ÇANKAYA UNIVERSITY

Software Requirements Specification

Emo-Tagger- A Community Driven Emotion Annotation Tool

Kübra Begüm Üleş   201611059

Hamza Basil Waheed   201911500

Samed Tuğkan Çinko   201811020

Selin Uzunoğlu   202011041

Serhat Yıldız   202211404

Taha Emre KARATAŞ   202011025

Table of Contents

List of Figures

1. INTRODUCTION

1.1 Purpose

The purpose of this document is to specify the requirements for developing a web application where users label music files based on their emotional impact. This project utilizes crowdsourcing to gather a diverse range of interpretations on the emotional content of music, creating a comprehensive dataset that reflects how different listeners respond emotionally to the same music. Through this platform, we aim to compile a dataset of 10,000 labeled music files, representing a wide spectrum of emotional reactions from users.

To maintain elevated levels of user engagement, the application will incorporate gamification elements, including points, badges, and levels. These features make the labeling process more enjoyable and rewarding, encouraging users to participate actively. Users will be able to track their progress, join challenges, and view their ranking on leaderboards, introducing a competitive and fun element to the experience. This approach aims to foster sustained engagement, contributing to the project's goal of gathering an extensive and accurate set of emotion-based music labels.

1.2 Scope of the Project

The core functionality of this application is to provide a platform where users can listen to curated music tracks and express their emotional responses by selecting from a range of predefined emotion categories. This process aims to capture diverse emotional feedback that will help create a comprehensive dataset, useful for understanding how different types of music affect listeners emotionally. The platform appeals to a wide range of users, from casual listeners seeking self-reflection to music enthusiasts, researchers, and those interested in exploring how music influences emotional well-being.

By allowing users to select from a set of emotion tags such as "Happy," "Sad," "Energetic," or "Relaxed," the application provides a structured, yet user-centered way of recording emotional data. This dataset serves as the foundation for future analysis and research across fields like music therapy, psychology, and data science, offering insights into patterns like common emotional responses to certain genres, tempos, or lyrical themes. It is designed to allow for in-depth exploration of how demographics and personal background may impact emotional responses to various musical elements.[4]

Key features of the app include:

- **Emotion Labeling and Survey Feedback Collection:** Users listen to each music file and choose an emotion that aligns with their experience. Additionally, they have the option to provide more detailed feedback, describing how the song impacted them personally. This data captures not only immediate emotional reactions but also in-depth, nuanced responses that contribute to a richer understanding of music's emotional impact.

- **Gamification Elements:** By integrating points, badges, levels, and leaderboards, the app encourages ongoing participation, making the labeling process more engaging and rewarding. These elements help foster a competitive and fun environment, where users are motivated to track their progress, participate in challenges, and view their standings on leaderboards. Gamification not only boosts engagement but also aids in collecting a larger, more accurate dataset.[2]

- **Interactive User Profiles and Emotion Tagging:** Each user's emotional responses and survey results saved in a personalized profile, which includes interactive elements like mood charts, timelines, and visual graphs. This profile allows users to reflect on their emotional journey through music over time, identifying trends, such as times when they feel more relaxed or energized, or observing preferences for certain genres. Users can filter their emotional data by genre, song, or artist, helping them gain insight into how music impacts their mood.

- **Data Analytics and Reporting for Researchers and Administrators:** Administrators and researchers can use the collected data to perform real-time emotional trend analysis. A comprehensive analytics dashboard enables detailed reporting, with options to view data filtered by demographics, music genre, emotion tags, and more. Advanced tools like heat maps and emotion-based charts provide a visual representation of emotional trends, helping to identify patterns and refine music selection or emotional tags. The data insights derived from this analysis could prove valuable in music therapy, entertainment, advertising, and other fields interested in the psychological impact of music.

- **Personalized Music Recommendation Engine:** Based on users' emotional responses, the app will offer tailored music recommendations. For instance, if a user frequently tags songs as "Energetic" or "Happy," the system will recommend more upbeat tracks that align with these emotions. Conversely, if the user selects "Calm" or "Reflective," they will receive suggestions for more introspective or soothing music. The recommendation engine adapts over time, becoming increasingly personalized as it learns from users' ongoing feedback.

- **Community and Social Features:** To foster a sense of community, the app includes social features where users can connect with friends, share emotional responses, compare profiles, and join discussions on how music influences emotions. Features like shared playlists or collaborative surveys, where users create and share music lists focused on specific emotions (e.g., "Relaxing Tunes" or "Energetic Beats"), add an interactive and social aspect to the platform, enhancing user engagement.

- **Privacy and Data Security:** Given the sensitivity of emotional and personal data, robust security measures are implemented. The application will comply with data privacy regulations (e.g., GDPR), ensuring responsible data handling practices. Users have control over their data and may delete their accounts if desired.[5]

This document provides essential guidelines for developers, project managers, and stakeholders to ensure clarity on the project's purpose, functions, and objectives. It aligns all phases of development to meet user needs and delivers an engaging, user-centered experience. This project aims to gather valuable emotional data and enhance users' understanding of how music affects their emotions.

1.3 Glossary

AI (Artificial Intelligence): The simulation of human intelligence by machines, often used in data analysis, pattern recognition, and emotion prediction. AI enables Emo-Tagger to process and analyze emotional data to improve recommendations.

API (Application Programming Interface): A set of protocols and tools allowing different software applications to communicate and share data, enabling Emo-Tagger to integrate with third-party services like music platforms.

Annotation: The process of labeling or tagging data (in this case, music) with relevant information, such as emotions, to make it useful for machine learning and analytics.

Backend: The server-side part of Emo-Tagger, where data processing, storage, and retrieval are managed. This supports functionalities such as data storage, emotion analysis, and user management.

