Tribhuvan University

Faculty of Humanities and Social Sciences

A PROJECT REPORT ON

Emotion Based Music Recommendation System

Submitted to

Department of Computer Application

Nepal Kasthamandap College, Kalanki

In partial fulfillment of the requirements for theBachelors in ComputerApplication

Submitted by:

Hitesh Dangol (14402080)

July, 2025

Under the Supervision of:

Achyut Shah Thakuri

Tribhuvan University

Faculty of Humanities and Social Sciences

Nepal Kasthamandap College

Supervisor's Recommendation

I hereby recommend that this project prepared under my supervision by Hitesh Dangol entitled "**Emotion Based Music Recommendation System**" in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

Achyut Shah Thakuri

Project Supervisor

BCA Department

Nepal Kasthamandap College

Tribhuvan University

Faculty of Humanities and Social Sciences

Nepal Kasthamandap College

Letter of Approval

This is to certify that this project prepared by Hitesh Dangol entitled "**Emotion Based Music Recommendation System**" in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

Achyut Shah Thakuri **Project Supervisor** BCA Department Nepal Kasthamandap College	Er. Sujan Poudel **Project Co-odinator** BCA Department Nepal Kasthamandap College
Roshan Poudel **Internal Examiner** Nepal Kasthamandap College	Ananda KC **External Examiner** Ratna Rajya Campus

Abstract

Emotion-Based Music Recommendation System is an intelligent online platform that recommends songs based on the user's current emotional state. Using a CNN-based algorithm, the system analyses facial expressions captured through a webcam to detect emotions such as happiness, sadness, anger, or surprise. Based on the detected emotion, it automatically generates a personalized music playlist to enhance the user's mood or match their current feeling.

The platform allows users to play, pause, and skip songs through a custom music player, offering a seamless and interactive listening experience. All user interactions and emotion data are securely stored in a database, ensuring personalized and organized music delivery. The system enhances user satisfaction by providing real-time, emotionally-driven music recommendations powered by machine learning.

This innovative approach transforms traditional music players by integrating AI and deep learning into emotional awareness, creating a smarter, more intuitive way to enjoy music.

Keywords: Emotion-Based Music Recommendation, CNN algorithm, facial emotion detection, AI music player, personalized playlists, deep learning.

Acknowledgement

We would like to express our sincere gratitude to our supervisor, Mr. Achyut Shah Thakuri, for giving us the valuable opportunity to work on this innovative project titled **"Emotion-Based Music Recommendation System"**. His guidance and encouragement throughout the project helped us explore and understand many new concepts, tools, and technologies, particularly in the field of artificial intelligence and emotion detection using TensorFlow. We are also deeply thankful to our BCA Program Coordinator, Er. Sujan Poudel, for his constant support, motivation, and valuable insights, which played a vital role in the successful completion of this project and our personal growth during this academic journey. Furthermore, we would like to extend our heartfelt thanks to our parents and friends for their continuous encouragement and support, especially during the challenging phases of this project. Finally, we express our gratitude to everyone who directly or indirectly contributed to the successful completion of this project.

Yours Sincerely,

Hitesh Dangol

Table of Contents

List of Figures

List of Tables

List of Listing

List of Abbreviation

Abbreviation	Description
CSS	Cascading Style Sheet
DBMS	Database Management System
DFD	Data Flow Diagram
HTML	Hypertext Markup Language
JS	JavaScript
MySQL	My Structured Query Language
PHP	Hypertext preprocessor
XAMPP	X-Platform, Apache, MySQL, PHP, Perl
EBMRS	Emotion Based Music Recommendation System

Chapter-1. Introduction

1.1 Introduction

Human emotions, including mood and psychological well-being, are influenced by music. User preferences, listening history, and genre-based filters are the key elements of traditional music recommendation systems. However, these strategies often fail to adapt to the listener's emotional state at that moment.

An Emotion-Based Music Recommendation System (EBMRS) aims to enhance the user experience by suggesting music based on recognized emotions. This system analyzes voice tones, facial expressions, and physiological signals (like heart rate) to determine the user's emotional state using artificial intelligence (AI), machine learning (ML), and signal processing techniques. This analysis informs the algorithm's music recommendations, which either suit or improve the user's mood.

