Software Requirements Specification

For

Feelio

NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

Software Requirements Specification

for

Feelio

Project Mentor

Md. Rafid Mostafiz
Lecturer
Institute of Information Technology
Noakhali Science and Technology University

Prepared by

Md. Mamun Hossain **ID : MUH2025019M**
Md Foysal Mahmud **ID : MUH2025026M**
Kazi Ashikur Rahman **ID : MUH2025008M**

Submission Date: 12 June 2023

Table of Contents

1. Introduction 1

1.1 Problem Statement 1

1.2 Purpose 2

1.3 Project objective 2

1.3.1 General Objective 2

1.3.2 Specific Objective 2

1.4 Project Scope 3

1.5 Glossary 3

1.6 Overview 4

2 Stakeholders and Characteristics 4

3 Design and Implementation Constrains 5

3.1 User Interface Technology 5

3.1.1 Programming Language 6

3.1.2 CSS Framework 6

3.1.3 Bootstrap 6

3.2 Server-Side Technology 7

3.2.1 PHP 7

3.2.2 Python 7

3.2.3 Django 7

3.2.4 Database Server 8

4 Requirement Specification 8

4.1 Functional Requirement 8

4.1.1 User login and register 8

4.1.2 Emotion detection from uploaded images 9

4.1.3 Emotion detection from uploaded text 9

4.1.4 Emotion detection from real-time video streaming 9

4.1.5 Classification of emotions 10

4.1.6 Music playlist generation 10

4.1.7 User feedback mechanism 10

4.1.8 Notification 11

4.1.9 Real-time emotion updates 11

4.1.10 Admin Control 12

4.1.11 Log out 12
4.2 Data Requirement 13
4.2.1 Password Protection 13
4.2.2 Image dataset for emotion detection 13
4.2.3 Text dataset for emotion detection 14
4.2.4 Video dataset for emotion detection 14
4.2.5 Music database 15
4.2.6 User feedback and preference data 15
4.3 Performance Requirement 16
4.3.1 Scalability 16
4.3.2 High Accuracy 16
4.3.3 Capacity Requirements 17
4.3.4 Safety Critical Requirement 17
4.3.5 Robustness or Fault-Tolerance Requirements 17
4.4 Maintainability and Supportability 17
4.4.1 Maintenance Requirements 17
4.4.2 Supportability Requirements 18
4.5 Security Requirements 18
4.5.1 Access Requirements 18
4.5.2 Integrity Requirements 18
4.5.3 Privacy Requirements 18
4.6 Usability and Human Integrity Requirements 18
4.6.1 Ease of Use Requirements 18
4.6.2 Accessibility Requirements 19
4.7 Look and Feel Requirements 19
4.7.1 Appearance Requirements 19
4.8 Style Requirements 19
4.9 Legal Requirements 19
5 Requirement Engineering Process 20
5.1 Requirement Elicitation Techniques 20
5.1.1 Hold Interviews 20
5.1.2 Perform Document Analysis 20
5.1.3 System Interface Analysis 21

5.1.4 Distribute Questionnaires 21
5.2 Requirement Validation 21
5.2.1 Review the Requirements 21
5.2.2 Test the Requirements 21
5.2.3 Simulate the requirements 22
6 Use Case Diagram 23
7 Use Case Description 24
8 Activity Diagram 44
9 Requirement Traceability Matrix 64
9.1 Functional Requirements 64
9.2 Use Case 65
10 Appendix 67
10.1 Prioritization of requirements 67
10.1.1 Three-level Scale 67
10.1.2 Prioritization of the requirements of "Feelio" 67
11. References 69
12 Database Design ..... Error! Bookmark not defined.
12.1 Schema of the database ..... **Error! Bookmark not defined.**
12.2 Schema Diagram ..... **Error! Bookmark not defined.**

List of Tables

Table 1: Access Control (Create account) 24
Table 2: Access Control (Login) 25
Table 3: Access control (Activation Account) 26
Table 4: Get video 27
Table 5: Upload Image 28
Table 6 : Get Image 29
Table 7 Upload Text 30
Table 8 : Get Text 31
Table 9 : Retrieve mood 32
Table 10 : Approved mood 33
Table 11: Retrieve Genre 34
Table 12: Playlist 35
Table 13: Control Music 36
Table 14: Manage Database 37
Table 15: Notification 38
Table 16: Feedback 39
Table 17: Change Password 40
Table 18: Remember Me 41
Table 19: Edit Profile 42
Table 20: Logout 43

List of Figures

Figure 1: Use Case Diagram 23
Figure 2 : Create Account 44
Figure 3: Login 45
Figure 4 : Activate Account 46
Figure 5: Get Video 47
Figure 6 : Image Upload 48
Figure 7 : Get Image 49
Figure 8: text Upload 50
Figure 9: Get Text 51
Figure 10: Retrieve Mood 52
Figure 11: Approved Mood 53
Figure 12: Retrieve Genre 54
Figure 13: Playlist 55
Figure 14: Control Music 56
Figure 15: Manage Database 57
Figure 16: Notification 58
Figure 17: Feedback 59
Figure 18: Change Password 60
Figure 19: Remember Me 61
Figure 20: Edit Profile 62
Figure 21: Logout 63
Figure 22: Schema Diagram ..... **Error! Bookmark not defined.**

1. Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, abbreviations, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete **“Feelio”.** By presenting the problem statement in detail, the purpose of this document is to collect, evaluate, and provide a deeper understanding of the whole "**Feelio**" application. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs. The detailed requirements of the **“Feelio”** are provided in this document.

1.1 Problem Statement

The significance of music on an individual's emotions has been widely acknowledged throughout history. From ancient times to the present, music has served as a source of relaxation and emotional relief after a day filled with toil and hard work. Numerous studies have provided evidence that music's rhythm itself acts as a powerful tranquilizer, capable of soothing and uplifting our emotional state.

However, many individuals struggle with the daunting task of selecting songs that align with their current emotions. When faced with long, unsorted lists of music, individuals often feel demotivated and overwhelmed, making it difficult for them to find the specific songs they desire to listen to. As a result, users often resort to randomly picking songs from their music folders or opting to play their entire music collection on shuffle. Unfortunately, this traditional approach frequently fails to deliver songs that truly match the user's current emotional state.

Consider an example where a person is feeling sad and seeks solace in heavy rock music to release their sadness. It becomes impractical for the individual to manually search through their extensive playlist for all the heavy rock songs. Consequently, they resort to random selection or playing all the songs available, diminishing the effectiveness of the music's ability to connect with and uplift their emotions.

Moreover, the traditional method of searching and selecting songs has become monotonous and tiresome. This approach has remained unchanged for several years, and users are increasingly becoming bored with the lack of innovation and personalization in their music listening experience.

Thus, there is a clear need for an emotion-based music recommendation system that can address these challenges and provide a solution. By leveraging advanced technologies and algorithms, this system can automatically detect the user's emotions and recommend songs that align with their current emotional state. By offering personalized music recommendations, users will be able to

enhance their music listening experience, establish a deeper emotional connection, and truly benefit from the therapeutic qualities of music.

1.2 Purpose

The purpose of the Software Requirements Specification (SRS) document for this project is to provide a comprehensive overview of the requirements and functionalities of "Feelio". The SRS serves as a guide for the development team, stakeholders, and any other interested parties involved in the project. It details the features, constraints, and interfaces necessary for the successful implementation and deployment of the system.

The SRS document aims to define the scope of the project, including the platforms on which the system will be compatible and the integration with the user's music library. It also outlines the core functionalities, such as emotion detection, personalized music recommendations, and an intuitive user interface. Additionally, the SRS may specify any external references or technologies to be used as the basis for the development of the "Feelio" System.

Overall, the purpose of the SRS document is to provide clear and detailed requirements for the development team, ensuring a shared understanding of the project's objectives and guiding the implementation process. It serves as a foundation for communication, decision-making, and validation throughout the development lifecycle, enabling the successful realization of the system.

1.3 Project objective

1.3.1 General Objective

The main objective of this project is to develop the "Feelio" for all kinds of music lovers which aimed to serve as a platform to assist individuals to play and listen to the songs according to his emotions. It is aimed to provide a better enjoyment of entertainment to the music lovers.

1.3.2 Specific Objective

The Specific Objective for this project is specified as below:

i. To propose a facial expression detection model to detect and analyze the emotion of an individual.
ii. To accurately detect the four basic emotions, namely normal, happy, sad and excitement.
iii. To integrate the music player into the proposed model to play the music based on the emotions detected.

1.4 Project Scope

Currently there is no commonly used application or system which able to detect the emotion of individual and play music according to the emotion detected. This system will propose a new lifestyle to all music lovers which will ease them when searching for playlists. The target users will be the music lovers. English will be the main medium of language used in the proposed model and specifically aimed to detect some basic emotion such as normal, happy, sad or surprise. The evaluation of this system will base on the accuracy in detecting the correct facial expression as well as playing the right category of songs.

The scope of study will be as follow:

i. Study on the different method in expression detection. With the improvement of technology in image processing, more and more experts did researches or introduced different technique in processing a specific area or small area on an image. All these techniques can be applied to the facial expression processing. Researches had to be done in order to understand each technique which will then be useful in the project development.

ii. ii. Get information on the tools appropriate for the facial expression detection in order to build the proposed model for this project. Different tools (software and hardware) are studied on their feasibility and functionality as well as user-friendliness in order to figure out the most suitable and applicable tools to develop it.

1.5 Glossary

This section provides definitions for all document names, acronyms, and abbreviations. The application domain's terms and concepts are defined.

GUI	Graphical User Interface
API	Application Programming Interface
SRS	Software Requirement Specification
UI	User Interface
SDLC	Software Development Life Cycle
MB	Megabytes
XML	Extensible Markup Language
RESTful	Representational State Transfer
HTML	Hyper Text Markup Language

1.6 Overview

The "Feelio" is an innovative software application designed to enhance the music listening experience by accurately detecting and responding to the user's emotions. Leveraging advanced technologies such as computer vision and audio processing, the system analyzes the user's facial expressions, voice, and other input sources to classify their emotional state in real-time. By understanding the user's emotions, the system intelligently recommends songs from their personal music library or online streaming services, creating a personalized and emotionally resonant music playlist. The system seamlessly integrates with the user's music library, allowing easy access and management of their collection. With a user-friendly interface, the system empowers users to effortlessly interact with the recommendations, providing feedback and customization options. The overall goal of the project is to deliver a powerful emotion-based music recommendation system that revolutionizes the music listening experience, connecting users with songs that align with their emotions, and creating a captivating and immersive musical journey.

2 Stakeholders and Characteristics

1. Users

The primary stakeholders of the system are the users who will be using the "Feelio". They will interact with the system to receive personalized music recommendations based on their emotions. Users may have different musical preferences, emotional states, and levels of technical expertise.

2. Development Team

The development team consists of software engineers, data scientists, UX/UI designers, and other professionals responsible for designing, developing, and testing the system.

3. Project Managers

Project managers supervise the planning, execution, and delivery of the "Feelio". They are responsible for coordinating the development team, managing timelines, and ensuring the successful completion of the project.

4. Business Owners/Sponsors

Business owners or sponsors are stakeholders who have a interest in the project's success. They provide the necessary resources, funding, and support for the development and deployment of the system.

5. Music Industry Professionals

Music industry professionals such as artists, record labels, and music streaming platforms may have an interest in the system. They may provide input on music licensing, content partnerships, and the integration of their platforms or catalogs into the system.

6. Data Privacy and Security Experts

With the collection and analysis of user data, data privacy and security experts play a crucial role. They ensure that the "Feelio" obeys to privacy regulations, implements appropriate data protection measures, and safeguards user information.

7. User Experience Researchers

User experience researchers conduct studies, surveys, and interviews to gather feedback and insights from users. Their input helps shape the system's design, features, and usability, ensuring that it meets the needs and preferences of the target audience.

8. Quality Assurance/Testers:

Quality assurance professionals and testers are responsible for thoroughly testing the system to identify and resolve any bugs, usability issues, or performance problems before its release. Their expertise ensures the system meets high standards of quality and reliability.

