# Music Recommendation System

## 1. Introduction

### 1.1 Purpose

The purpose of this document is to define the requirements for the development of a Music Recommendation System. By using music recommender system, the music provider can predict and then offer the appropriate songs to their users based on the characteristics of the music that has been heard previously.

### 1.2 Scope

The Music Recommendation System will provide personalized music recommendations based on user preferences, history, and other relevant factors.

Current system mainly focuses on doing the recommendation on content based features that is based on its acoustic parameters. But in future hybrid model can be created that uses both content based and collaborative(user history, feedback, liking etc.) for recommendations.

## 2. Overall Description

### 2.1 Product Perspective

The Music Recommendation System will operate as a standalone application with the ability to integrate with external music streaming platforms through APIs.

### 2.2 Product Features

- **User Registration and Authentication:**
  - Users should be able to register and log in securely.
- **User Profile:**
  - Users can create and manage their profiles, including preferences and favorite genres.
- **Music Recommendation Engine:**
  - The system will employ a recommendation algorithm to suggest music based on user preferences, listening history, and trends.
- **Search and Browse:**
  - Users can search for specific songs, artists, or genres and browse recommended playlists.
- **Playlist Creation and Management:**
  - Users can create, edit, and delete playlists.
- **Integration with External Platforms:**
  - The system should be able to integrate with popular music streaming platforms using their APIs.

### 2.3 Operating Environment
The system should be compatible with major web browsers (Chrome, Firefox, Safari) and mobile platforms (iOS, Android).

## 3. Specific Requirements
### 3.1 External Interface Requirements
#### 3.1.1 User Interfaces
- The user interface should be intuitive and responsive.
- Include screens for user registration, login, profile management, music search, and playlist management.

#### 3.1.2 Software Interfaces
- Integrate with external music streaming platforms such as Spotify, Apple Music, and others using their APIs.

### 3.2 Functional Requirements
- **User Registration and Authentication:**
  - Users can register with a valid email address.
  - Passwords must be securely stored and encrypted.
  - Users can log in using their credentials.
- **User Profile:**
  - Users can set and update their music preferences.
  - Profile information should include user bio, profile picture, etc.
- **Music Recommendation Engine:**
  - Implement a recommendation algorithm to suggest music based on user preferences.
  - Consider factors like genre, artist, user history, and trending music.
- **Search and Browse:**
  - Users can search for music by song title, artist, or genre.
  - Browse feature should display recommended playlists and songs.
- **Playlist Creation and Management:**
  - Users can create, edit, and delete playlists.
  - Add or remove songs from playlists.
- **Integration with External Platforms:**
  - Connect to external music streaming platforms via APIs.
  - Fetch user data and recommendations from integrated platforms.

### 3.3 Non-Functional Requirements
- **Performance:**
  - The system should provide quick response times for user interactions.
  - Handle a large number of concurrent users.
- **Security:**
  - Implement secure authentication and authorization mechanisms.
  - Protect user data and ensure secure communication.
- **Scalability:**

    - The system should be scalable to accommodate an increasing number of users and data.
- **Reliability:**
    - Ensure the system's reliability to provide continuous service without frequent disruptions.
- **Usability:**
    - The user interface should be user-friendly and accessible.
    - Provide help and guidance features for users.