1.2 Problem Statement

Traditional music recommendation systems use user preferences, listening history, and genre-based algorithms to suggest songs. However, these methods often fail to capture the listener's emotional state in real time, leading to recommendations that may not be consistent with their current mood or psychological needs.

Music has a big impact on emotions, and listening to the right music at the right time can improve mood, reduce stress, and enhance overall wellbeing. However, searching for relevant songs by hand can be time-consuming and inefficient. It is necessary to have a system that can provide real-time, personalized music recommendations while also dynamically evaluating a user's emotions.

This project's objective is to develop an Emotion-Based Music Recommendation System (EBMRS) that makes use of voice analysis, facial recognition, physiological signals, or

1.3 Objectives

- To create a portal for playing music.
- To suggest music based on real time emotion detection.

1.4 Scope and Limitations

The scope of the Emotion-Based Music Recommendation System involves creating an intuitive and user-friendly web platform that recommends music based on the user's emotional state. The system will use facial expression analysis through a webcam to detect emotions such as happy, sad, or anger. Based on the detected emotion, it will suggest personalized playlists as the user's mood. The platform will feature a custom music player with controls like play, pause, next, and repeat.This system aims to make music discovery more meaningful by connecting songs to the listener's current emotional state. However the system has following limitations

- Limited Emotion Accuracy
- Camera and Lighting Dependency
- One Emotion per Session
- Offline Usage Constraints

1.5 Development Methodology

A systematic approach to software development, the Agile Method is a flexible, iterative method to software development and project management that prioritizes quick turnaround, client feedback, and teamwork. Agile promotes flexibility and ongoing improvement over the course of a project. This model will be used in the development of the Emotion-Based Music Recommendation System (EBMRS), guaranteeing accurate specifications, multiple stages, and extensive testing prior to deployment.

- **Gathering requirements:** This phase involves understanding what the users need and how the system should help them. Instead of defining every detail upfront, the team works in short, focused cycles to gather enough information to start building the system.
- **Design:** In this phase, the emotion based music recommendation system is designed based on the gathered requirements. This includes defining the user interface, database design, and other key system components.
- **Develop:** Work can begin once the team has defined the requirements. Designers and developers begin to work on their projects. This project aims at deploying a working product.
- **Test:** In this phase, testing the system to identify and finding errors are done. This phase helps to ensure the development meets the specified requirements as given requirement.

- **Deploy:** The emotion based music recommendation system is made available for usage following the ending of testing. This include setting up servers, allowing user access to the system, and ensuring an easy transition from testing to operation.
- **Review:** This phase in agile is when the client checks if the product is meeting their needs and gives continuous feedback if any changes are required.