9. Technical Support:

Technical support teams assist users by addressing their queries, providing troubleshooting assistance, and resolving any technical issues they may encounter while using the system. They play a crucial role in maintaining user satisfaction and system usability.

3 Design and Implementation Constrains

We have employed design and implementation constraints to ensure the success of this project. It also refers to a tool that allows developers and testers to inspect and interact with the application's user interface (UI) elements.

3.1 User Interface Technology

The visual layout of the components that a user could interact with in a website or technical product is referred to as user interface design, or UI design. In other terms, it is a website's visual design.

3.1.1 Programming Language

JavaScript is an ECMAScript-compliant high-level, frequently just-in-time compiled language. It has first-class functions, dynamic typing, and prototype-based object orientation. Its multi-paradigm, allowing you to program in event-driven, functional, or imperative styles.

JavaScript XML is abbreviated as JSX. It's just a JavaScript syntactic extension. It allows us to create HTML directly in React (within JavaScript code). It is straightforward to generate a template in React using JSX, but it is not a simple template language; instead, it has all of JavaScript's capability.

It is faster than standard JavaScript because it optimizes when converting to standard JavaScript. Rather than dividing the markup and functionality in different files.

3.1.2 CSS Framework

Cascading Style Sheets (CSS) is a language for specifying the appearance of a document written in a markup language like HTML. Along with HTML and JavaScript, CSS is a key component of the World Wide Web. Semantic UI is a website using UI component framework. Developers may use Semantic UI to create websites with quick and clear HTML, as well as a fully mobile responsive experience. Semantic UI offers a React-integrated version called Semantic UI React, which includes the following functionalities:

- jQuery Free.
- Declarative API.
- Augmentation.
- Shorthand Props.
- Sub Components.
- Auto Controlled State

3.1.3 Bootstrap

Bootstrap is a sizable repository of reusable code that comes in handy for developers. It is a JavaScript, CSS, and HTML frontend development framework. Using Bootstrap, web developers and designers can easily create fully responsive websites. It might be regarded as the most well-liked CSS framework for creating mobile-first and responsive applications.

3.2 Server-Side Technology

Server-side development refers to the actions that take place behind the scenes when an application is used. It primarily focuses on databases, scripting, website architecture, backend logic, APIs, and Servers.

3.2.1 PHP

The general-purpose programming language (GPPL) is PHP. It is mostly used as a server-side scripting language for the creation of websites. Web development is also simplified by the PHP frameworks. This framework makes it easier to reuse existing code and eliminates the need to create lengthy, intricate code for web apps. The majority of PHP frameworks are free source and simple to use. Because PHP is open-source and cost-free, developers may install it easily and utilize it right away. All major operating systems, including Windows, Unix, Linux, etc., support PHP. Web applications created using PHP may simply operate on any platform. PHP makes a safe connection with databases and connects to them with ease. It features an integrated module that may be used to quickly connect to the database. The primary purposes of the PHP framework are to simplify the construction of web applications and to automatically maintain the code. The built-in tools and features of PHP frameworks make it simpler to defend online applications from outside assaults and security risks.

3.2.2 Python

To build this system we want to use python programming language. First, collect a dataset of labeled images, text, and videos representing different emotions. Next, preprocess the data by resizing images, converting text to numerical representations, and extracting video features. Apply the trained model to detect emotions in uploaded content, obtaining probabilities or class labels. Create a music database associating emotions with songs or playlists. Utilize the detected emotion to recommend suitable music tracks or playlists to the user. Develop a user interface using libraries like Django, allowing users to upload content and receive music recommendations based on their detected emotions. Implement a recommendation algorithm based on user preferences and the detected emotions.

3.2.3 Django

In our project, we can leverage the Django framework to develop a user-friendly interface for our system. Django is a powerful Python web framework that provides a robust set of tools and features for building web applications. Using Django, we can create a web application that allows users to interact with our system seamlessly. We can design a user interface where users can upload images, text, or stream videos

to detect their emotions. Django's built-in form handling and file handling capabilities make it easy to handle user inputs and process them within our system. With Django's templating engine, we can dynamically generate HTML pages and present the music recommendations to the users based on their detected emotions. We can design intuitive and visually appealing web pages using HTML, CSS, and JavaScript, and seamlessly integrate them into our Django project.

3.2.4 Database Server

A piece of hardware that runs database software is called a database server. Recommended music , information logs, and other types of digital data with the use of database software. Large volumes of digital information may be easily stored, arranged, and maintained using database servers. Database servers perform by combining a database management system with memory and storage capacity for databases (DBMS).

4 Requirement Specification

All the requirements based on the elicitation process are described in this section.

4.1Functional Requirement

Functional requirements are those requirements that are used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. It consists of what task the system should perform, the processes involved, which data the system should hold and the interfaces with the user.

4.1.1 User login and register

FR-1	User Registration and Login to a registered account.		
Description	User should register his/her account for the first time and be able to log in to the account which was registered once. Already registered users will not face this stage.		
Stakeholders	System User	**Priority**	High

4.1.2 Emotion detection from uploaded images

FR-2	Emotion detection from uploaded images		
Description	The system shall analyze uploaded images and extract emotional features to determine the user's emotion.		
Stakeholders	System User	**Priority**	High

4.1.3 Emotion detection from uploaded text

FR-3	Emotion detection from uploaded text		
Description	The system shall process uploaded text and employ natural language processing techniques to detect emotions.		
Stakeholders	System User	**Priority**	High

4.1.4 Emotion detection from real-time video streaming

FR-4	Emotion detection from real-time video streaming		
Description	The system shall utilize computer vision algorithms to analyze real-time video streams for emotions		
Stakeholders	System User	**Priority**	High

4.1.5 Classification of emotions

FR-5	Classification of emotions		
Description	The system shall categorize detected emotions into happy, sad, excitement, and neutral to form a basis for music recommendations.		
Stakeholders	System User	**Priority**	High

4.1.6 Music playlist generation

FR-6	Music playlist generation		
Description	Based on the detected emotions, the system shall generate personalized playlists containing songs matching the user's emotional state.		
Stakeholders	System User	**Priority**	High

4.1.7 User feedback mechanism

FR-8	User feedback mechanism		
Description	The system shall include a feature for users to provide feedback on recommended playlists, allowing them to rate the relevance and quality of recommendations.		
Stakeholders	System User	**Priority**	Low

4.1.8 Notification

FR-8	Notification mechanism		
Description	In our system user will be notified about any matter of system by sending notification to users.		
Stakeholders	System User	**Priority**	Low

4.1.9 Real-time emotion updates

FR-9	Real-time emotion updates		
Description	The system shall continuously monitor and update the user's emotional state based on dynamic changes, ensuring the playlist adapts to shifting emotions.		
Stakeholders	System User	**Priority**	Medium

4.1.10 Admin Control

FR-12	Admin Control		
Description	There will be admin panel, they can control everything after logging in admin panel. Action will be taken by admin against anyone who posts fake or bad comments.		
Stakeholders	Admin	**Priority**	Medium

4.1.11 Log out

FR-13	User log out from their account.		
Description	The user will be able to log out of his/her account at the end of his need. Users will need to log in again for later use.		
Stakeholders	System User	**Priority**	Medium

4.2 Data Requirement

The Data Requirements section of the SRS provides information on the data used by the software application/system.

4.2.1 Password Protection

DR-1	Users' password will be protected		
Description	Hashing will be used to keep users' passwords protected.		
Stakeholders	Developers	**Priority**	High

4.2.2 Image dataset for emotion detection

DR-2	Image dataset for emotion detection		
Description	Image dataset for emotion detection: The system shall require a labeled dataset of image samples that express different emotional states, including happiness, sadness, excitement, and neutrality.		
Stakeholders	Data Provider, System User	**Priority**	High

4.2.3 Text dataset for emotion detection

DR-3	Text dataset for emotion detection		
Description	Text dataset for emotion detection: The system shall require a labeled dataset of text samples, such as user descriptions, captions, and comments, that express different emotional states, including happiness, sadness, excitement, and neutrality.		
Stakeholders	Data Provider, System User	**Priority**	High

4.2.4 Video dataset for emotion detection

DR-4	Video dataset for emotion detection		
Description	Video dataset for emotion detection: The system shall require a labeled dataset of videos samples that express different emotional states, including happiness, sadness, excitement, and neutrality.		
Stakeholders	Data Provider, System User	**Priority**	High

4.2.5 Music database

DR-5	Music database		
Description	The system shall require a comprehensive and up-to-date music database that includes a wide range of songs, albums, and artists. The database will serve as the source for generating music recommendations based on the user's detected emotion. It should provide relevant metadata such as song titles, artists, genres, and audio features (e.g., tempo, mood) to enhance the recommendation process.		
Stakeholders	Music Database Provider, System User	**Priority**	High

4.2.6 User feedback and preference data

DR-6	User feedback and preference data		
Description	The system shall collect and store user feedback and preference data, including ratings, reviews, and user-specified preferences (e.g., favorite genres, artists). This data will be used to improve the accuracy and relevance of the music recommendations over time, providing a personalized experience for each user.		
Stakeholders	Music Database Provider, System User	**Priority**	Medium

4.3 Performance Requirement

It is important to maintain the performance of the software system. To ensure performance we maintain these steps:

4.3.1 Scalability

PR-1	**Scalability**		
Description	The system should handle a large number of users and process data efficiently.		
Stakeholders	System Administrator, User	**Priority**	High

4.3.2 High Accuracy

PR-2	High Accuracy		
Description	The system should accurately detect emotions from various sources for precise recommendations.		
Stakeholders	User, Data Science Team	**Priority**	High

4.3.3 Capacity Requirements

PR-2	Concurrent user capacity		
Description	The system shall be capable of supporting a large number of concurrent users accessing the application simultaneously. It should have the capacity to handle the peak load of user requests, including emotion detection, playlist generation, and music streaming.		
Stakeholders	System Administrator, System User	**Priority**	High

4.3.4 Safety Critical Requirement

There are no safety-critical requirements for our project.

4.3.5 Robustness or Fault-Tolerance Requirements

There are no Robustness or Fault-Tolerance Requirements for our project.

4.4 Maintainability and Supportability

4.4.1 Maintenance Requirements

MR-1	Make the code maintainable.		
Description	Code must be developed so that it can be modified later and will be readable.		
Stakeholders	Developers	**Priority**	High

4.4.2 Supportability Requirements

This system meets Testability, Maintainability, Compatibility, Configurability, Serviceability, and installation ability which are related to supportability requirements.

4.5 Security Requirements

Securing information is much more important for a system to get users' dependability. Here are some of them:

4.5.1 Access Requirements

For accessing information, the system will use some authorization techniques to ensure that the correct data is used by the correct user.

4.5.2 Integrity Requirements

Integrity requirements refer to a security system that ensures an expectation of data quality. It also ensures that all data of the system would never be exposed to malicious modification or accidental destruction. For preventing anonymous access to user passwords, the system will use an encryption technique called the Hash Function for encrypting user passwords.

4.5.3 Privacy Requirements

Privacy requirements are enhanced to protect stakeholders' privacy. In this way, all data or a partial part of data is going to be disclosed according to the system's privacy policy. To ensure privacy, the central database should be protected by the anonymous. Users are permitted to get access to those data which are being associated with them which can be ensured by the user login system.

4.6 Usability and Human Integrity Requirements

This system will provide a more user-friendly environment

4.6.1 Ease of Use Requirements

Our system will be easier to use by any type of people and they don't need any training to use the system.

4.6.2 Accessibility Requirements

To get access to the application, the application provides authorization/authentication. This application uses various modules.

SR-1	The system provides security strategies.
Description	The system is designed in a way that allows all modules to access a mechanism that provides security services.

4.7 Look and Feel Requirements

Look and feel requirements mainly refer to how the system will look.

4.7.1 Appearance Requirements

AR-1	Text color and font		
Description	Our system has to be different and attractive from other existing audio players using a better look and feel.		
Stakeholders	Developer, User	**Priority**	High

4.8 Style Requirements

There are no style requirements in our system.

4.9 Legal Requirements

Legal requirements normally refer to the terms and conditions or privacy policy of any organization. The terms and conditions of our application are that no third-party software or person is allowed to use our data for their business purpose.