Crowdsourcing: The process of gathering input from a large group of people online, used in Emo-Tagger to collect emotional labels from a diverse user base, enhancing the emotional dataset and improving machine learning models through varied perspectives.

Community Engagement: A feature allowing users to interact, share, and discuss their emotional responses to music within Emo-Tagger, creating a social and interactive experience.

Classification: In the context of Emo-Tagger, the grouping of music is based on emotions, moods, or other factors for easier recommendation and analysis.

Data Annotation: Specifically for audio and music, this involves categorizing and tagging audio files based on predefined emotional criteria, which becomes valuable training material for AI and ML applications.

Data Privacy: Measures ensuring that user information and emotional responses are protected from unauthorized access, adhering to legal and ethical standards.

Data Repository: A centralized storage within Emo-Tagger that contains emotional responses and related data for analysis and visualization.

DBMS (Database Management System): Software used to create, manage, and interact with databases, where Emo-Tagger's user and survey data is stored and managed.

Emotion Annotation: Tagging songs with specific emotions, such as happiness or sadness, based on user feedback. This helps in understanding and analyzing the emotional effects of music.

Emotion Spectrum: A range of emotions (e.g., joy, calmness, nostalgia) that users can select to describe their reactions to a song, allowing Emo-Tagger to capture complex emotional responses.

Encryption: The process of encoding data to protect it during transmission or storage, ensuring that only authorized users can access sensitive information within Emo-Tagger.

Frontend: The client-side or user-facing part of Emo-Tagger, where users interact with the application through interfaces such as forms, buttons, and visualizations.

FTP (File Transfer Protocol): A network protocol used to transfer files between computers, which may be used in managing Emo-Tagger's server files.

Gamification: Using game-like elements (e.g., points, badges, challenges) in Emo-Tagger to increase user engagement, motivating users to actively participate in the survey and data labeling.

Interface: The visual and interactive components of Emo-Tagger (e.g., buttons, menus, survey forms) that allow users to interact with the application.

JSON (JavaScript Object Notation): A lightweight data format used for exchanging structured data between Emo-Tagger's client and server, making it easier to send user feedback and emotional data.

Leaderboard: A ranking system within Emo-Tagger that displays top contributors, encouraging users to engage more actively by ranking based on contributions like surveys and annotations.

Machine Learning (ML): A subset of AI that enables Emo-Tagger to learn from data and predict trends in emotional responses, supporting emotion analysis and improving recommendation systems.

Metadata: Data about data. For Emo-Tagger, metadata may include song details like title, artist, and genre, which help in organizing and analyzing user responses.

Primary Emotions: Basic emotions such as happiness, sadness, anger, or fear, which are universally recognized and easy to identify.

Profile: A personal section where users can manage their Emo-Tagger account, view past activity, and track engagement metrics.

Real-Time Data Processing: The ability of Emo-Tagger to analyze and process data as soon as it is inputted, allowing users to see immediate feedback or trends.

Registered Users: Individuals who create an account in Emo-Tagger, enabling them to participate in surveys, earn rewards, and access exclusive community features.

Scalability: The ability of Emo-Tagger to expand its system capacity to support a growing user base and data without affecting performance.

Secondary Emotions: More complex emotions, such as nostalgia or serenity, providing a nuanced view of user responses beyond basic emotions.

Sentiment Analysis: The process of analyzing text or audio to determine the emotional tone, potentially used in Emo-Tagger to analyze free-text comments for emotional content.

Survey: A set of questions provided to users to gather feedback on their emotional response to specific songs, creating a foundation for emotional annotations.

Tagging: The process of labeling music with emotional markers, helping Emo-Tagger to categorize songs based on emotional responses for analysis and recommendations.

Trend Graphs: Visual representations of patterns within Emo-Tagger that show how users' emotional responses to songs change over time, allowing insights into popular emotional trends.

UI (User Interface: The visual components of Emo-Tagger, such as buttons, forms, and graphics, that users interact with.

User Experience (UX): The overall experience and satisfaction a user has while interacting with Emo-Tagger, impacted by design, usability, and system performance.

Web Application: The web-based version of Emo-Tagger accessible through browsers, designed to offer similar functionalities to the mobile application for desktop users.

XML (Extensible Markup Language): A markup language used to structure data, often for data interchange between Emo-Tagger and other applications.

1.4 Overview of the Document

The Emo-Tagger project is an innovative and comprehensive music survey application designed to deeply explore and understand the emotional connections users have with music. By allowing users to listen to a variety of songs and tag their emotional responses, the app builds a rich, extensive, and diverse dataset of these reactions. This valuable dataset serves multiple purposes, both personal and academic, enabling detailed study and analysis of how different types of music impact individuals emotionally. The project leverages crowdsourcing techniques to gather a wide array of emotional interpretations from a diverse and varied group of users, thereby enhancing the dataset's accuracy, cultural depth, and richness.

Gamification elements are thoughtfully integrated into the platform to keep users engaged, motivated, and encouraged to contribute regularly, ensuring that the dataset remains robust, comprehensive, and high-quality. The resulting extensive dataset supports significant advancements in AI-driven emotional music analysis, which can be applied in various areas such as personalized music recommendation systems, therapeutic music interventions, and cultural studies. By offering users a platform to reflect on, share, and analyze their emotional responses to music, Emo-Tagger bridges the gap between music and emotions, providing a rich, engaging, and immersive experience for anyone interested in exploring this dynamic and multifaceted relationship. The app not only facilitates personal exploration and discovery but also contributes to academic research and understanding, making it a valuable tool for both individual users and researchers alike.