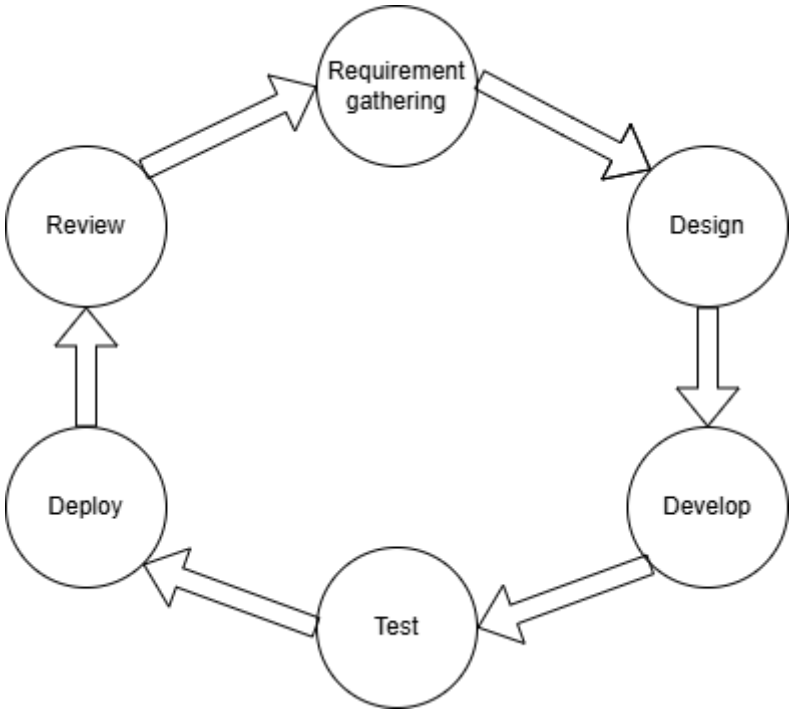


Figure 1.1 Agile method for EBMRS

1.6 Report organization

Table 1.1 Outline of the Report of EBMRS

Introduction	An overview of the Emotion-Based Music Recommendation System is given in this document, along with a discussion of its main goals, the main problems with conventional music recommendation techniques, the system's scope, and its limits. It provides a thorough grasp of how the system seeks to improve the music listening experience by suggesting songs that correspond with the user's present emotional state. It also explains the development approach used throughout the system's design.

Background Study	The limitations of traditional music recommendation systems, which depend on human input or listening history, are examined in this section. It examines the difficulties and shortcomings of these systems, such as their incapacity to react to the emotional condition of a user in real time. These drawbacks emphasize the necessity of an emotion-driven, automated music recommendation system that can make dynamic music recommendations based on face expression identification, enhancing user engagement, customization, and experience. .
System Analysis	The Emotion-Based Music Recommendation System's functional and non-functional needs are described in this part. Key features like emotion recognition, music playback, and user management are included, along with performance criteria like system responsiveness and user-friendliness. To determine whether the system is technically feasible, operationally efficient, and financially feasible for local setup, a feasibility analysis is also carried out.
Implementation and Testing	An overview of the programming languages, frameworks, libraries, hardware components, and platforms utilized in the creation of the Emotion-Based Music Recommendation System is given in this section. In order to make sure the system is dependable, effective, and user-friendly, it also describes the testing methods used to assess the system's performance, emotion recognition accuracy, and general user experience.
Conclusion	The results and overall effects of the Emotion-Based Music Recommendation System are summed up in this section. It highlights the benefits attained, such as improved user involvement, tailored music recommendations, and a smooth emotion-based recommendation system. In order to increase the system's functionality, it also suggests future directions include enhancing emotion recognition accuracy, adding

	additional emotion categories, incorporating real-time feedback mechanisms, and looking into support for internet streaming platforms.

Chapter-2. Background Study and Literature Review

2.1 Background Study

The ability of music to affect human emotions and mental states has long been acknowledged. Different kinds of music can evoke a range of emotions, including happiness, sadness, relaxation, and excitement, according to research in psychology and neuroscience. Collaboration filtering, content-based filtering, or hybrid models that take into account user preferences, listening history, and metadata like genre, artist, and popularity are the mainstays of traditional music recommendation systems. However, because these methods fail to consider a user's emotional state in real time, they frequently produce recommendations that might not be in line with their present state of mind.

Advances in signal processing, machine learning, and artificial intelligence (AI) have made emotion recognition technologies a viable way to improve music recommendations. Numerous methods, such as physiological signal monitoring, speech emotion analysis, and facial expression recognition, can be used to detect emotions. While voice emotion analysis identifies emotions based on tone, pitch, and modulation, facial recognition technology uses micro expressions to categorize emotions. Furthermore, wearable sensors can record physiological indicators like skin conductance and heart rate variability, which offer deeper insights into emotional states.

Spotify, Apple Music, YouTube Music, and other commercial music streaming services rely heavily on user behaviour and past preferences, but they don't have real-time emotional adaptation. Although some emotion-aware recommendation systems based on research have been created, they have drawbacks like poor accuracy, hardware dependence, and privacy issues. An Emotion-Based Music Recommendation System (EBMRS) offers a novel solution in light of the increasing interest in mental health applications and personalized user experiences. By dynamically modifying music recommendations based on real-time emotions, such a system can promote mental health therapy, improve stress management, promote relaxation, and increase user engagement. In order to produce a genuinely customized and emotionally responsive music experience, this study investigates the viability, approaches, difficulties, and possible effects of combining AI-driven emotion recognition with sophisticated music recommendation algorithms. [1]