5 Requirement Engineering Process

Requirements Engineering (RE) determines software requirements according to customer requirements or needs. Requirements engineering process includes requirements elicitation, needs modeling, requirements analysis, requirements assurance & validation, and requirements management.

5.1 Requirement Elicitation Techniques

Requirements elicitation is the practice of researching and finding system requirements for users, customers, and other stakeholders also referred to as "requirement gathering". Requirement elicitation can be done by contacting participants directly or by doing some research, analysis, and testing.

5.1.1 Hold Interviews

Conduct one-on-one or group interviews with relevant stakeholders, such as potential users, domain experts, system administrators, and data scientists. Use open-ended and targeted questions to gather their insights, expectations, and requirements regarding the emotion detection, playlist generation, and music recommendation aspects of the system.

- Purpose: To gain a deeper understanding of the stakeholders' perspectives, needs, and requirements for the emotion-based music recommendation system.

5.1.2 Perform Document Analysis

Review existing documents, reports, research papers, and relevant literature related to emotion detection, music recommendation systems, and user experience. Analyze academic papers, industry guidelines, and documentation from similar systems to extract valuable insights, best practices, and requirements that can be applied to your system. Identify patterns, common functionalities, and potential challenges documented in these sources.

- Purpose: To leverage existing knowledge and expertise in the field to inform the design and development of the emotion-based music recommendation system.

5.1.3 System Interface Analysis

Analyze existing music recommendation systems, user interfaces, and user experience (UX) design patterns. Explore popular music streaming platforms, music recommendation algorithms, and related applications to gain insights into successful interface designs, usability features, and interactive components. Identify strengths and weaknesses of existing systems and consider how to incorporate effective interface elements into your own system.

- Purpose: To understand the current state-of-the-art in user interfaces for music recommendation systems and identify design elements that can enhance the usability and user experience of your system.

5.1.4 Distribute Questionnaires

Develop a structured questionnaire consisting of both closed-ended and open-ended questions. Distribute the questionnaire to a wide range of potential users and other stakeholders, such as music enthusiasts, online music platform users, and individuals interested in emotional well-being. Analyze the responses to gather quantitative and qualitative data about their preferences, expectations, and requirements for the system.

- Purpose: To collect data from a larger sample of stakeholders, allowing for a broader understanding of user preferences and requirements.

5.2 Requirement Validation

Requirement validation ensures that the requirements are correct and reflect the quality you want from this program. In the beginning, our requirements looked good but when we read them and tried to work with them, they came out having ambiguities and gaps.

5.2.1 Review the Requirements

Negative peer review, especially the type of rigorous review called evaluation, is unique among the highest quality software processes available. We had a team of reviewers representing different perspectives and carefully examined written needs, analysis models, and related information on disability.

5.2.2 Test the Requirements

The test creates another view of the requirements. We also performed writing tests regarding assurance of whether the expected performance was found or not. Getting tested by the user needs to document the expected product behavior under specified conditions.

5.2.3 Simulate the requirements

To stimulate requirements, trading tools are available that we have used to simulate a proposed system in place or to add details of written requirements. The simulation takes prototyping to the next level.

6 Use Case Diagram

Figure 1: Use Case Diagram

7 Use Case Description

Table 1: Access Control (Create account)

Use Case	Access Control	
Goal	User wants to create an account in the "Feelio" System.	
Preconditions	N/A	
Success End Condition	A user account is created.	
Failed End Condition	User account is not created.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Create Account" or "Sign up" button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User opens the "Feelio" application.
	2	User clicks the "Create Account" or "Sign up" button.
	3	User provides required information such as name, email, password etc.
	4	Server checks Gmail is already existed or not.
	5	User activated the account.
	6	Account is created.
Alternative Flows	**Step**	**Branching Action**
	2a	User have an account.
	2b	User needs to be clicked login button.
	2c	User get access of the application.
	3a	User does not provide information.
	4a	Server shows that information invalid or used before.
	4a1	User needs to change provided information.
Quality Requirements	**Step**	**Requirement**
	1	Server will respond within 3 to 5 seconds .

Table 2: Access Control (Login)

Use Case	Access Control	
Goal	User login to "Feelio" system.	
Preconditions	User have an account.	
Success End Condition	Successfully login to "Feelio" system.	
Failed End Condition	Unable to login.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Login" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User opens the "Feelio" system.
	2	User provides Gmail and password.
	3	Server confirms the password with that Gmail.
	4	User successfully logs into application.
Alternative Flows	**Step**	**Branching Action**
	2a	User does not have an account.
	2a1	User clicks the "Create Account" button to create an account.
	4a	Provides the correct email and password.
Quality Requirements	**Step**	**Requirement**
	1	User can login whenever he/she want (Availability).

Table 3: Access control (Activation Account)

Use Case	Access Control	
Goal	User activate account.	
Preconditions	Provide Gmail.	
Success End Condition	Successfully create account.	
Failed End Condition	Unable to create account.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	User needs to click on the provided link which was sent by the system in specified Gmail account.	
Main Success Flows	**Step**	**Action**
	1	User provides Gmail and password and clicked signup
	2	System sends an activation link to the given Gmail.
	3	User click on that link to activate account.
	4	Account activated.
Alternative Flows	**Step**	**Branching Action**
	2a	System doesn't send activation link.
	2a1	User again provide information and signup.
	3a	User doesn't click on activation link
	4a	Account is not activated.
Quality Requirements	**Step**	**Requirement**
	2	Server must send an activation link to the given Gmail within 5 minutes.

Table 4: Get video

Use Case	Get video	
Goal	User will receive a playlist based on her/his emotion from real time video streaming.	
Preconditions	User must login into application.	
Success End Condition	User receive a playlist.	
Failed End Condition	User doesn't receive any music playlist.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Video Capture" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User go to video capture section.
	2	System sends a notification for accessing camera.
	3	User allows the camera access.
	4	Face detector detect the face and retrieve a mood based on expression.
	5	System accesses the database and show a playlist of music based of approved mood.
	6	User can listen the music based on their emotion.
Alternative Flows	**Step**	**Branching Action**
	3a	System sends a notification for accessing camera.
Quality Requirements	**Step**	**Requirement**
	1	Accuracy rate of emotion detection will be high.
	2	Robustness

Table 5: Upload Image

Use Case	Upload Image	
Goal	User will be able to upload the image.	
Preconditions	User must be logged in.	
Success End Condition	Image will be uploaded successfully.	
Failed End Condition	Unable to upload.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Image upload" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	Users go to image upload section.
	2	User browse the device section and select an image file.
	3	Upload the image by clicking "upload" button.
Alternative Flows	**Step**	**Branching Action**
	2a	Selects the image file again.
Quality Requirements	**Step**	**Requirement**
		Response time will be 2 seconds.

Table 6 : Get Image

Use Case	Get Image	
Goal	User will receive a playlist based on her/his emotion from the uploaded image.	
Preconditions	User must logged in.	
Success End Condition	User receive a playlist.	
Failed End Condition	User doesn't receive any music playlist.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Image upload" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User clicks on "Image upload" button.
	2	Selects an image file and clicks upload.
	3	System scan the image and retrieve a mood based on expression.
	4	System accesses the database and show a playlist of music based of approved mood.
	6	User can listen the music based on their emotion.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
	1	Accuracy rate will be high.

Table 7 Upload Text

Use Case	Upload Text	
Goal	User will be able to upload the typing text.	
Preconditions	User must type a text.	
Success End Condition	Text will be uploaded successfully.	
Failed End Condition	Unable to upload	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Text upload" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	Users go to text upload section.
	2	User can type the text based their own emotion.
	3	Upload the text by clicking "upload" button.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
		Uploaded time will be minimum.

Table 8 : Get Text

Use Case	Get Text	
Goal	User will receive a playlist based on her/his emotion from the uploaded image.	
Preconditions	User must logged in.	
Success End Condition	User will receive a playlist.	
Failed End Condition	User doesn't receive any playlist from the application.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Text upload" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User type the text based on their current mood.
	2	Upload the written text by clicking "upload" button.
	3	System detect the text and retrieve a mood based on expression.
	4	System accesses the database and show a playlist of music based of approved mood.
	5	User will receive a playlist.
Alternative Flows	**Step**	**Branching Action**
	2a	N/A
Quality Requirements	**Step**	**Requirement**
		Detection accuracy will be high rated.

Table 9 : Retrieve mood

Use Case	Retrieve mood	
Goal	System retrieve the mood.	
Preconditions	Start the video of upload an image or type a text.	
Success End Condition	Successfully retrieve the user mood.	
Failed End Condition	Unable to retrieve the mood.	
Primary Actors: **Secondary Actors:**	System User	
Trigger	"Get video" or "Get image" or "Get text" button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	Users start the real-time video.
	1b	Face detector detect the face.
	1c	System retrieve a mood based on their expression.
	2	User upload an image.
	2b	System scan the image and retrieve a mood.
	3	User upload a text.
	3b	Text detector detects the text and retrieve a mood.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
		Response time will be at most 3 seconds

Table 10 : Approved mood

Use Case	Approved mood	
Goal	User mood will be approved	
Preconditions	Camera must be on or upload any image or type any text on text upload section.	
Success End Condition	Successfully approve the user mood.	
Failed End Condition	Unable to approve.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Get video" or "Get image" or "Get text" button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	Users start the real-time video.
	1b	Face detector detect the face.
	1c	System retrieve a mood based on their expression and mood will be approved.
	2	User upload an image.
	2b	System scan the image and retrieve a mood.
	2c	The system approved the mood.
	3	User upload a text.
	3b	Text detector detects the text and retrieve a mood.
	3c	The system approved the mood.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
		Emotion detection accuracy will be high.

Table 11: Retrieve Genre

Use Case	Retrieve Genre	
Goal	System retrieve a genre of songs based on detected emotion.	
Preconditions	Needs to be accessed database.	
Success End Condition	Successfully retrieve a genre of songs.	
Failed End Condition	Failed the retrieve genre.	
Primary Actors: **Secondary Actors:**	System User	
Trigger	N/A	
Main Success Flows	**Step**	**Action**
	1	User start the video or upload an image or upload a text.
	2	System will approve a mood based on its training.
	3	System access the database and return a genre a songs based approval mood.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
	2	There are at least 100 songs of every genre.

Table 12: Playlist

Use Case	Playlist	
Goal	User receive a playlist.	
Preconditions	Camera must be on or upload any image or type any text on text upload section.	
Success End Condition	User receive a playlist successfully.	
Failed End Condition	Unable to receive a playlist.	
Primary Actors: **Secondary Actors:**	System User	
Trigger	N/A	
Main Success Flows	**Step**	**Action**
	1	User start the video or upload an image or upload a text.
	2	System will approve a mood based on its training.
	3	System access the database and return a genre a songs based approval mood.
	4	System provide a playlist based on retrieval genre.
Alternative Flows	**Step**	**Branching Action**
	3	N/A
Quality Requirements	**Step**	**Requirement**
		There have been 100 songs of every playlist.

Table 13: Control Music

Use Case	Control Music	
Goal	User can control the music	
Preconditions	N/A	
Success End Condition	Successfully control the music.	
Failed End Condition	Unable to control	
Primary Actors: **Secondary Actors:**	User System	
Trigger	N/A	
Main Success Flows	**Step**	**Action**
	1	If user clicks "pause" or "start" then music will be paused and start.
	2	User can stop the music by clicking the "stop" button.
	3	If user clicks the "previous" button previous music will be played.
	4	User can go to the next songs by clicking by "next" button.
	5	If user clicks "repeat" button then current music will be repeated.
Alternative Flows	**Step**	**Branching Action**
	2a	Music play manually.
	3a	Next songs will be played.
	5a	Next songs will be played
Quality Requirements	**Step**	**Requirement**
		N/A

Table 14: Manage Database

Use Case	Manage Database	
Goal	System will manage the database.	
Preconditions	N/A	
Success End Condition	Database will be managed successfully.	
Failed End Condition	Database management failed.	
Primary Actors: **Secondary Actors:**	System / Admin N/A	
Trigger	N/A	
Main Success Flows	**Step**	**Action**
	1	Go the add songs section.
	1a	Provide the song details
	1b	Select a genre of songs.
	1c	Add into specific genre.
	2	Admin can also remove the song.
	2a	Select the songs that will be removed.
	2b	Delete the song from database
	3	Admin can edit the songs details.
	3a	Provide the edited details.
	3b	Clicks the "Save" button.
	4	Admin will be able to create a new playlist.
	4a	Collect the specific numbers of songs
	4b	Provide every song details
	4c	Provide genre of music and click to add.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
	2	N/A

Table 15: Notification

Use Case	Notification	
Goal	User will be notify	
Preconditions	User must logged in.	
Success End Condition	User will be able to receive any notification.	
Failed End Condition	Unable to receive notification.	
Primary Actors: **Secondary Actors:**	System User	
Trigger	N/A	
Main Success Flows	**Step**	**Action**
	1	User to go the notification section.
	2	User able to see the all notifications come from the system such as update version notification etc.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
	2	All-important notice will give to user as notification.