2. OVERALL DESCRIPTION

2.1 Product Perspective

The Emo-Tagger is a community-driven Emotion Annotation Tool, which is a unique platform designed to explore and analyze the emotional impact of music on listeners. Unlike traditional music rating platforms, Emo-Tagger transcends basic ratings by capturing the nuanced and multidimensional emotions that music evokes. By allowing users to annotate their emotional responses to music, the product facilitates an in-depth exploration of how individual experiences and preferences shape the emotional impact of songs. Emo-Tagger serves as both a survey and analytical tool, featuring a responsive website and mobile applications to provide an accessible and intuitive user experience. These platforms enable a broad audience to participate in surveys, review aggregated data, and explore music through a range of emotional perspectives. Positioned within the broader landscape of psychological and music research, Emo-Tagger's insights have the potential to advance understanding in studies on the psychological effects of music. This tool fosters a community where users can engage with music on a deeper level, sharing emotional responses and musical experiences. By aggregating data on emotional reactions to various genres, artists, and songs, Emo-Tagger

offers valuable insights for music enthusiasts, researchers, and industry stakeholders, thereby enhancing understanding of the complex relationship between music and emotion.

2.1.1. Development Methodology

The Emo-Tagger project follows an agile software development approach to meet user needs effectively and deliver high-quality, scalable emotional annotation results. Agile methodology enables iterative and incremental development with frequent user feedback, allowing flexibility in the development process. The project is developed in structured, repeating sprints, with each sprint focusing on specific functionalities such as user interface development, data collection modules, data analysis tools, and gamification features.[1]

1. Requirements Analysis

In the initial phase of development, a comprehensive analysis of the target users' needs was conducted. Given the platform's goal of exploring the relationship between music and emotions, specific interactions within the app, types of data collected, and processing methods were identified in detail. Based on this analysis, the core functionalities and technical requirements of the application were defined.

2. Prototyping and Design

Following the requirements analysis, the design phase began with an emphasis on creating accessible, user-friendly interfaces for the web. Initial designs were developed as low-fidelity prototypes to quickly gather user feedback, and these prototypes were iteratively refined based on responses and usability testing.

3. Modular Development

Emo-Tagger's architecture is modular, comprising independent components for data collection, data processing, gamification, data analysis, and reporting. This modular structure allows each component to be tested separately and updated as needed. In line with the agile methodology, each sprint focused on completing specific modules that were then integrated and tested.

4. Gamification and Crowdsourcing Integration

To enhance user engagement, the platform incorporates gamification elements (points, badges, levels). Using a crowdsourcing approach, the application aims to reach a broad, culturally diverse audience, encouraging participation from users of different backgrounds. For these features, various interaction and reward mechanisms were designed to sustain user motivation.

5. Testing and Feedback

After each development phase, extensive testing was conducted, and user feedback was collected. Functional, user acceptance, and performance tests were performed throughout the development to ensure platform quality. The application was continuously improved and debugged based on user feedback.

6. Deployment and Maintenance

Once development was completed, the Emo-Tagger platform was launched to a broad user base. To ensure continuous user participation and rapid implementation of updates, maintenance and development work remains ongoing. New features are added, and existing features are refined based on user feedback and analysis.

This methodology has resulted in the successful development of Emo-Tagger as a dynamic, user-driven platform. The agile approach provided a flexible, user-centered development process that enhanced product quality by focusing on responsiveness to user needs and iterative improvement.

2.2 User Characteristics

2.2.1. Participants

- **2.2.1.1.** Participants must have an interest in music and emotional analysis.
- **2.2.1.2.** Participants must have basic familiarity with using web applications.
- **2.2.1.3.** Participants must have an active email account.

2.2.2. Admin

- **2.2.2.1.** Admin must have experience or interest in data management and analysis.
- **2.2.2.2.** Admin must know how to use a computer proficiently.
- **2.2.2.3.** Admin must understand basic music categorization and emotional labeling principles.

3. REQUIREMENTS SPECIFICATION

3.1 External Interface Requirements
3.1.1 User Interfaces

The user interface will be accessible via the internet and website.

3.1.2 Hardware Interfaces

The hardware components required for the successful execution of the project and for the website to provide uninterrupted and fast service are as follows:

Server Hardware
Web Servers:

- CPU: High performance processors are required to process user requests quickly.
- RAM: A large amount of RAM is important for database queries and application performance.
- Storage: SSD storage (preferably NVM SSD) for fast access will be used for storing the song database and user tags.
- Network Connectivity: High bandwidth and low latency network connections ensure users have a fast and uninterrupted experience.

Database Servers:

- CPU: Good performance processors are required to process database queries quickly.
- RAM: A sufficient amount of RAM ensures that database operations are performed quickly.
- Storage: High speed and large capacity SSD storage units are used to store all songs and tags.
- Network Connectivity: High bandwidth network connections are required for fast data transfer between database servers.
- Backup and Restore Systems
- Backup Servers
- Storage: Large capacity and reliable storage (preferably RAID configurations) is used for backup.
- Network Connectivity: High-speed network connections enable fast backup operations.
- Backup Software: Software that can perform automatic backup and restore operations should be used.

User Hardware

- Browser Compatibility: Users must use modern web browsers (Google Chrome, Mozilla Firefox, Safari, Microsoft Edge, etc.) to access the website.

- Internet Connection: A fixed or mobile internet connection is required to access the website. A low latency and high-speed internet connection improves the user experience.
- Audio Devices: Users need audio output devices such as speakers or headphones to listen to songs.

3.1.3 Software Interfaces

The application will require an internet browser for web access. No additional external software interfaces are required.

3.1.4 Communications Interfaces

The application will require internet connectivity for data synchronization, but no additional external communication interfaces are necessary.

3.2 Functional Requirements

3.2.1 User Registration and Profile Management Use Case

Use Case 1: User Registration and Profile Management

Use Case Name: User Registration and Profile Management

Use Case Number: UC01

Authors: Begüm

Actors: Participant, Admin

Overview: Allows users (both participants and admin) to register, log in, manage their profile, and log out of the platform.

Related Use Cases: UC02 (Emotional Tagging of Songs), UC07 (Access Privacy Settings)

Typical Flow Description:

Precondition: User must access the registration or login page.