2.2 Literature Review

The difficult task of recommending music based on emotions has garnered a lot of research interest recently. This paper uses Support Vector Machines (SVMs) to propose a novel method for emotion-based music recommendation. Instead of using pronouns as inputs, our method uses images to better capture the user's emotional state. A dataset of pictures linked to various emotional states was gathered, and convolutional neural networks (CNNs) were used to extract features. Next, using the input image, we trained an SVM model to forecast the user's emotional state. Lastly, we suggested musical selections based on the user's emotional state using the SVM model's output. Our tests on a sizable dataset demonstrate that our strategy performs better than current techniques in terms of accuracy and user satisfaction. By offering tailored music recommendations according to the user's emotional state, the suggested method offers a promising path for emotion-based music recommendation systems that can improve the user experience. [2]

Music acts as a universal language in the fast-paced world of today, guiding us through happy and sad times alike. Finding the ideal song to fit our current emotional state, however, can frequently feel like a never-ending search through the enormous ocean of musical options. Our project presents a novel Emotion-Based Music Recommendation System in recognition of the profound relationship between music and emotions. This system analyzes users' facial expressions in real time to determine their current emotional state using the state-of-the-art Convolutional Neural Networks (CNN) technology. The system creates a music playlist based on the user's current moods and tastes by analyzing their facial expressions as seen in webcam streams. This music system, which is seamlessly integrated with well-known streaming services like YouTube and Spotify, seeks to reinvent the music discovery process by providing users with an emotionally compelling and realistic experience. [3]

Choosing which songs to listen to from the vast selection of songs available on the Internet can frequently be difficult. The expression of a song's emotions and, more significantly, its emotional impact on the listener are frequently overlooked in the realm of musical preferences, despite the fact that musical genre is crucial in forming and exhibiting social identity. Selecting music by genre and/or artist was essentially the only choice just a few decades ago. The availability of personalized recommendations and customized playlists on digital music platforms has significantly altered this. Music is typically chosen by people

according to their interests and mood. Humans are known to express themselves through their facial expressions. Many people think that when they reach a certain point in time, they have so many songs in their song library that they can't decide what to play. Therefore, creating a recommendation system that could identify a user's mood and make song suggestions could significantly cut down on the amount of time spent searching for music. [4]

A person's life can be greatly improved by music since it serves as a significant form of entertainment for listeners and music lovers and occasionally even offers a therapeutic approach. Many music players have been created with features like fast forward, reverse, variable playback speed (seek & time compression), local playback, streaming playback with multicast streams, volume modulation, genre classification, and more in the modern world of multimedia and technology, which is characterized by constantly growing advancements. Consequently, they are automatically generated according to the user's emotional state. A person's face can reveal a lot about their emotional state. A camera is used to directly extract the necessary input from the human face. One use for this input is to extract data that can be used to determine a person's emotional state. The list of songs that fit the "mood" that was determined from the previously supplied input can then be obtained using this data. This helps create a suitable playlist based on a person's emotional characteristics and does away with the laborious and time-consuming process of manually sorting or organizing songs into various lists. [5]
A facial emotion-based music recommendation system has been created as a result of the development of facial emotion recognition technology. This system analyzes the listener's facial expressions to determine their emotional state and suggests music that reflects it. The architecture and implementation of a facial emotion-based music recommendation system, along with the underlying algorithms and technologies, are examined in this paper. Along with possible future research directions, we also go over the system's drawbacks and difficulties. [6]

Demand for customized music recommendation systems that can recognize and accommodate individual preferences is rising as a result of the volume of digital music content. This study suggests a machine learning-based emotion-based music recommendation system that is implemented with Python. By suggesting music tracks based on emotional context, the system seeks to increase user satisfaction and engagement

while offering a more engaging and customized listening experience. A strong pipeline for preprocessing data, feature extraction from audio signals, and the creation of machine learning models trained on datasets labeled with emotions are important parts of the system. Python libraries for data manipulation, feature extraction, and model training include Pandas, NumPy, and Scikit-Learn. The system uses deep neural networks and other cutting-edge machine learning algorithms to extract high-level emotional features from audio data. The suggested system's accuracy in making recommendations, user satisfaction, and adaptability to shifting user preferences and emotional states are all evaluated. The outcomes, which show the efficacy and efficiency of the implemented emotion-based music recommendation system, are derived from user studies and objective metrics. [7]