## 4. Other Requirements

### 4.1 Legal and Compliance Requirements

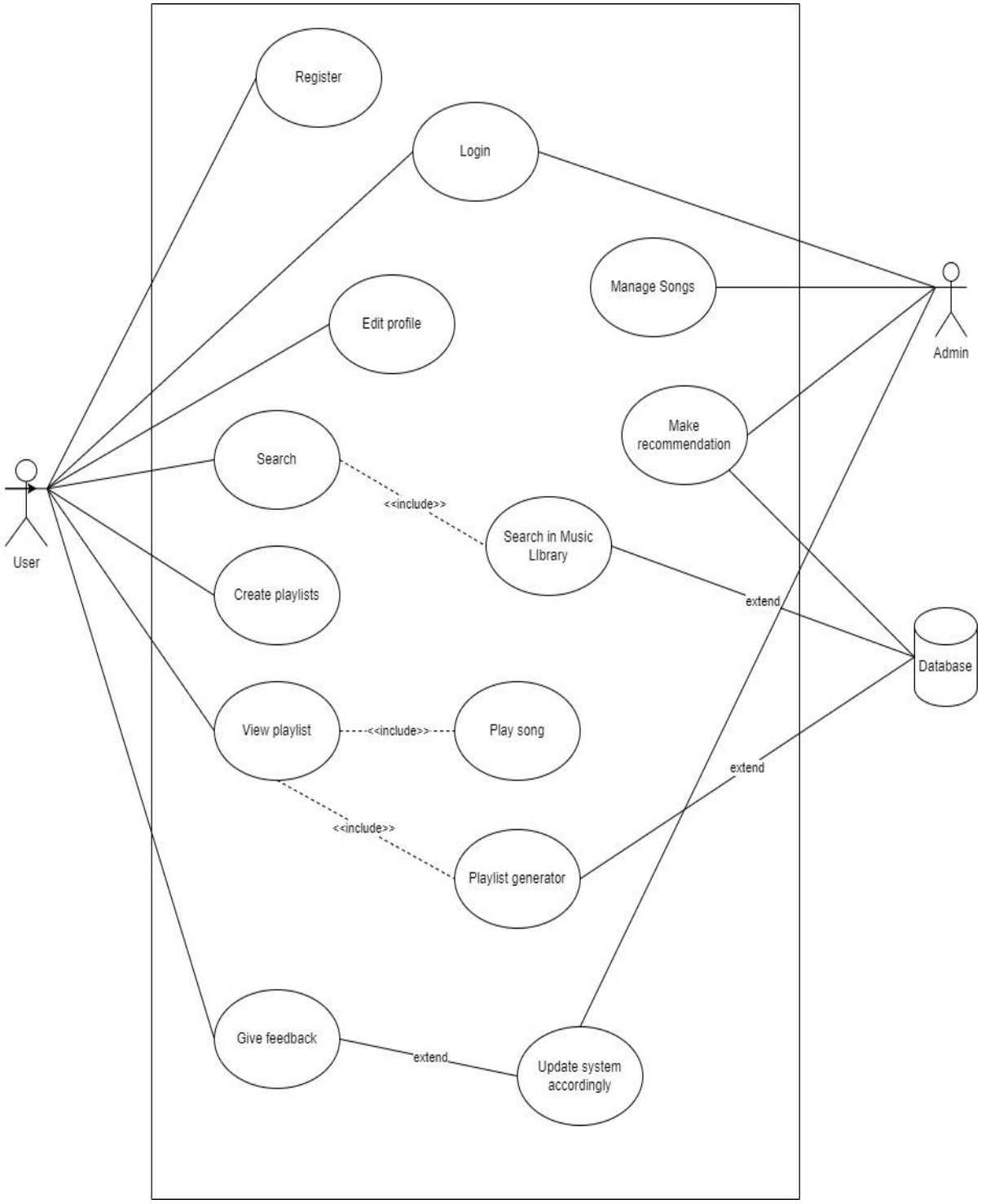
- Comply with data protection regulations and user privacy laws.