1. User provides required information (e.g., username, email) on the registration page.

2. The system validates information and creates a user profile if details are correct.

3. User logs into the system using credentials.

4. Admin logs in using elevated credentials to access admin functionalities.

5. Both users can edit profile information, such as username and profile picture, from their profile management page.

6. User logs out to end the session.

Post-condition: User's profile is registered, managed, or updated as needed.

Alternative Flow Description:

Precondition: The system detects incomplete or invalid information during registration.

1. System prompts users to re-enter missing or correct invalid information.

2. User submits corrected information.

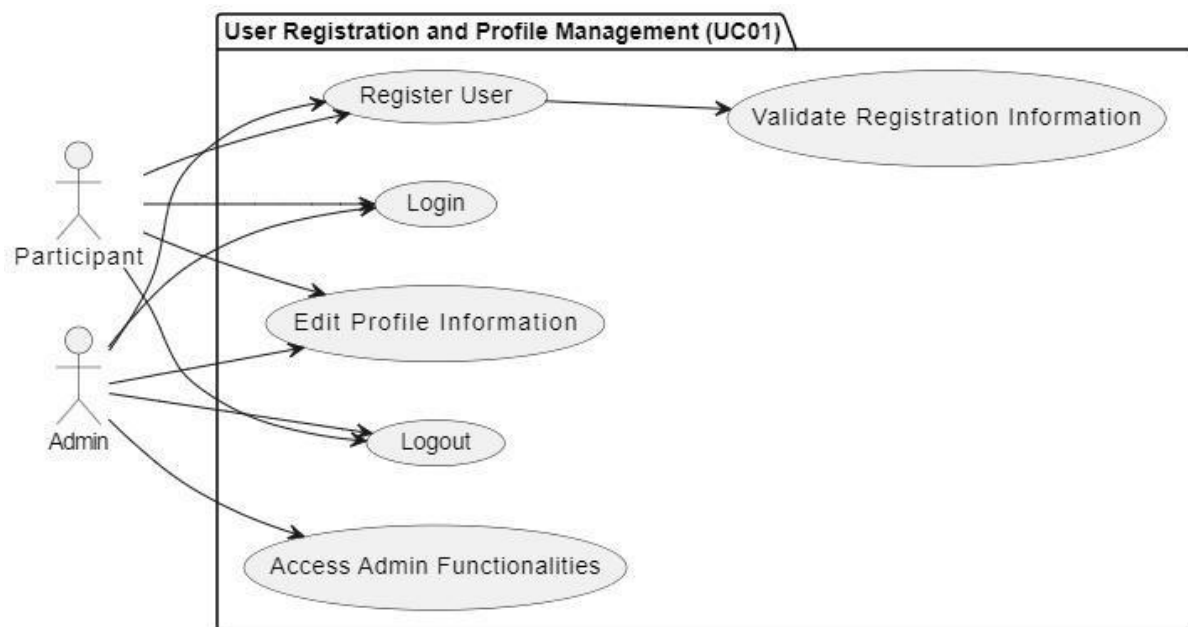**Post-condition**: User registration is completed successfully, and the system creates a new profile.

Figure 1 User Registration and Profile Management

3.2.2 Emotional Tagging of Songs Use Case

Use Case 2: Emotional Tagging of Songs
Actors Participant
Description: Enables participants to select songs and tag them with the emotions they experience, contributing to a crowdsourced emotional dataset.
Steps:
Participant selects a song from the available list.
Participant tags the song with one or more emotions (e.g., happiness, nostalgia).
System saves the emotional tags along with the song data.
The system updates the overall emotional profile of the song based on user contributions.

Use Case Name: Emotional Tagging of Songs

Use Case Number: UC02
Authors: Selin
Actors: Participant

Overview:

Allows participants to select a song from a list, tag it with emotions they experience, and contribute to a crowdsourced emotional dataset. The system saves these emotional tags, updates the song's emotional profile, and aggregates the data based on user contributions.

Related Use Case:

- UC01 (User Registration and Profile Management)
- UC05 (Gamification for User Engagement)

Typical Flow Description:

Precondition: The user must log in and access the emotional tagging system.

1. Participant selects a song:
   - The participant browses the available list of songs.
   - The system allows the user to select a song for tagging.
2. User tags the song with emotions:
   - The participant assigns one or more emotional tags to the selected song (e.g., "Happiness" or "Nostalgia").

3. System saves the emotional tags:
   - The system stores the emotional tags along with the song metadata in the database.
4. System updates the song's emotional profile:
   - Aggregates emotional tags submitted by all participants.
   - Recalculates the overall emotional profile dynamically based on user inputs.
5. Participant views updated profile:
   - The updated emotional profile of the song is displayed in real-time.

Post-condition: The song's emotional profile is updated and reflects the aggregated tags provided by all participants.

Alternative Flow Description:

- **Precondition**: The participant makes an error while tagging a song.
1. Participant decides to undo a tag:
   - The user selects the "Undo Tag" option to modify or remove a previously submitted tag.
2. System removes or updates the tag:
   - The system updates the emotional profile of the song accordingly.
- **Post-condition**: The incorrect or unwanted tag is successfully removed or modified, and the system recalculates the song's emotional profile.
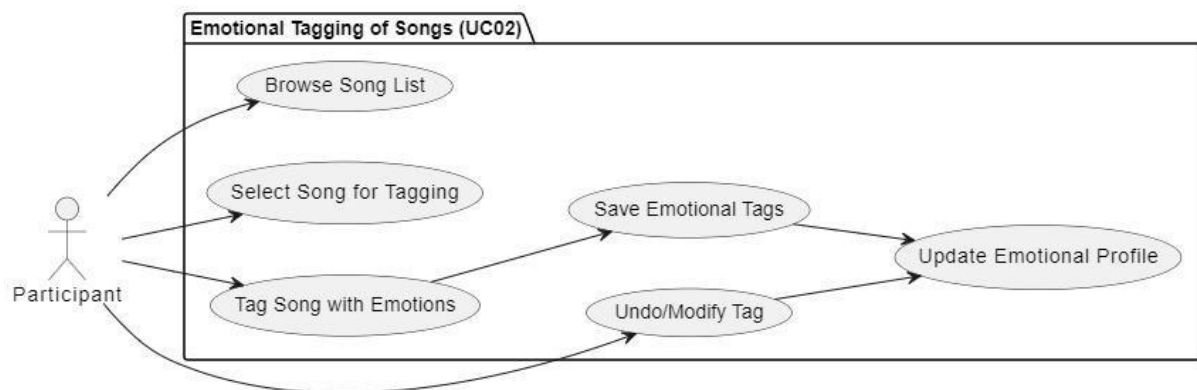

Figure 2 Emotional Tagging of Songs

3.2.3 View Emotional Trends Use Case

Use Case 3: View Emotional Trends

Use Case Name: View Emotional Trends
Use Case Number: UC03
Authors: Serhat
Actors: Participant
Overview: This use case describes how a participant can view aggregated emotional responses for a selected song, providing insights into how other users have emotionally responded to the same music.
Related Use Case: UC02 (Emotional Tagging of Songs)

Typical Flow Description:

Precondition: The participant must be logged in to the platform.
  1. The participant navigates to the song library or search bar and selects a song they are interested in.
  2. The system retrieves the aggregated emotional data for the selected song, showing the percentage of users who tagged the song with emotions like "happy," "calm," "energetic," etc.
  3. The system displays emotional trends in a visual format, such as a pie chart or bar graph, allowing the participant to view the emotional responses.
  4. The participant can optionally choose to filter the data by criteria (e.g., demographics or time range).
 Post-condition: The participant views the emotional trends, gaining insight into the general emotional impact of the song among other users.

Alternative Flow Description:

Precondition: The system cannot retrieve the aggregated emotional data due to insufficient data points.
  1. The participant selects a song to view emotional trends.
  2. The system checks for the availability of sufficient emotional tags for this song.
  3. The system notifies the participant that more emotional tags are needed to display trends.
  4. The participant is prompted to tag the song with their own emotional response or return to the previous screen.

 Post-condition: If the participant chooses to add a tag, their response contributes to building the dataset for future trend analysis.
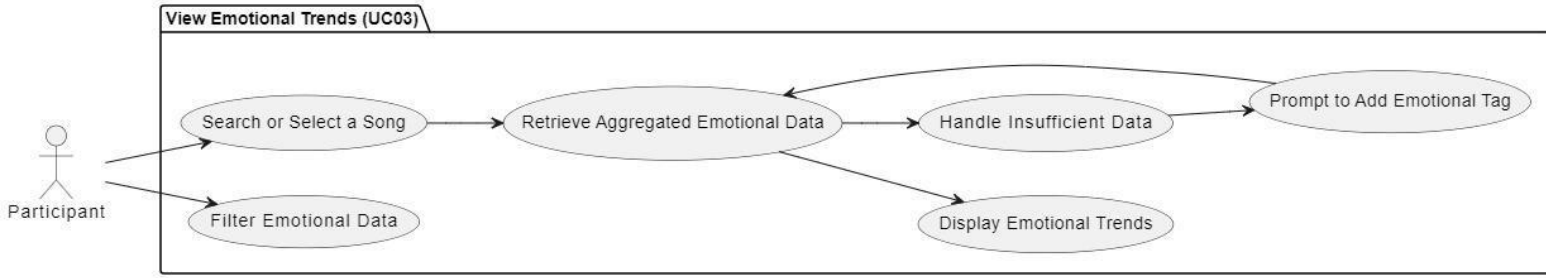
Figure 3 View Emotional Trends

3.2.4 Crowdsourced Data Collection Use Case
Use case 4: Crowdsourced Data Collection

Use Case Name: Crowdsourcing Emotional Tags for Songs
Use Case Number: UC04
Authors: Emre
Actors: Participant, Admin

Overview:
This use case describes how participants contribute emotional tags for songs, building a comprehensive dataset through crowdsourcing. Participants tag songs based on their emotional responses, and the system records this data. Admins monitor data volume and diversity to ensure the dataset's quality and comprehensiveness.[3]

Related Use Cases:
UC02 (Emotional Tagging of Songs)
UC08 (Song and Emotion Data Analysis)

Typical Flow Description:
Precondition: The participant must be logged in to access the emotional tagging system.

1. The participant listens to and selects a song from the available list.
2. The system provides tagging options based on predefined emotions (e.g., "Happiness," "Sadness").
3. The participant assigns one or more emotional tags to the selected song.
4. The system records the emotional tags and updates the aggregated emotional dataset.
5. Admin reviews data volume and diversity to ensure the dataset is comprehensive.

Post-condition: The emotional dataset is updated with new participant contributions, enhancing its quality and coverage.

Alternative Flow Description:

Precondition: The participant faces issues while tagging a song.

1. The participant selects a song and attempts to tag it.
2. The system detects incomplete or invalid tagging input.
3. The system prompts the participant to correct or complete the tagging.
4. The participant resubmits valid emotional tags, and the system updates the dataset.