Table 16: Feedback

Use Case	Feedback	
Goal	User will be able to provide his/her valuable opinion.	
Preconditions	User must logged in.	
Success End Condition	Successfully receive a feedback from users.	
Failed End Condition	Unable to provide feedback.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Feedback" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User go the feedback section.
	2	User write his/her opinion form of text.
	3	Upload the feedback.
	4	System will receive a feedback from user.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
	3	System must be responsible to the users feedback

Table 17: Change Password

Use Case	Access Control	
Goal	Change user password	
Preconditions	Go to the edit profile	
Success End Condition	Change user's current password	
Failed End Condition	Unable to change password	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Change password" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User click on "Change Password"
	2	System asks current password and new password
	3	User provide current and new password.
	4	User click the "Save" Button.
	5	System changes the password.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
		N/A

Table 18: Remember Me

Use Case	Remember Me	
Goal	System remembers email and password	
Preconditions	N/A	
Success End Condition	System remembers login information.	
Failed End Condition	Unable to remember login information.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Remember me" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User click on "Remember Me" Button.
	2	Server store the information.
	3	Whenever user try to login again system provides the stored information.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
		N/A

Table 19: Edit Profile

Use Case	Edit Profile	
Goal	Update information.	
Preconditions	N/A	
Success End Condition	Successfully update information.	
Failed End Condition	Unable to update information.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Edit Profile" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User click the "Edit Profile".
	2	User click the "Change username".
	2a	System asks new username. User provide new username.
	2b	User click the "Save" Button
	2c	System updates the username.
	3	User click on "Change Password"
	3a	System asks current password and new password
	3b	User provide current and new password.
	3c	User click the "Save" Button.
	3d	System changes the password.
Alternative Flows	**Step**	**Branching Action**
	2a1	Provide a valid username.
	3b1	Provide a valid password.
Quality Requirements	**Step**	**Requirement**
		N/A

Table 20: Logout

Use Case	Logout	
Goal	Logout from "Feelio" system.	
Preconditions	System updates the board whenever	
Success End Condition	Successfully logout from "Feelio" system.	
Failed End Condition	Remain in a login state.	
Primary Actors: **Secondary Actors:**	User System	
Trigger	"Logout" Button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	User using the "Feelio" system after login
	2	User clicks the "Logout" button.
	4	User logout successfully.
Alternative Flows	**Step**	**Branching Action**
		N/A
Quality Requirements	**Step**	**Requirement**
		N/A

8 Activity Diagram

Activity diagram (Create Account)

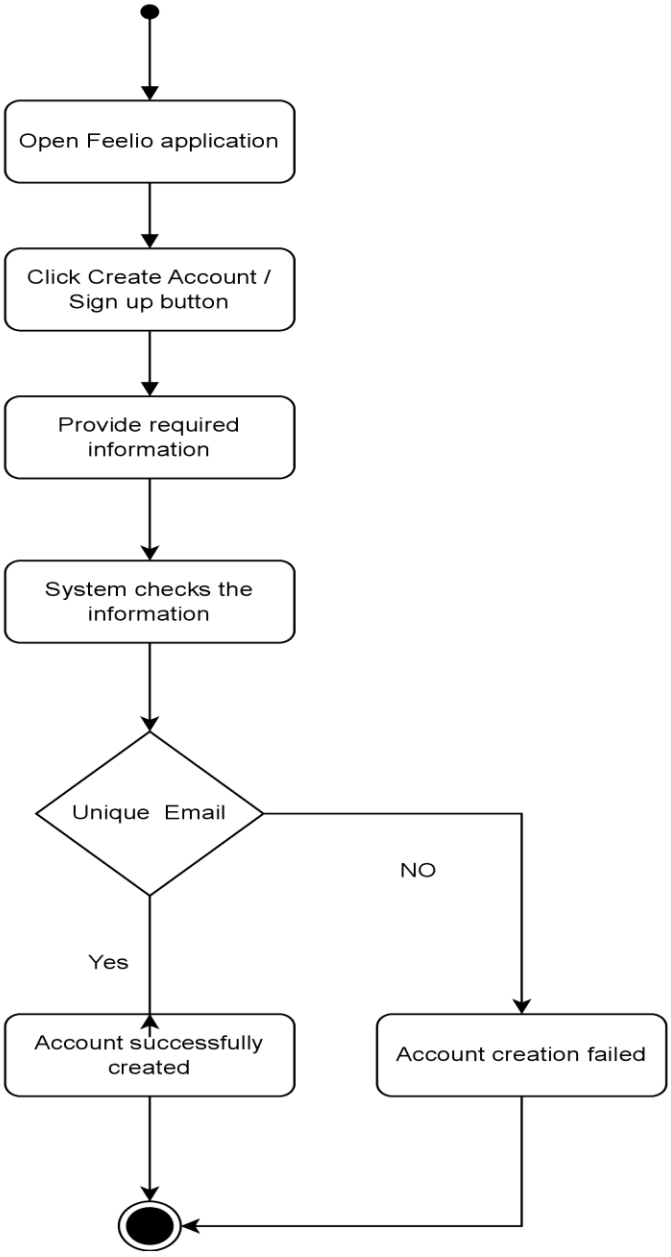

Figure 2 : Create Account

Activity diagram (Login)

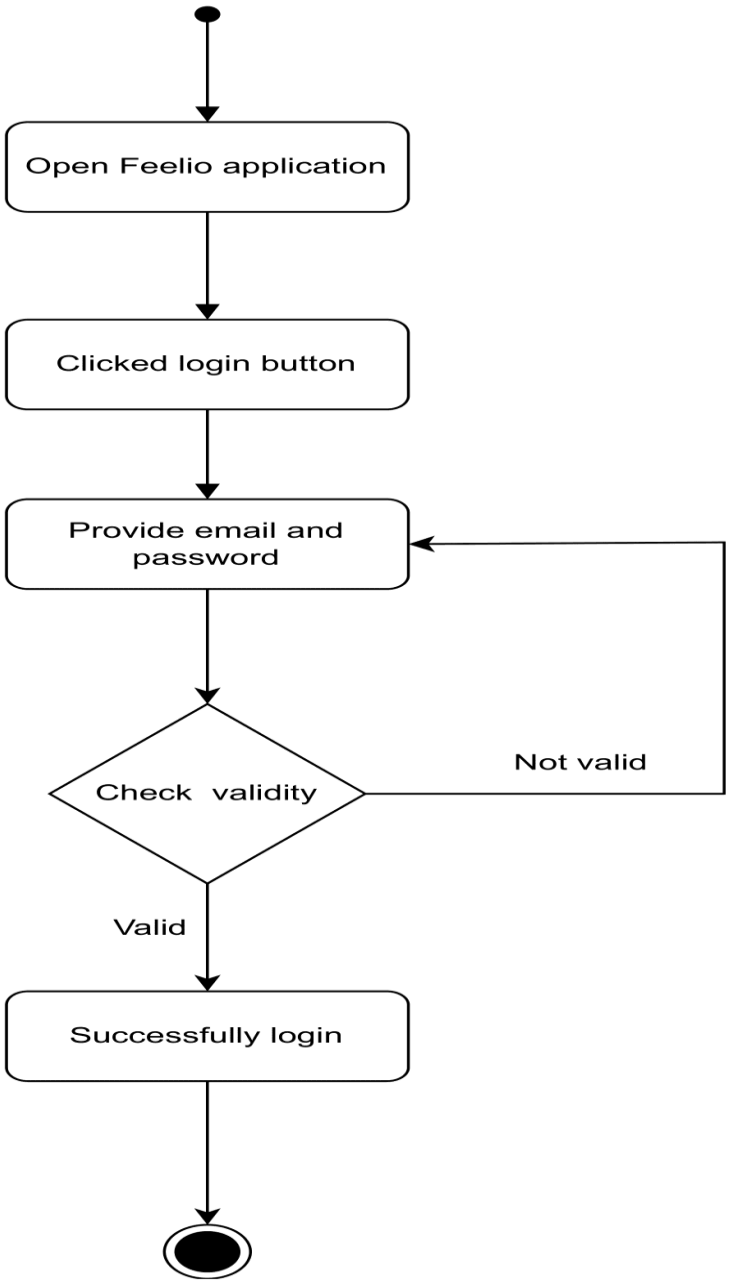

Figure 3: Login

Activity diagram (Activate Account)

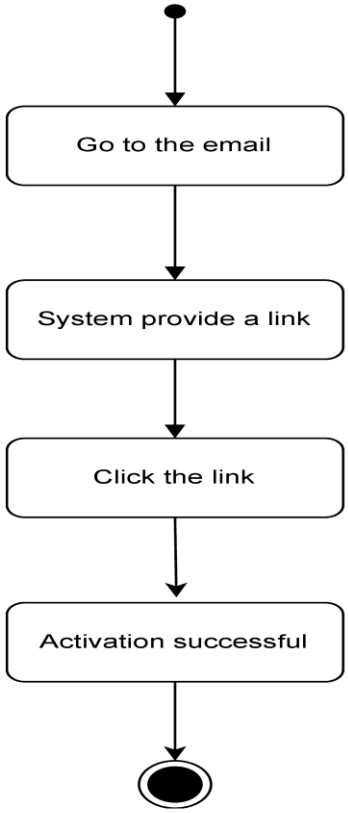

Figure 4 : Activate Account

Activity diagram (Get video)

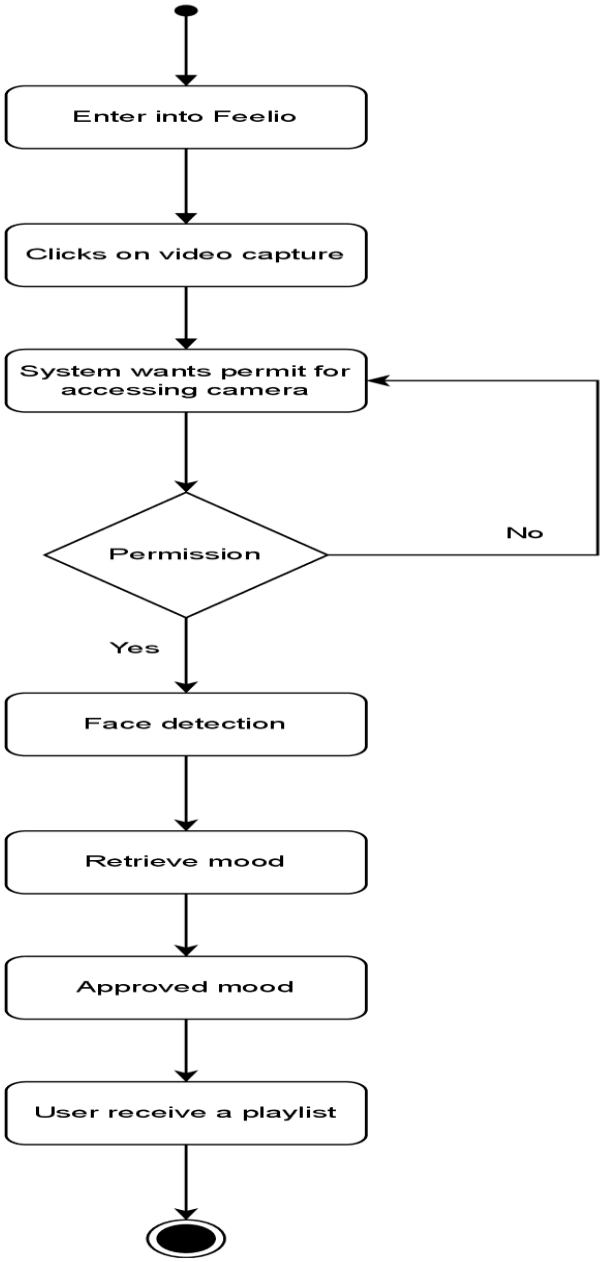

Figure 5: Get Video

Activity diagram (Image upload)