These days, human emotion is very important.Human feelings, which can be expressed or not, are the foundation of emotion.Emotion conveys a person's unique behavior, which can take many different forms.Emotion extraction reveals each person's unique behavioral state. This project's goal is to identify emotion and extract facial features.and to play music based on the emotion that was identified. Nevertheless, a lot of current methods rely on historical data to recommend music, and the other algorithms are typically slow, less precise, and even call for extra hardware like physiological sensors or EEG. Either an integrated camera or a local capturing device records facial expressions.Here, we employ an algorithm for the identification of the feature from the picture that was taken. Therefore, the suggested system will play music automatically based on the facial expressions that are recorded. [8]

Many kinds of music are now easily accessible through music platforms. They keep working to enhance research and music organization in order to address the selection issue and facilitate finding new music. In order to assist people in selecting the appropriate music for each circumstance, recommendations have grown in popularity. However, there is still a distinction between recommendations that are emotion-focused and those that are personal. Humans benefit from music, which is frequently used to reduce stress and illness, control emotions, and manage both mental and physical activity. There are numerous therapeutic domains and uses for music therapy to enhance well-being. The design of music recommendations that are informed by the user's emotions, ideas, and activity content will be covered in this article. We developed a Convolutional Neural Network (CNN) model specifically designed to suggest music based on the user's facial expressions, assisting

individuals in selecting music that suits their various moods. Its testing accuracy is 76% and its validation accuracy is 97%. [9]

The process of converting an image into digital form and applying various operations to it is known as image processing. This is done to improve the image or draw out important details. One way to communicate nonverbally is through facial expressions. There are eight universal facial expressions: contempt, disgust, fear, anger, happiness, sadness, surprise, and neutral. Therefore, it is crucial to recognize these facial emotions. A technologically based monitoring system for the elderly that uses video images to identify emotions. Our suggested system uses video analysis technology, which uses video data to enable real-time monitoring of senior citizens' living conditions. The system will send a message to their children and family in the event of an emergency. [10]

This essay offers a thorough analysis of facial recognition technology, outlining its development over time, examining its uses and functionality, talking about the difficulties it faces, and speculating about its potential. From the beginning to the present, the development of the technology is traced, emphasizing the significant breakthroughs that have influenced it. The diversity and complexity of facial recognition technology are highlighted by an examination of several system types, such as 2D, 3D, and thermal. A thorough explanation of the steps involved in facial recognition is given, including data collection, face detection, feature extraction, and matching. We also explore the wide range of its applications in various fields, including social media, healthcare, retail, smartphone authentication, security, and surveillance. Facial recognition technology has many uses and amazing advantages, but it also has some significant drawbacks. These include issues with accuracy, ethical and privacy ramifications, and the requirement for thorough regulatory frameworks. A discussion of possible advancements and growth projections for facial recognition technology is included in the paper's conclusion. This review offers a thorough understanding of facial recognition technology, highlighting both its future implications and its applicability in our technologically advanced world. [11]

Chapter-3. System Analysis and Design

3.1 System Analysis

3.1.1 Requirement Analysis

Functional requirement

A thorough examination of functional requirements is necessary for the creation of an Emotion-Based Music Recommendation System (EBMRS). This guarantees that the system can accurately identify emotions and offer timely, pertinent music recommendations. The following are the key requirement that are collected

- **Emotion Detection**: Use of facial expression recognition to determine the user's emotional state.
- **Music Recommendation Engine**: Mapping detected emotions to suitable music genres or individual songs.
- **User Interface (UI)**: Interactive and user-friendly design for selecting input methods and adjusting preferences.
- **Music Database Integration**: Access to an extensive music library from online streaming platforms or local storage.