Post-condition: The dataset is updated with corrected tagging input, maintaining its integrity.
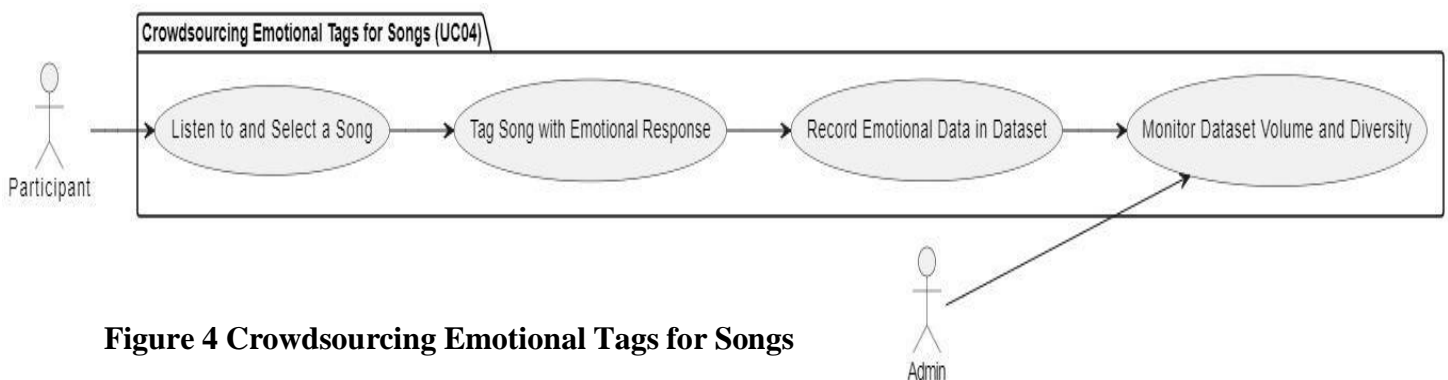

Figure 4 Crowdsourcing Emotional Tags for Songs

3.2.5 Gamification for User Engagement Use Case
Use Case 5: Gamification for User Engagement

Use Case Name: Gamification for User Engagement

Use Case Number: UC05
Authors: Selin
Actors: Participant

Overview:

This use case encourages user participation through gamification elements, including points, badges, levels, challenges, and leaderboards. Participants earn rewards by completing. activities and track their achievements through a leaderboard and badges system.

Related Use Cases:

- UC02 (Emotional Tagging of Songs)
- UC08 (S o n g and Emotion Data Analysis)

Typical Flow Description:

Precondition: The user must log in to access the gamification system.

1. **User starts an activity**:
   - The participant completes an activity such as tagging a song or participating in a challenge.
2. **System awards points and tracks progress**:
   - Points are awarded for the completed activity.
   - The system tracks progress toward unlocking badges or leveling up.
3. **Unlock badges or levels**:
   - When certain milestones are achieved, the user unlocks badges or advances to a new level.
4. **Update leaderboard with points**:
   - The participants' points are added to the leaderboard.
5. **Display leaderboard**:
   - The leaderboard showcases top contributors and encourages competition.
6. **User views achievements and leaderboard standings**:
   - The participant can view their badges, level, and rank on the leaderboard.

Post-condition: The participant's points, badges, and leaderboard position are updated in real-time.

Alternative Flow Description:

Precondition: The participant fails to complete an activity or opts out.

1. **System does not award points**:
   - If the activity is not completed, no points are awarded.
2. **User views progress**:
   - The participants can still view their existing points and achievements.

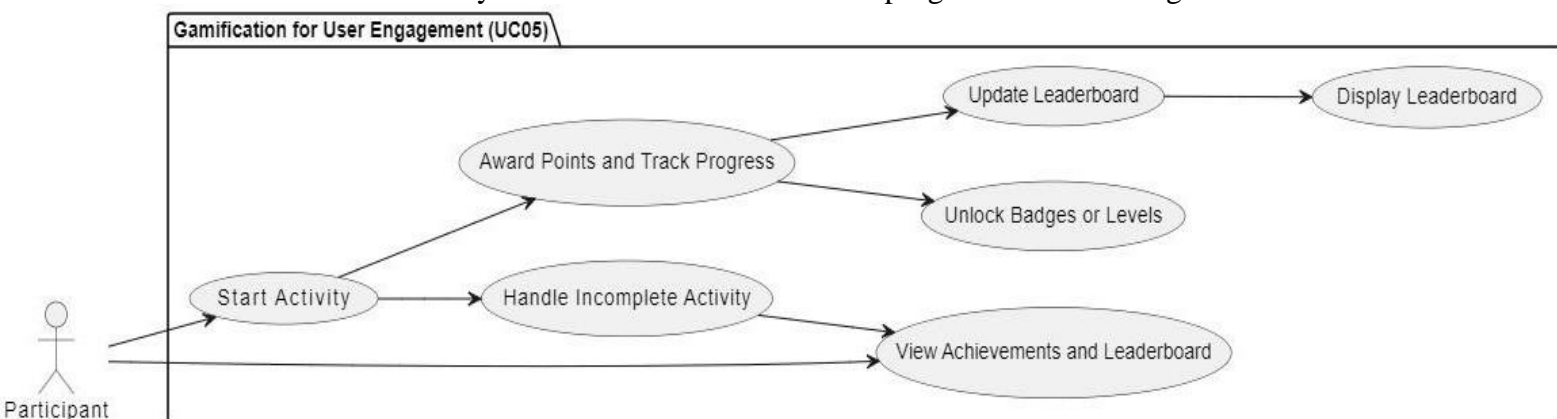**Post-condition**: The system retains the current user's progress without changes.

Gamification for User Engagement (UC05)

Figure 5 Gamification for User Engagement

3.2.6 Settings and Preferences Management Use Case
Use Case 6: Settings and Preferences Management

Use Case Name: Settings and Preferences Management
Use Case Number: UC06
Authors: Hamza
Actors: Participant, Admin

Overview:
This case allows users to adjust settings, such as notification preferences, theme selection, and volume controls. Adjustments are saved for future sessions.

Related Use Cases:

- UC01 (User Registration and Profile Management)
- UC06 (Settings and Preferences Management)

Typical Flow Description:
Precondition:

- The user must have access to the application.

