapshot 3 – Social and Collaborative Recommendations

3.1 Introduction

3.1.1 Purpose

This snapshot extends Snapshots 1 and 2 by specifying social and collaborative recommendation features, allowing users to follow other users and receive recommendations influenced by community listening patterns.[web:12]

3.1.2 Scope

Snapshot 3 adds follow relationships, friends' playlists, and collaborative recommendation logic while maintaining all previously implemented features.[web:12]

3.2 New Functional Requirements

3.2.1 Follow System

- FR-S1: The system shall allow a logged-in user to follow another user.
- FR-S2: The system shall allow a user to unfollow a previously followed user.
- FR-S3: The system shall display a list of users that the current user follows.

[web:12]

3.2.2 Friends' Picks and Collaborative Recommendations

- FR-S4: The system shall display a “Friends’ Picks” or similar section with songs liked by followed users.
- FR-S5: The system shall update the “Friends’ Picks” section when followed users like new songs.

- FR-S6: The system shall use collaborative information (e.g., similar users' likes) to supplement content-based and mood-based recommendations.

[web:12]

3.2.3 Privacy and Control

- FR-S7: The system shall only share non-sensitive activity (e.g., liked songs) as part of social features.
- FR-S8: The system shall allow users to view which parts of their activity are visible to followers.

[web:12]

3.3 Non-Functional Requirements (Snapshot 3)

3.3.1 Performance

- NFR-S1: Generating the “Friends’ Picks” list should complete within a few seconds for a typical number of followed users.

[web:12]

3.3.2 Usability

- NFR-S2: Follow and unfollow controls shall be clearly labeled and accessible from user profile or recommendation screens.

[web:12]

3.4 Legal and Ethical Considerations

Social features introduce additional privacy considerations, and the system shall inform users that their liked tracks may be visible to followers while avoiding exposure of personal identifiers beyond the project requirements.[web:12]

4 Snapshot 4 – Final Release and Future Work

4.1 Introduction

4.1.1 Purpose

This snapshot finalizes requirements by adding a Quick Mix feature and documenting refinement, quality, and future work for the AI Music Recommender System.[web:11][web:12]

4.1.2 Scope

Snapshot 4 focuses on polishing existing functionality, adding a lightweight new feature, and clarifying requirements that improve usability and performance while retaining all previous features.[web:11]

4.2 New Functional Requirements

4.2.1 Quick Mix Feature

- FR-F1: The system shall provide a “Quick Mix” button that generates a mixed playlist using the user’s recent history, current mood (if any), and social signals.
- FR-F2: The system shall start playing the first track of the Quick Mix playlist automatically after it is generated.

[web:11]

4.2.2 Refinement and Quality

- FR-F3: The system shall allow users to refresh a recommendation section (e.g., mood playlist, friends’ picks) to request a new set of songs.
- FR-F4: The system shall provide basic error messages and feedback when recommendation or network requests fail.

[web:11]

4.3 Non-Functional Requirements (Snapshot 4)

4.3.1 Performance and Reliability

- NFR-F1: Generating a Quick Mix playlist shall complete within a few seconds under normal usage.
- NFR-F2: The system shall handle typical concurrent usage scenarios without noticeable degradation, within project constraints.

[web:11]

4.3.2 Usability

- NFR-F3: Key actions (play, like, skip, mood change, Quick Mix) shall require at most a few taps or clicks from the main screen.

[web:11]

4.4 Future Work

Potential future improvements include:

- Integration of more advanced machine learning models for music recommendation and mood detection.
- Support for richer context signals such as time of day, activity, or device location.
- Deeper social features such as collaborative playlist editing and comments.
- Enhanced explainability for recommendations (e.g., showing why a song was recommended).

[web:8][web:11][web:12]

Glossary

Acronym	Meaning
UI	User Interface
API	Application Programming Interface
DB	Database
SRS	Software Requirements Specification
SDD	Software Design Document
ML	Machine Learning