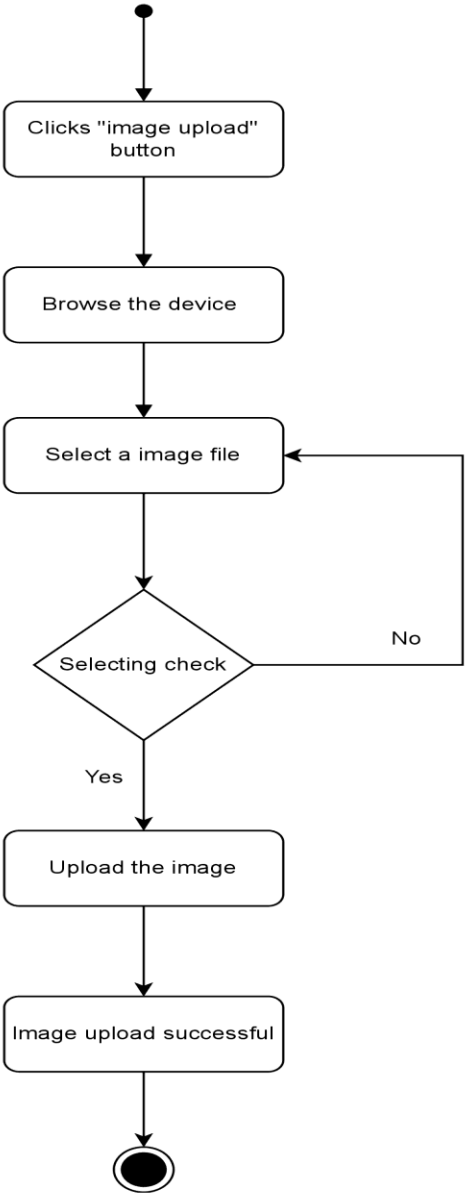

Figure 6 : Image Upload

Activity diagram (Get image)

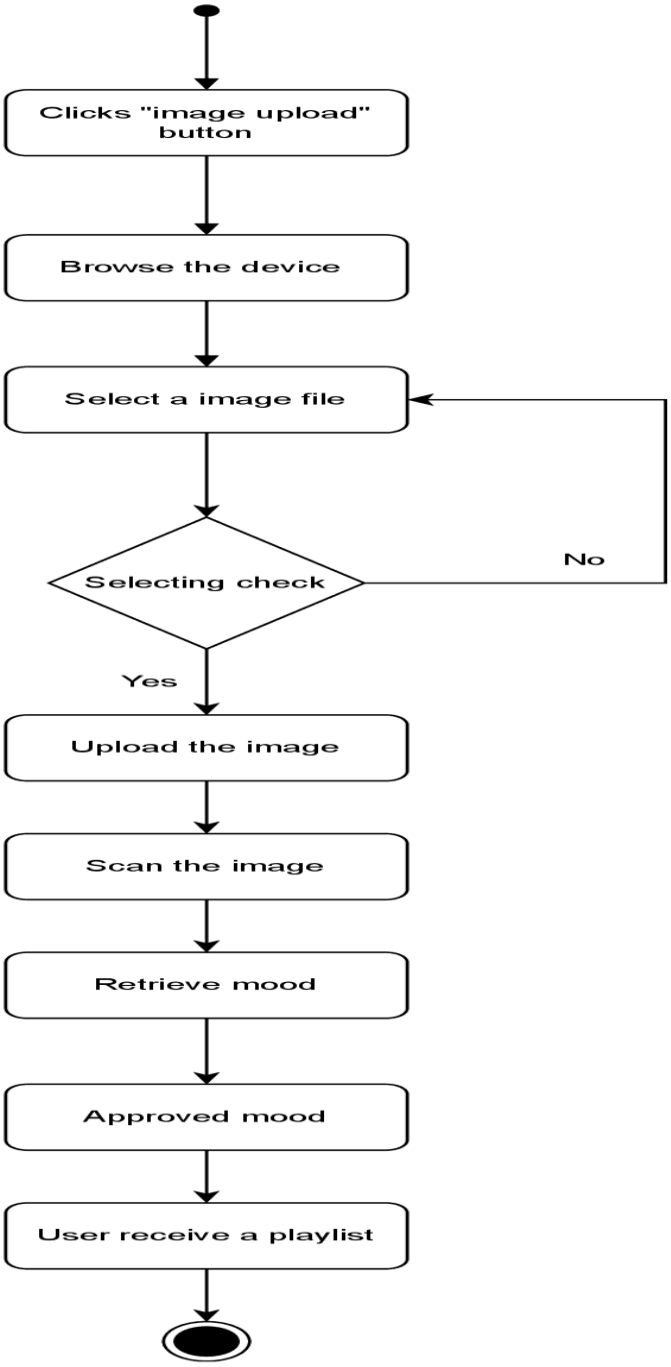

Figure 7 : Get Image

Activity diagram (Text upload)

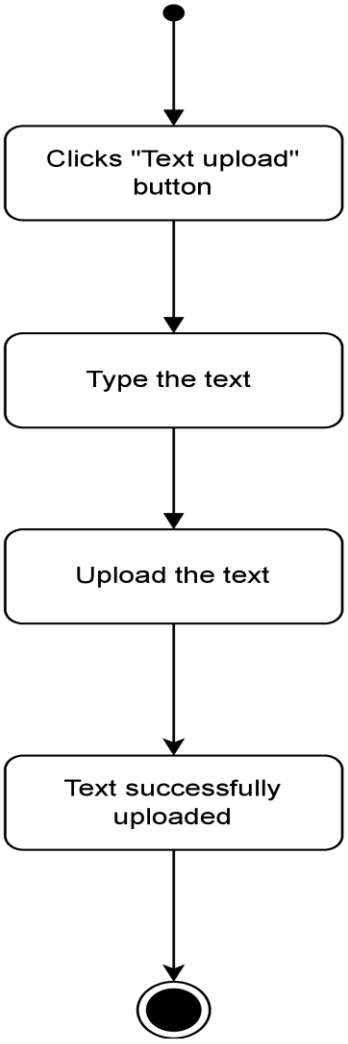

Figure 8: text Upload

Activity diagram (Get text)

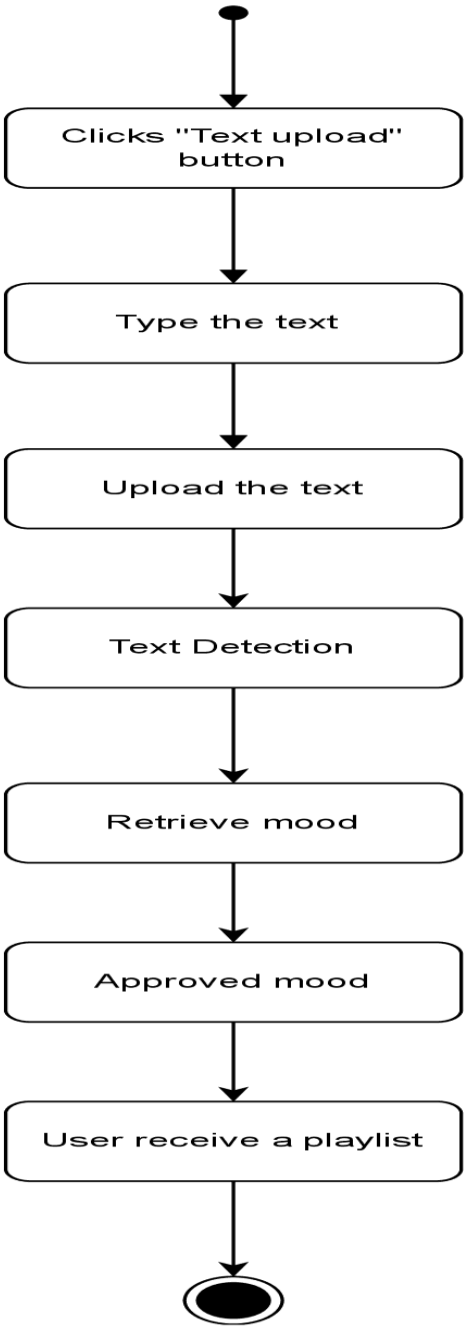

Figure 9: Get Text

Activity diagram (Retrieve mood)

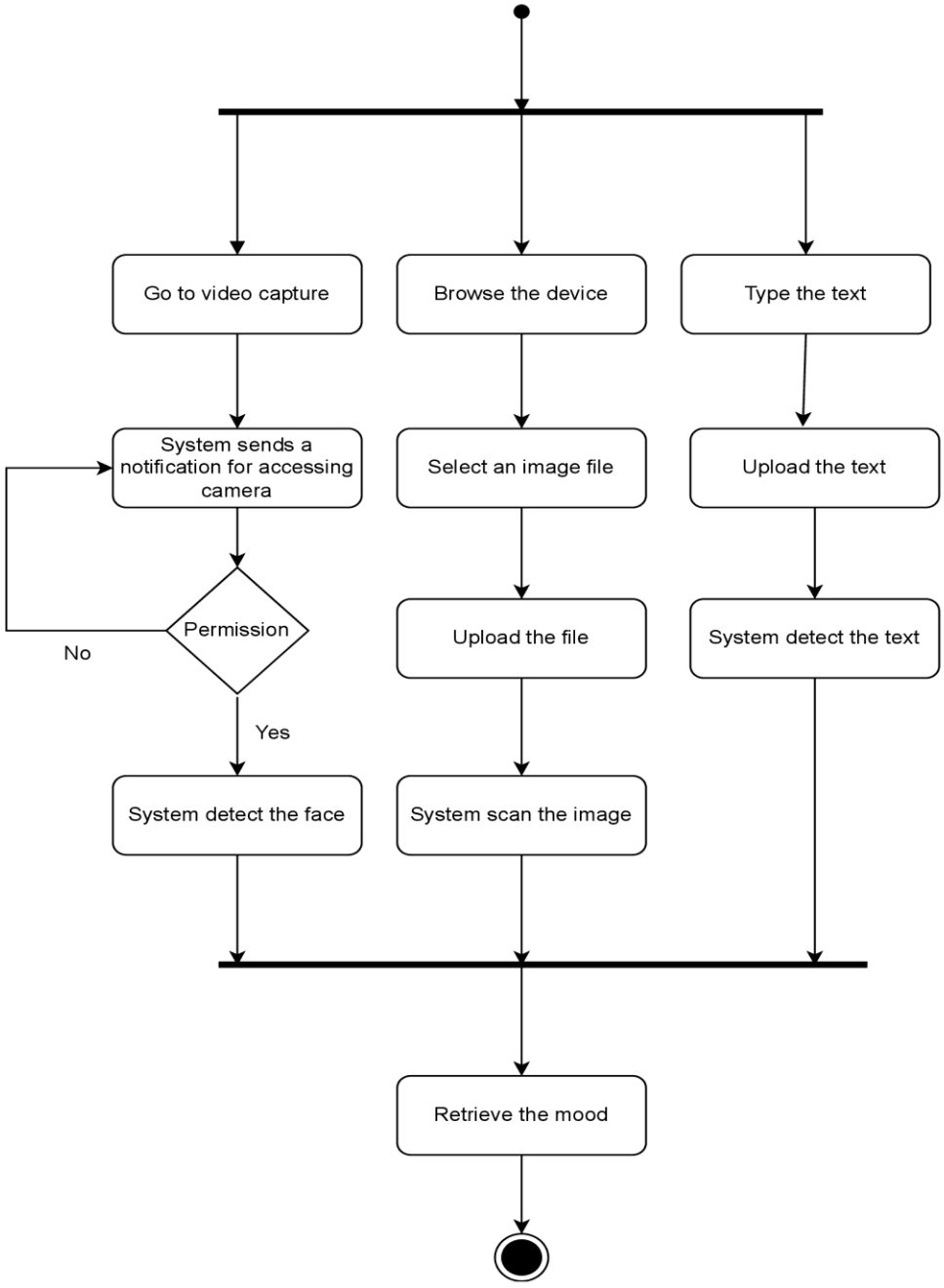

Figure 10: Retrieve Mood

Activity diagram (Approved mood)