Steps:

1. **User accesses the Settings menu:**
   - The participant opens the Settings menu to adjust preferences.
2. **User adjusts settings (e.g., volume control, theme, notifications):**
   - The participant selects and modifies various settings options.
3. **System saves settings for future sessions:**
   - The application stores the updated preferences for the user.

Post-condition:

- The user's customized settings are stored and applied in subsequent sessions.

Alternative Flow Description:
Precondition:

- The user fails to save changes or exits the settings menu.

Steps:

1. **The system does not apply unsaved settings:**

        ○ Changes are discarded if not explicitly saved.
2. **User retains previous settings:**
        ○ The system maintains the last saved settings.

Post-condition:

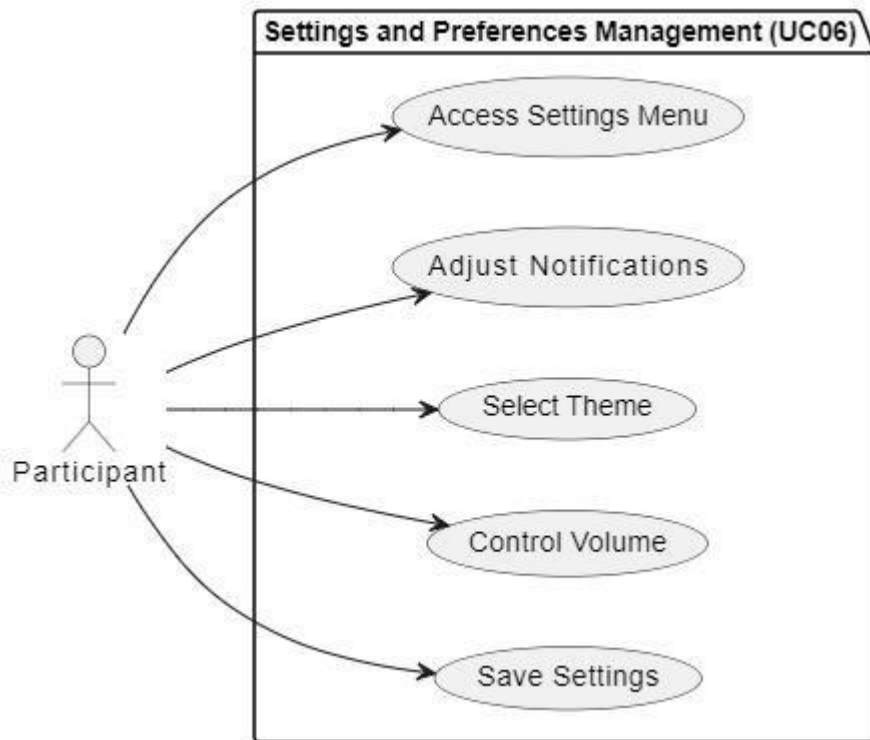- No changes are made to the settings, and the previous preferences remain in place.

Figure 6 Settings and Preferences Management

3.2.7 Access Privacy Settings Use Case
Use Case 7: Access Privacy Settings
Use Case Name: Access Privacy Settings
Use Case Number: U C 07
Authors: Tuğkan
Actors: Participant, Admin
Description: Allows users to control privacy settings, manage data sharing preferences, and access options related to data visibility.
Steps:
User navigates to Privacy Settings.
User enables or disables options related to data sharing. The
system updates the user's privacy preferences.

Typical Flow Description:

Precondition: User must be logged into the system.

1. User navigates to the "Privacy Settings" section from their profile page.
2. System retrieves and displays current privacy preferences.
3. User selects the desired options to enable or disable data sharing.
4. User modifies data visibility options (e.g., public, private, or restricted access).
5. The system validates the changes and updates the user's privacy preferences in the database.
6. Admins can access and modify privacy settings for all users as needed.

Postcondition: The user's updated privacy preferences are saved, and future interactions adhere to the new settings.
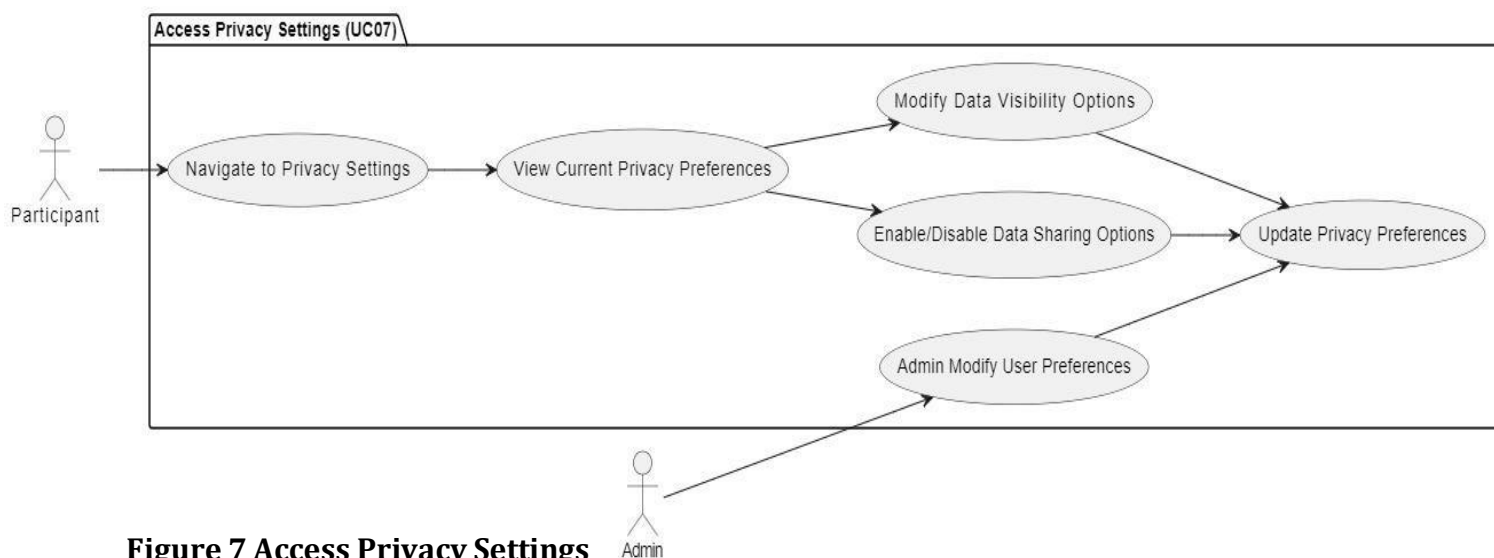