### 4.2 Documentation Requirements

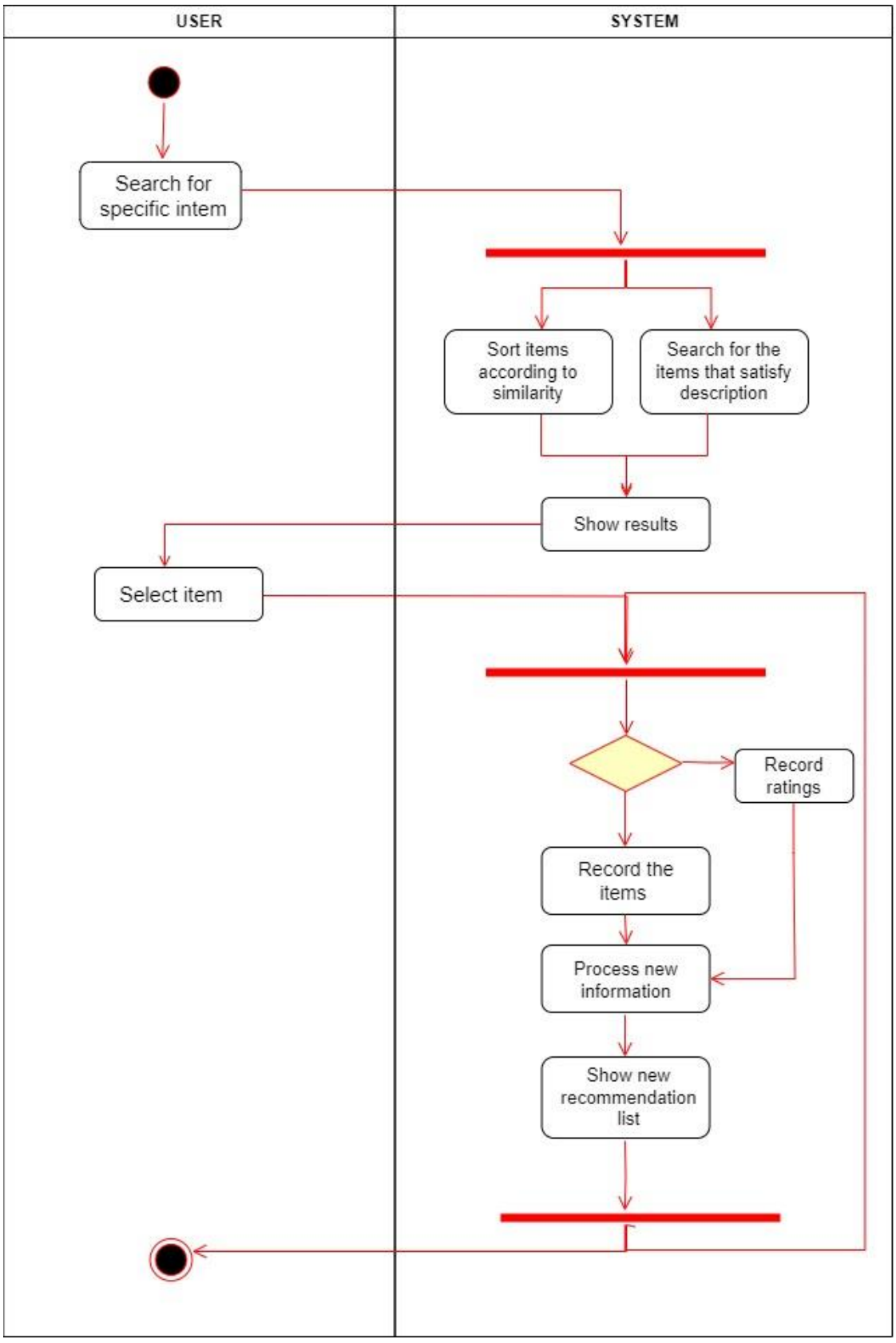
- Provide user documentation for using the system.
- Developer documentation for system maintenance and updates.