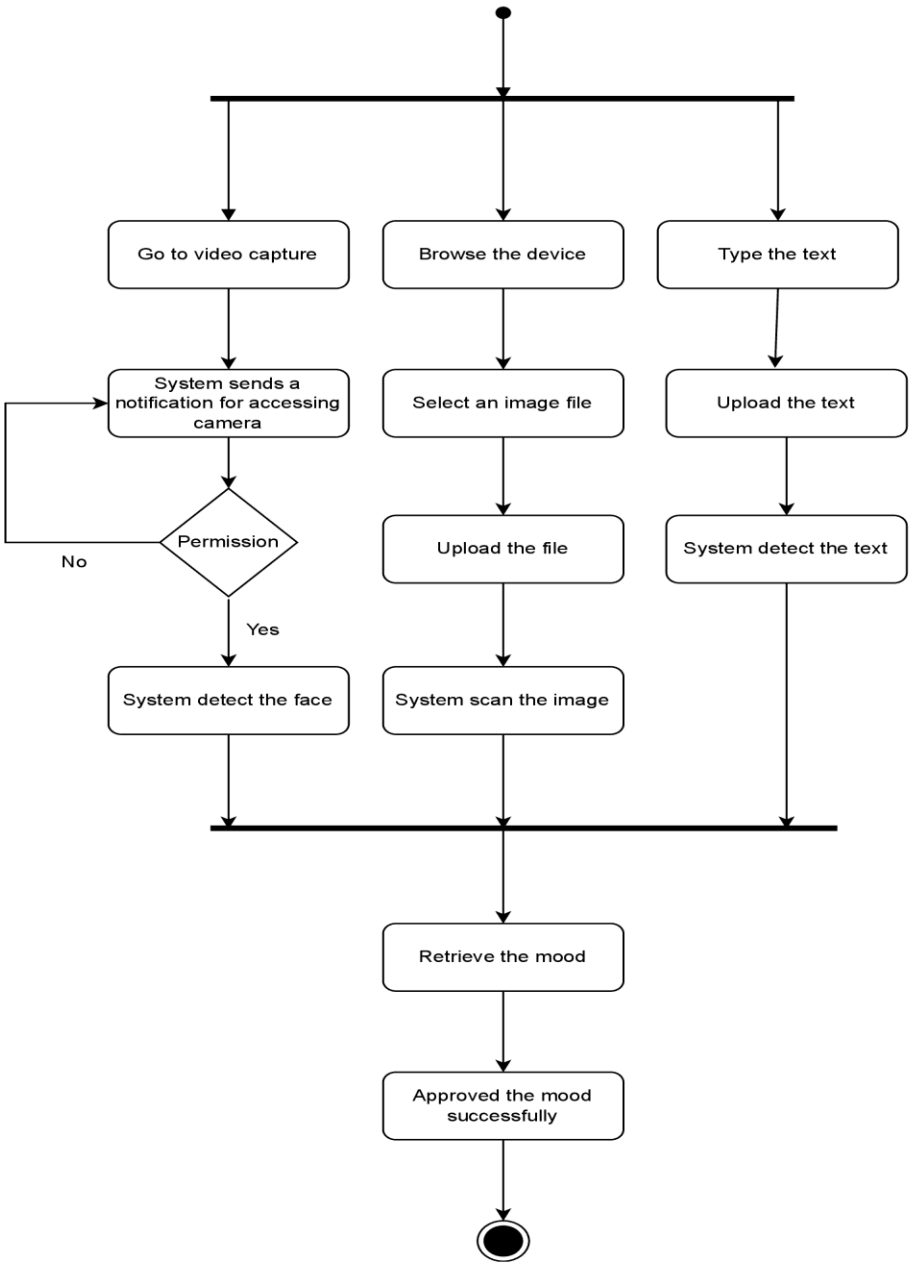

Figure 11: Approved Mood

Activity diagram (Retrieve Genre)

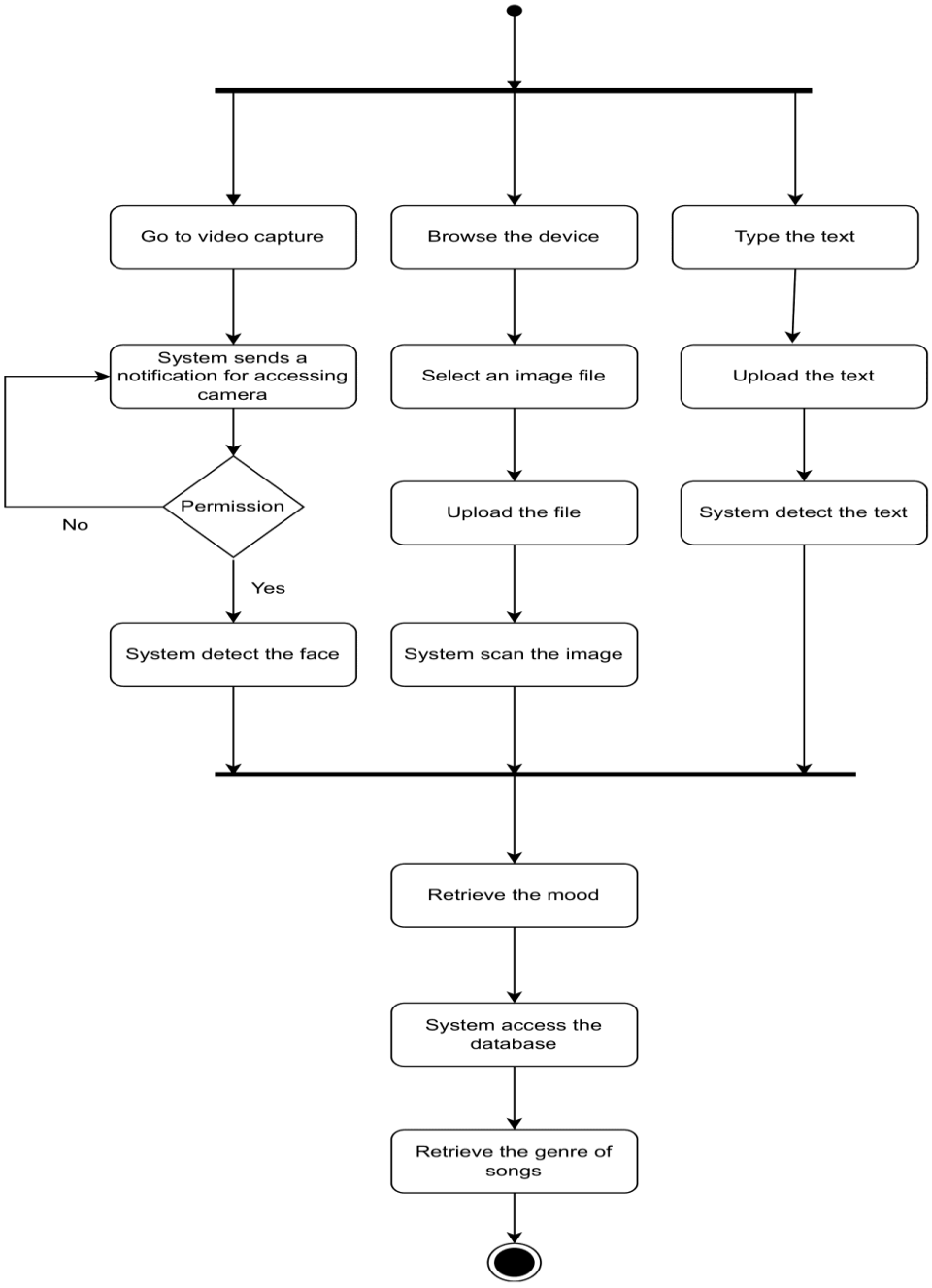

Figure 12: Retrieve Genre

Activity diagram (Playlist)

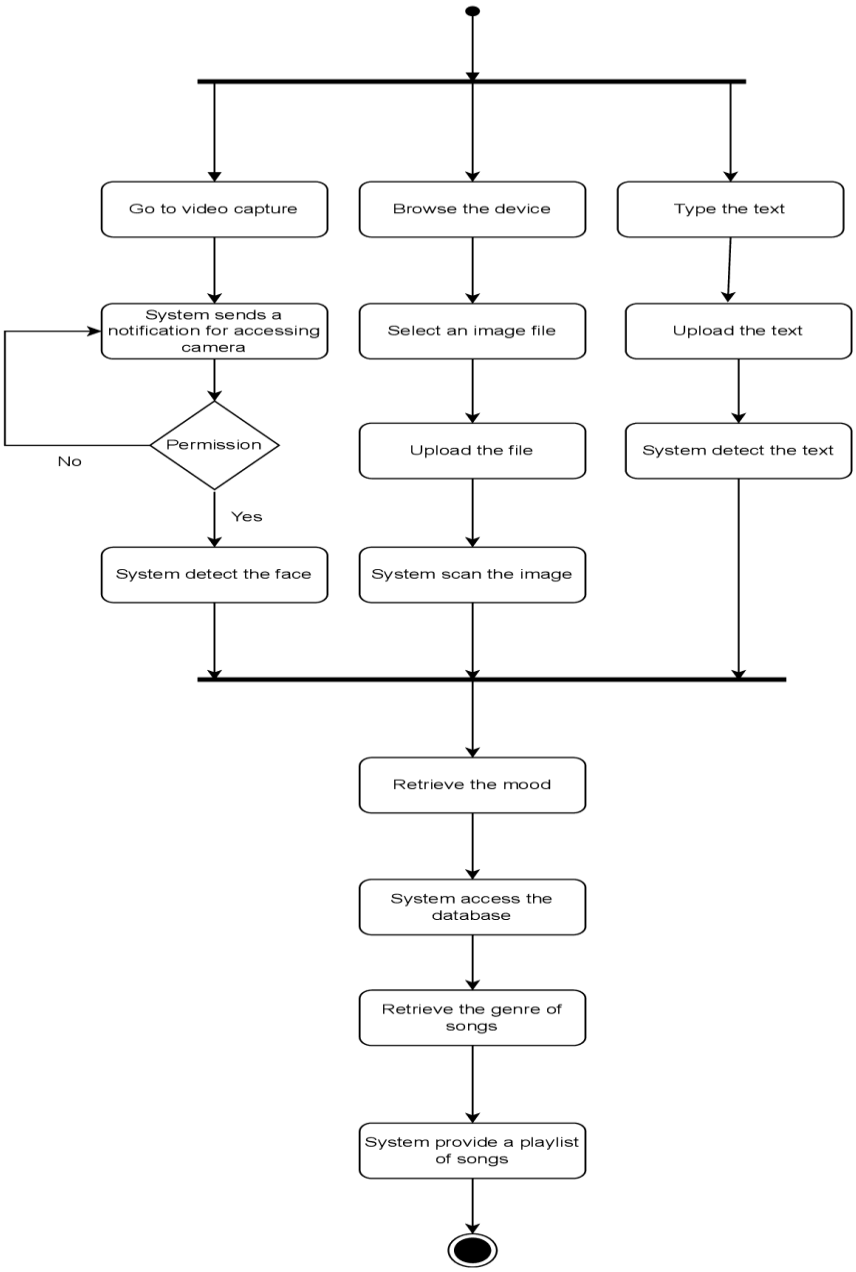

Figure 13: Playlist

Activity diagram (Control music)