Figure 7 Access Privacy Settings

3.2.8 Song and Emotion Data Analysis Use Case
Use Case 8: Song and Emotion Data Analysis

Use Case Name: Song and Emotion Data Analysis
Use Case Number: UC08
Authors: Tuğkan
Actors: Admin

Overview:
This case describes how an admin can analyze aggregated songs and emotion data to gain insights into user behavior and emotional trends. The admin can access the data analysis.

dashboard, view emotional data for various songs, and generate reports or visualizations for decision-making and trend analysis.

Related Use Cases:
UC02 (Emotional Tagging of Songs)
UC03 (View Emotional Trends)

Typical Flow Description:
Precondition: The admin must log in to access the analysis dashboard.

1. Admin navigates to the "Data Analysis Dashboard" from the admin panel.
2. The system retrieves and displays aggregated emotional data for different songs.
3. Admin filters the data by specific criteria (e.g., date range, song genre, or demographics).
4. Admin generates detailed reports or visualizations based on the selected filters.
5. The system displays insights using visual formats such as charts or graphs, helping the admin understand trends.

Post-condition: The admin gains insights into emotional trends and user behavior for strategic decision-making.

Alternative Flow Description:
Precondition: The system encounters issues while retrieving the aggregated data.

1. Admin attempts to access the data analysis dashboard.
2. The system notifies the admin of the issue (e.g., "Data unavailable" or "Error fetching data").
3. Admin retries accessing the data or contacts support for assistance.

Post-condition: The admin resolves the issue and accesses the dashboard successfully at a later time.
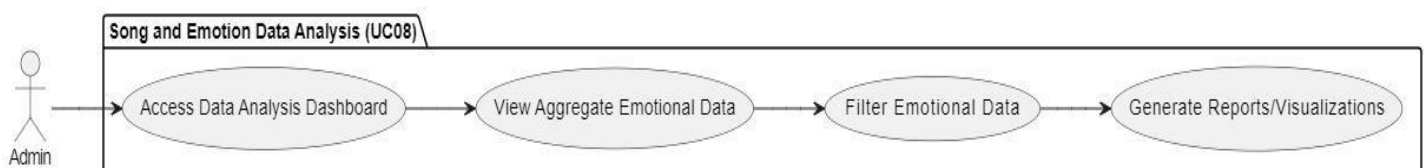

Figure 8 Song and Emotion Data Analysis

3.3 Performance Requirements

The project needs to be optimized for a large music database and high traffic volume. The website should ensure a fast and seamless user experience, especially when users are searching for songs, filtering, and interacting with the system. Given the 10,000+ songs and millions of user interactions, database queries and user actions must be processed with low latency. Additionally, since users will label songs based on the emotions they experience. While listening, the tagging system should function quickly and accurately.

3.4 Software System Attributes

3.4.1 Portability

- Emo-Tagger is designed as a web application.
- The application can be accessed via standard internet browsers, making it compatible with any device that supports web browsing. No additional hardware integration is required.

3.4.2 Performance

- Data retrieval and processing in the data layer should be optimized to ensure efficient access to and storage of large datasets.
- Queries and data transactions should be streamlined to minimize response time and server load.
- The system should handle real-time data requests efficiently, providing users with timely feedback without noticeable delays.
- Any data-intensive tasks, such as updating emotional response records, should be designed to complete within a few seconds to maintain smoother experience.

3.4.3 Usability

- The data layer should support clear and structured data retrieval, enabling seamless interaction between the user interface and the backend.
- Data error messages should be informative, providing guidance on how to resolve issues related to data retrieval or input errors.
- The system should allow easy integration of data from external sources if required for future enhancements.

3.4.4 Adaptability

- The data layer should be adaptable to potential changes in data structure, such as adding new data fields or modifying data relationships, without requiring significant code changes.
- The system should be flexible to accommodate future scalability and evolving data requirements as user engagement grows.

3.4.5 Scalability

- The data layer should support horizontal scaling to manage increasing data volume and concurrent users without performance degradation.
- The database should be optimized to handle high-volume read and write operations, ensuring consistent performance as the user base and dataset expand.

3.5 Safety Requirement

- Emo-Tagger should follow data security best practices to protect user information and emotional responses. This includes implementing encryption for data at rest and in transit, as well as role-based access control to prevent unauthorized data access.
- Data backup and recovery mechanisms should be in place to protect against data loss or corruption.

4 References

[1] K. Beck et al., "Manifesto for Agile Software Development," Agile Alliance, 2001. [Online]. Available: https://agilemanifesto.org/

[2] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, "From game design elements to gamefulness: defining 'gamification'," in *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, 2011, pp. 9-15.

[3] J. Howe, "The rise of crowdsourcing," *Wired Magazine*, vol. 14, no. 6, pp. 1-4, 2006.

[4] R. Cowie et al., "Emotion recognition in human-computer interaction," *IEEE Signal Processing Magazine*, vol. 18, no. 1, pp. 32-80, Jan. 2001.

[5] "General Data Protection Regulation (GDPR)," European Union, 2016. [Online]. Available: https://gdpr-info.eu/