## 5. UML Diagrams

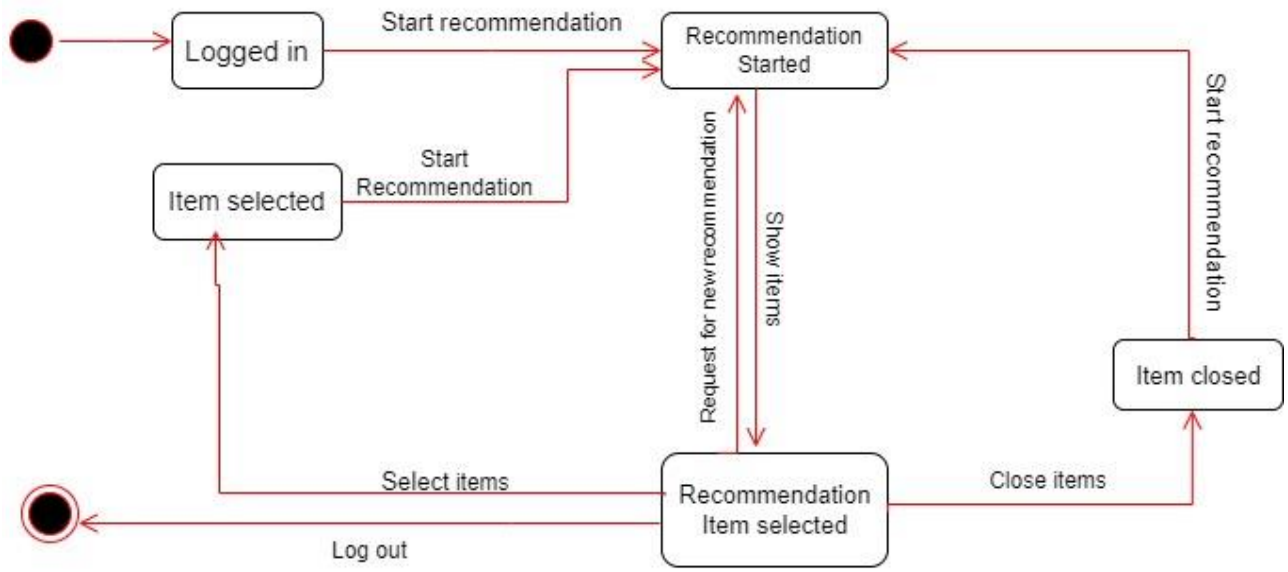
## 5.1 Usecase Diagram



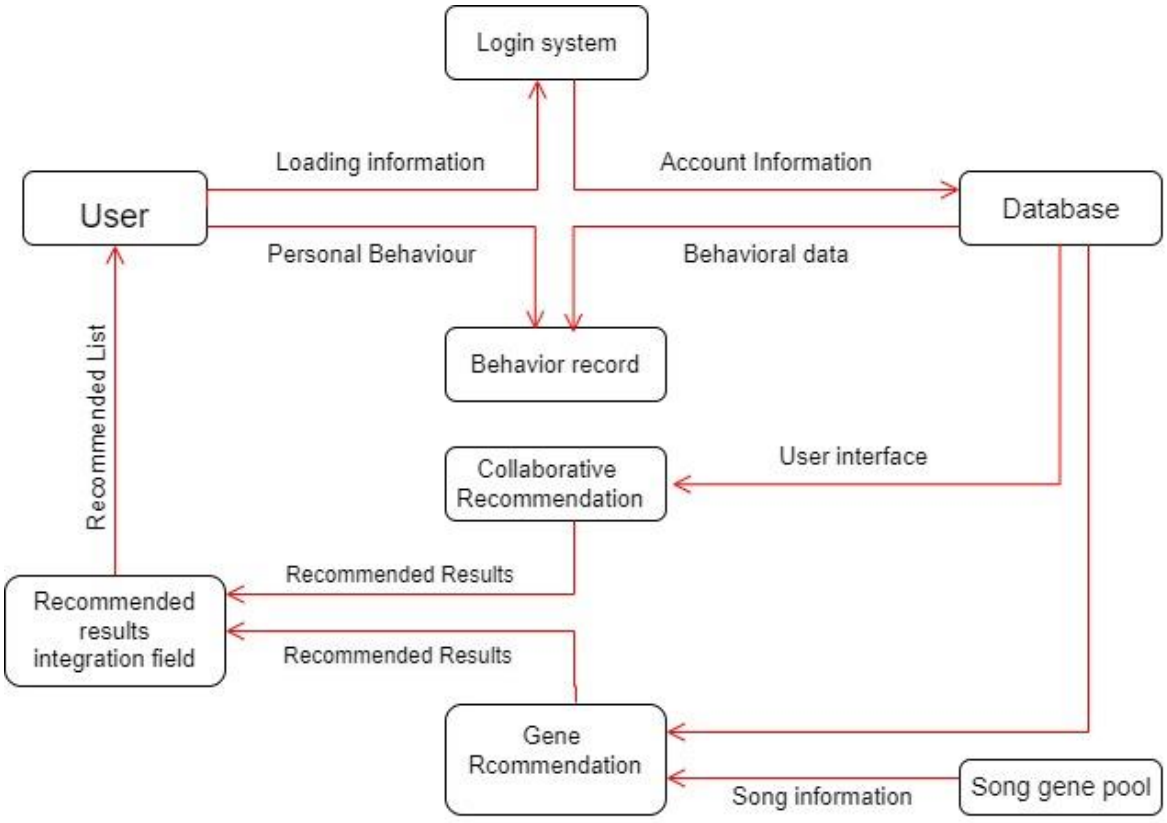
## 5.2 Activity Diagram



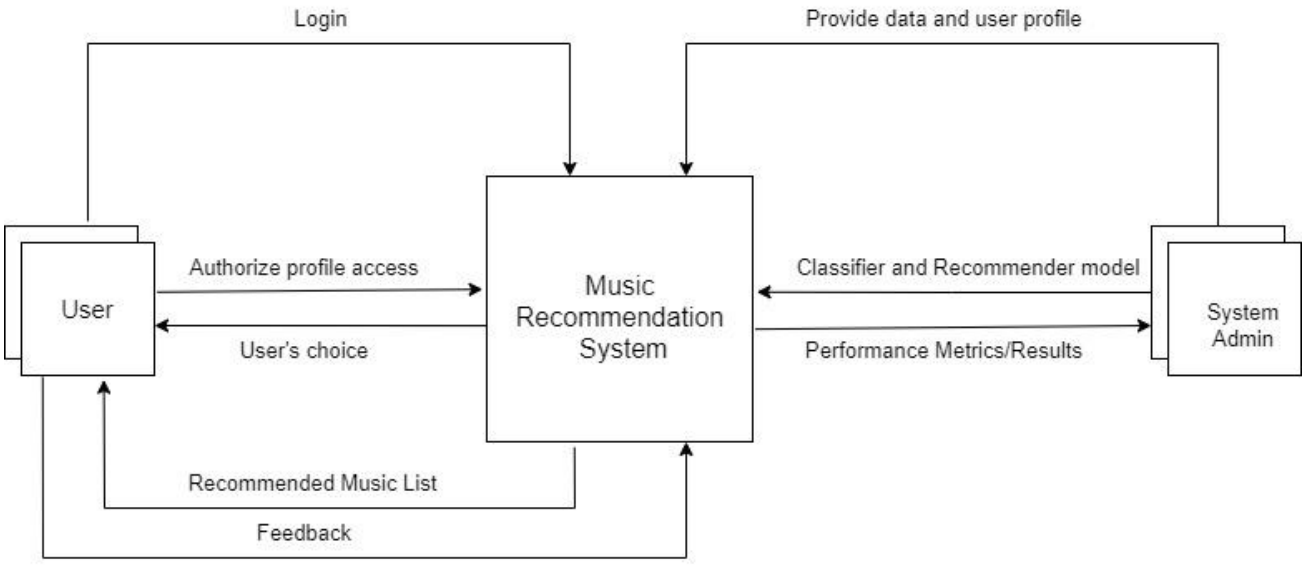

## 5.3 Statechart Diagram




## 5.4 Data Flow Diagram




## 5.5 Context Diagram


Login
Provide data and user profile
User
Authorize profile access
User's choice
Music Recommendation System
Classifier and Recommender model
Performance Metrics/Results
System Admin
Recommended Music List
Feedback