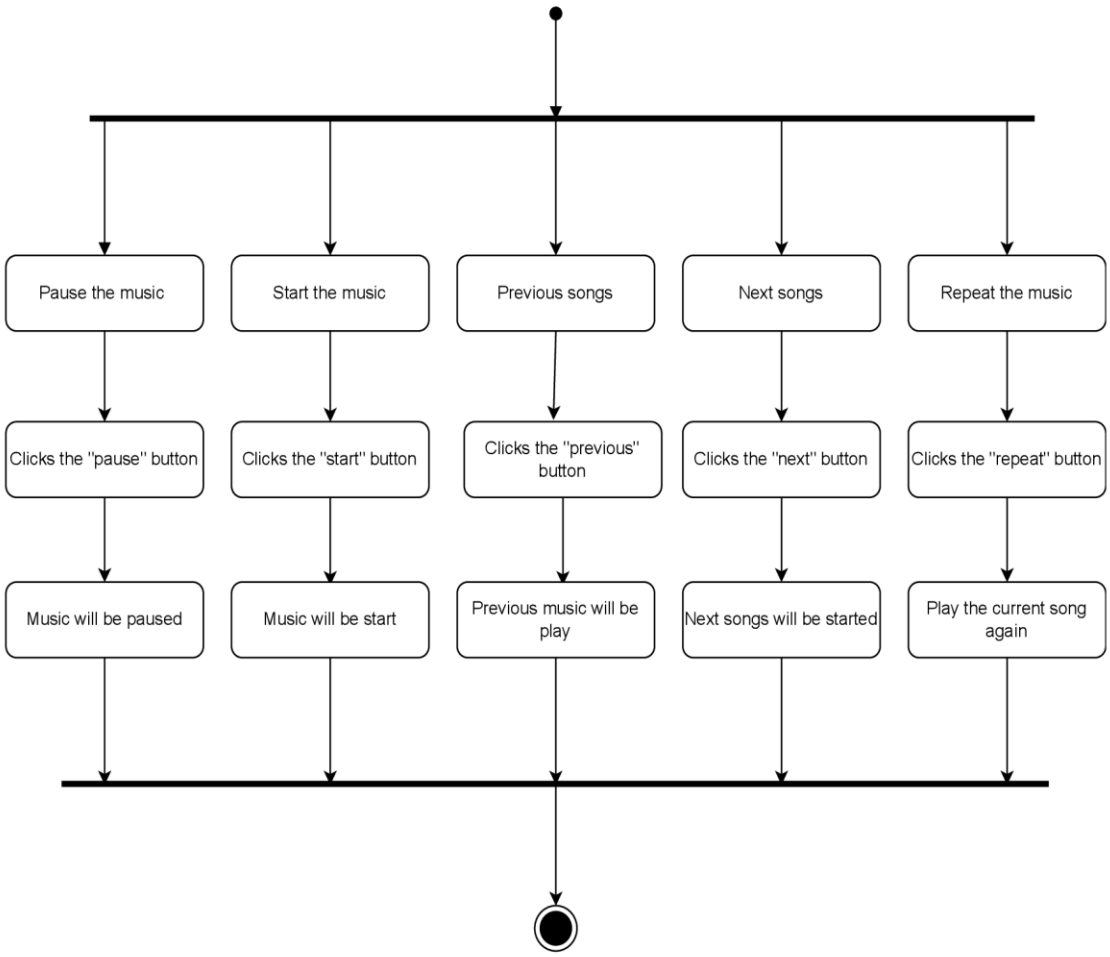

Figure 14: Control Music

Activity diagram (Manage database)

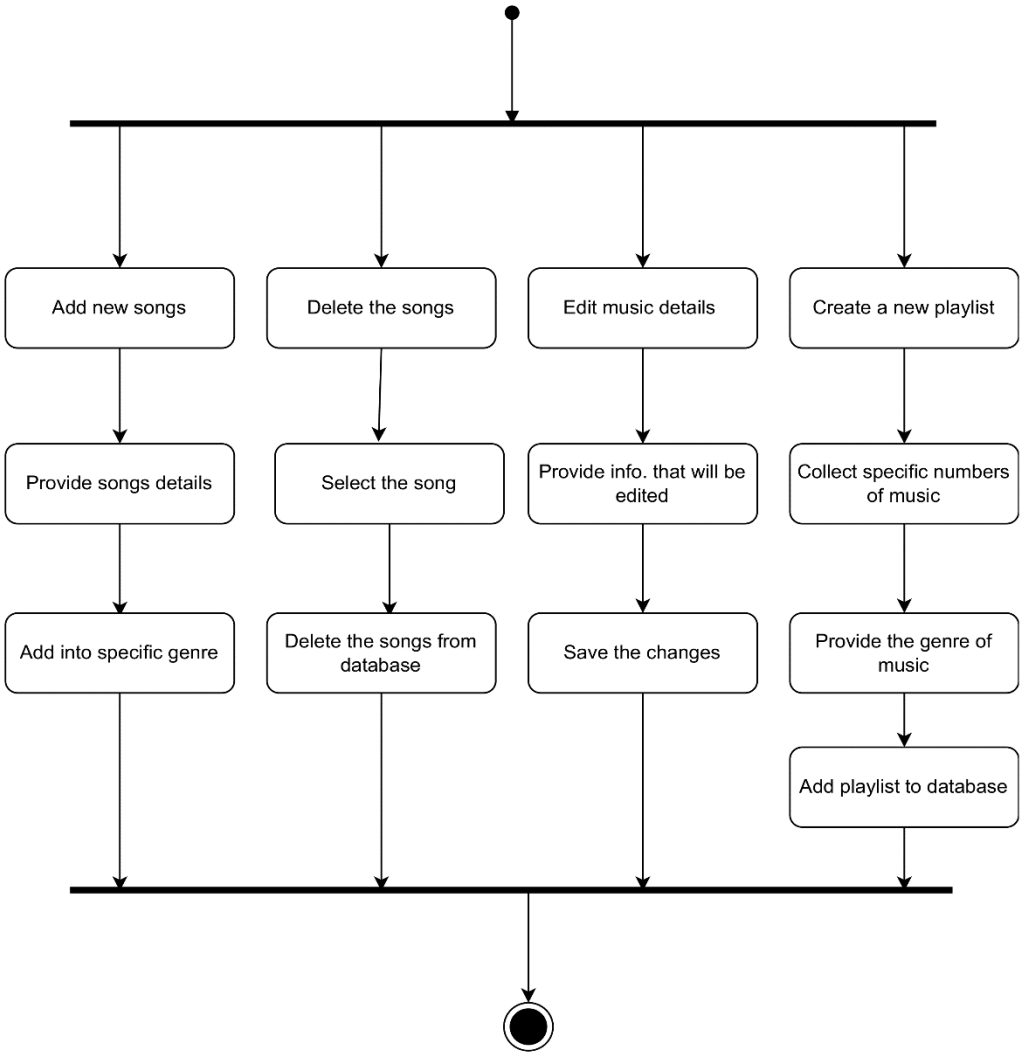

Figure 15: Manage Database

Activity diagram (Notification)

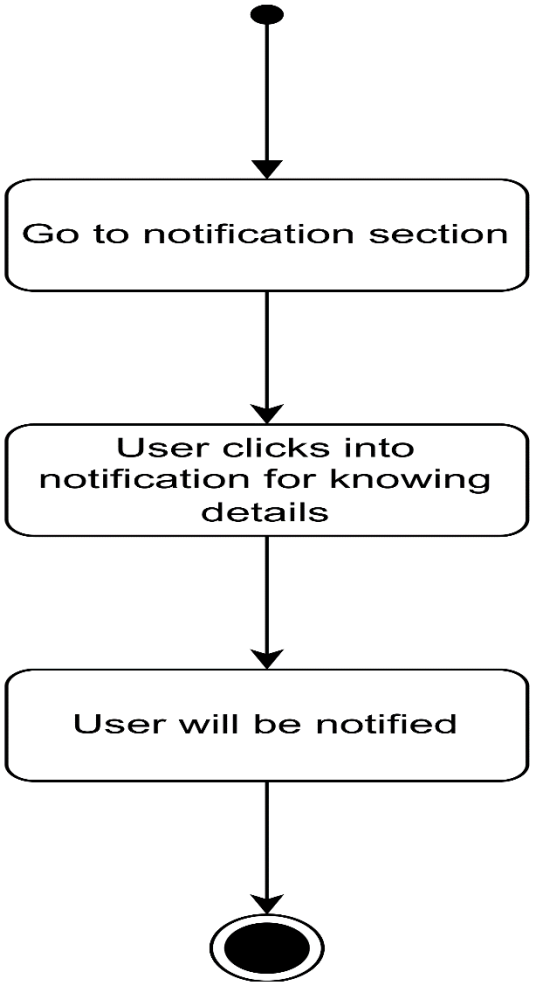

Figure 16: Notification

Activity diagram (Feedback)

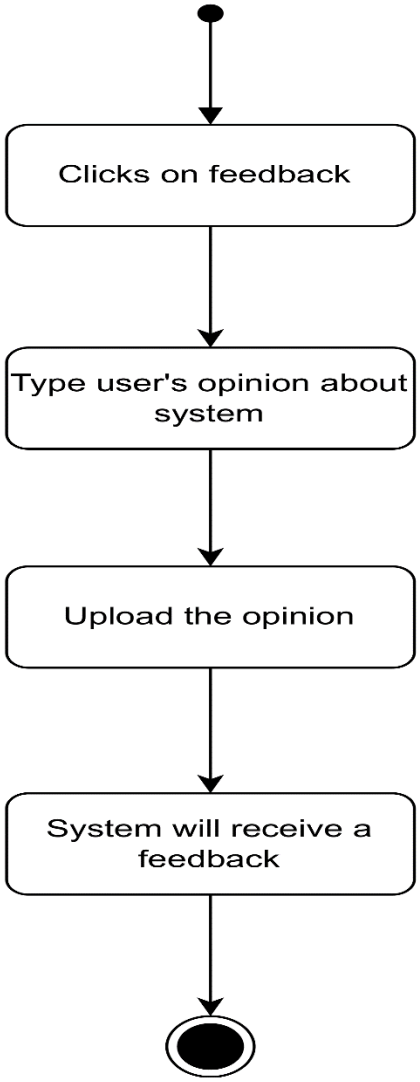

Figure 17: Feedback

Activity diagram (Change password)

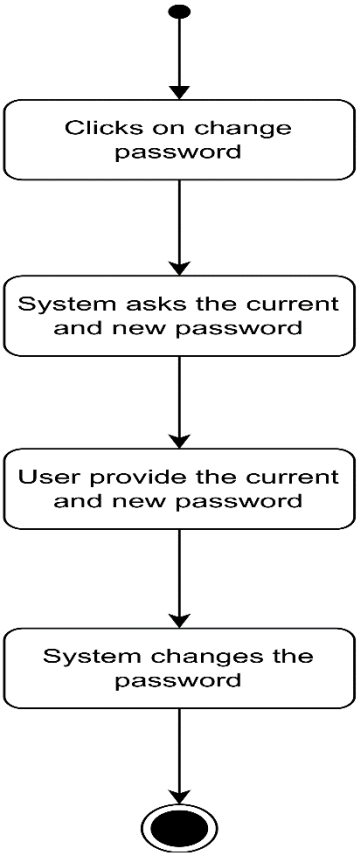

Figure 18: Change Password

Activity diagram (Remember me)

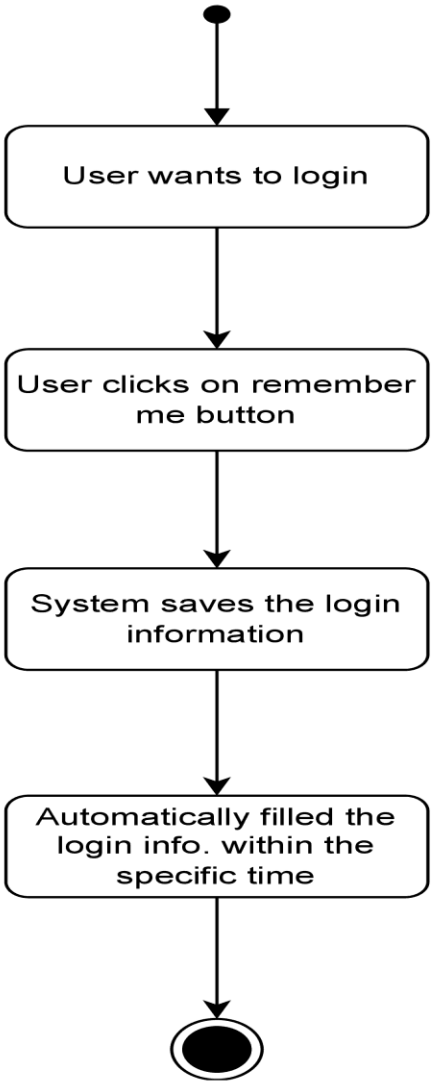

Figure 19: Remember Me

Activity diagram (Edit profile)

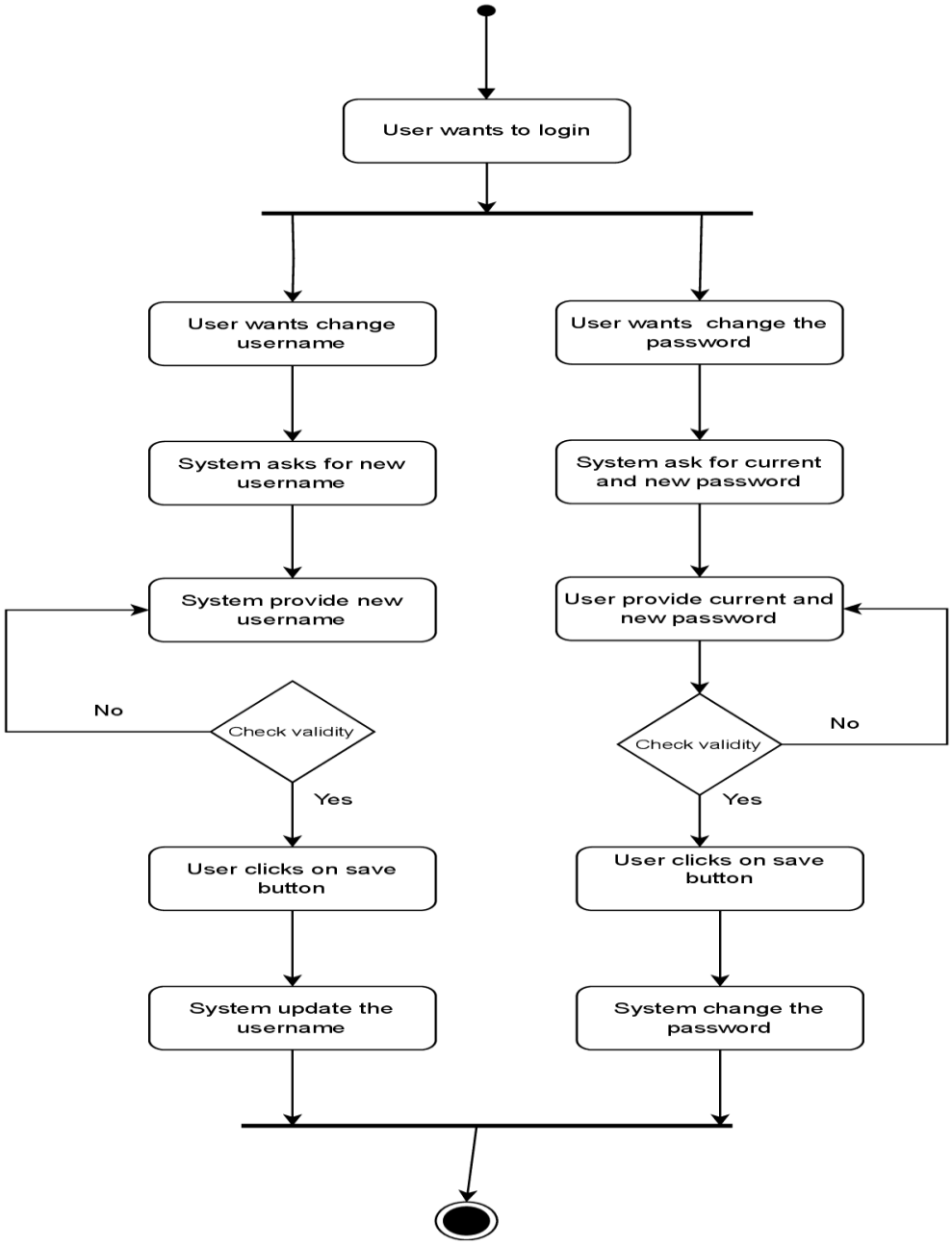

Figure 20: Edit Profile

Activity diagram (Log out)

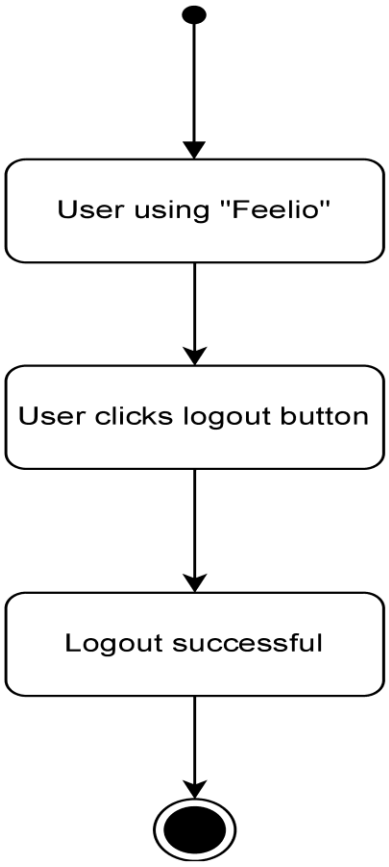

Figure 21: Logout

9 Requirement Traceability Matrix

9.1 Functional Requirements

FR1- User Registration and Login to a registered account.

FR2- Emotion detection from uploaded images.

FR3- Emotion detection from uploaded text.

FR4- Emotion detection from real-time video streaming.

FR5-Classification of emotions.

FR6-Music playlist generation.

FR7-User feedback mechanism.

FR8-Notification mechanism.

FR9- Real time emotion updates.

FR10- Admin control

FR11- - Log out from their account.

9.2 Use Case

UC1- Registration

UC2- Login

UC3- Activate account

UC4- Get video

UC5- Image upload

UC6- Get image

UC7 – Text upload

UC8- Get text

UC9- Retrieve mood

UC10- Approved mood

UC11- Retrieve genre

UC12- Playlist

UC13- Control music

UC14- Manage database

UC15- Notification

UC16- Feedback

UC17-Logout

A traceability matrix is a document, usually in the form of a table, used to assist in determining the completeness of a relationship by correlating any two baselined documents using a many-to- many relationship comparison. It is often used with high-level requirements (these often consist of marketing requirements) and detailed requirements of the product to the matching parts of high- level design, detailed design, test plan, and test cases.

UC / FR	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC8	UC9	UC10	UC11	UC12	UC13	UC14	UC15	UC16	UC17
FR1	✓																	
FR2					✓	✓												
FR3							✓	✓										
FR4				✓														
FR5															✓			
FR6												✓	✓					
FR7																	✓	
FR8																✓		
FR9															✓			
FR10	✓																	
FR11	✓																	

10 Appendix

10.1 Prioritization of requirements

We've prioritized the functional requirements by following Three-level Scale technique.

10.1.1 Three-level Scale

When a Business Analyst categorizes the requirements in any of the ordering or ranking scale, it is subject to the analyst's understanding of the business. Many analysts suggest that this method has some drawbacks and advocate methods that have more than one scale.

10.1.2 Prioritization of the requirements of "Feelio"

FR1 – High priority: Its essential requirement for our system. User access control is obvious to run this application.

FR2: High priority: It is very important requirement because user can receive a playlist from uploaded image.

FR3: High priority: Emotion detection from uploaded text is another very import requirement. Here user can receive a playlist from his/her typing text.

FR4: High priority: Emotion detection from real time video streaming is a high priority requirements here user can receive a playlist.

FR5: High priority: In the system emotion classification is very for system. Emotion classified into four categories such happy, sad, excitement, neutral.

FR6 – High priority: "Feelio" is basically an emotion based music player. Here there have been lots of songs in every playlist.

FR7 - Low priority: User can provide his/her opinion about the system.

FR8 - Low priority: It is slightly important that user will be notified about any matter of system if system want.

FR9: -Medium priority: The system continuously monitor and update the user's emotion.

FR10- Medium priority: There will be admin panel, they can control everything after logging in admin panel. Action will be taken by admin against anyone who posts fake or bad comments.

FR11 – Medium priority: The user will be able to log out of his account at the end of his need. Users will need to login again for later use.

11 Database Design

11.1 ER-diagram

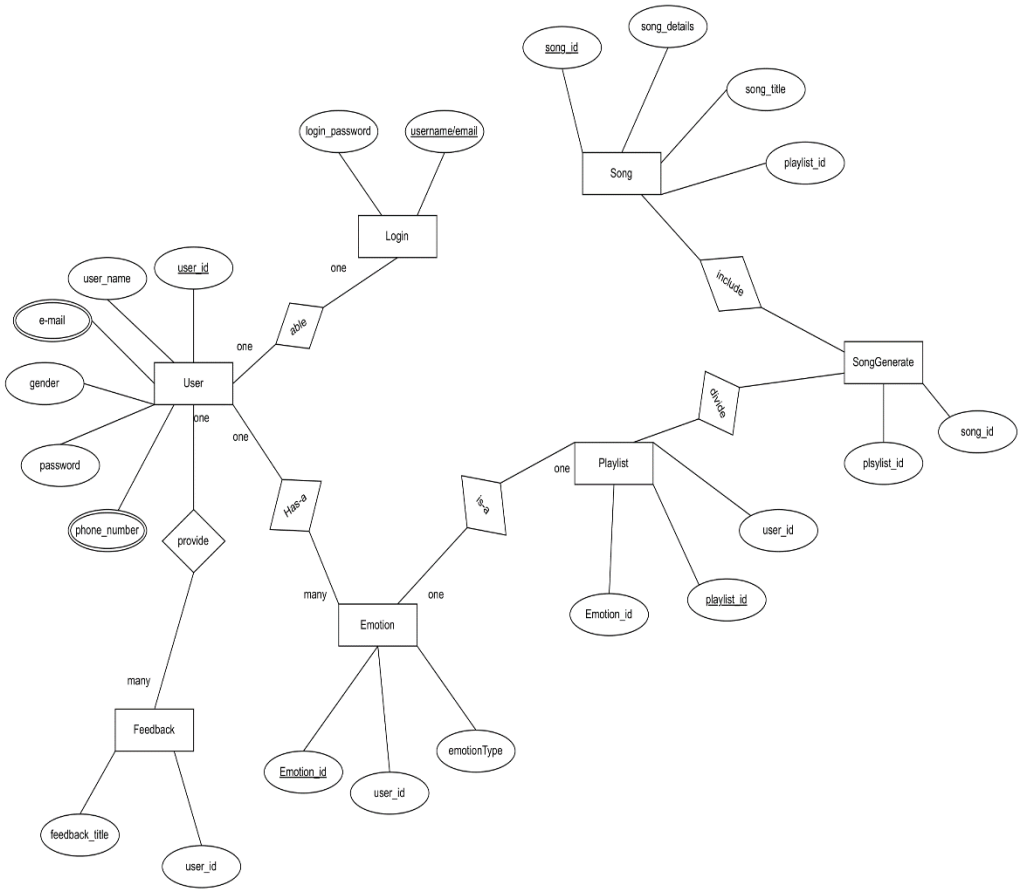

Figure 22: ER-Diagram

12. References

1. Software Engineering 9th Edition by Lan Sommerville.
2. Requirements Engineering Fundamentals by Klaus Pohl
3. Database System Concepts 6th Edition by Abraham Silberschatz
4. International Research Journal of Engineering and Technology (IRJET)