Use Case Diagram

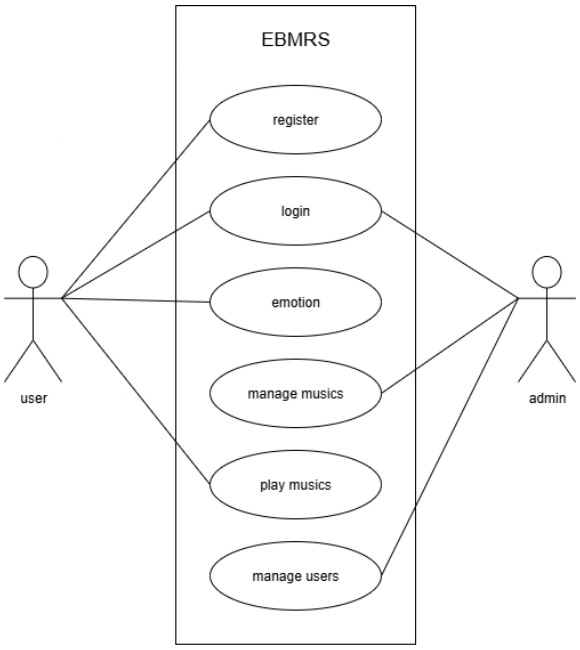


Figure 3.1 Use Case Diagram

The use case diagram for an emotion-based music recommendation system is displayed in the above figure. There are two actors in it: a user on the left and perhaps an administrator or external system on the right. The system offers five primary features (use cases): music access, playlist management, emotion detection, registration, and login. By interacting with these features, the user may choose music according to their mood. The external system may help with data processing or backend functions.

Non-functional requirement

- **Maintenance:** System will be easily maintainable, accessible to all users, and smoothly interact.
- **Performance:** System performance is smooth and accurate as the system database is normalized so it provides quick operation.

3.1.2 Feasibility Study

Before developing this platform, we conducted a feasibility study to access the technical, operational and economical feasibility of this project. The study is shown below

Technical Feasibility

To implement this proposed system and making it more feasible, the main tools and technologies that are used are as follows:

Table 3.1 Technical Feasibility Study Table of EBMRS

Technologies requirements	**Hardware requirements**	**Software requirements**
HTML	Laptop	MS Word
CSS	-	Visual Studio code
JavaScript	-	XAMPP Server
MYSQL	-	Draw.io
PHP/Python	-	-

These software which we are going to be used while making this project are freely available and technical skills are manageable so it is technical Feasible

Operational Feasibility

The project is operationally feasible as the required resources including skilled developers, project managers, and customer support personnel are fulfilled cause we can do this all. The system is completed within the schedule time and does not exceed the time period.

Table 3.2 Gantt chart of EBMRS

Task/Date	16th February	1st March	6th April	3rd May	20th June
Planning					
Requirement gathering					
Sprint 1-Basic UI and DB Setup					
Testing Sprint 1					
Sprint 2-emotion detection model					
Testing Sprint 2					
Sprint 3-Reports and UI Enhancement					
Testing Sprint 3					
Deployment					
Documentation					

The image shows a Gantt chart illustrating a project timetable from February 16th to June 20th. It describes several tasks that will be completed during the project's phases. The planning phase begins on February 16th and is completed quickly, followed by the requirement gathering phase, which lasts until early March. Sprint 1, which focuses on the basic UI and database setup, starts in early March and runs until early April. Once Sprint 1 is completed, the testing phase begins immediately.

Sprint 2, which involves the development of the emotion detection module, begins following Sprint 1 and lasts until the end of April. Testing for Sprint 2 begins just before May and ends in the first week of May. Sprint 3, which focuses on report production and UI enhancement, starts in early May and lasts until mid-June, with a testing phase running concurrently at the end of development.

Deployment is scheduled just before the final deadline and is finished by June 20th. Documentation is maintained continually throughout the project's length, commencing on February 16th and ending in June. This Gantt chart clearly shows the project's structured flow, overlapping tasks, and Agile-based methodology, which includes dedicated development and testing sprints.

Economic Feasibility

The project is economically feasible as there is a high demand for a comprehensive music recommendation platform that offers a complete solution for listening platform. The platform can earn money from advertisement and subscription.

3.1.3 ER-Diagram

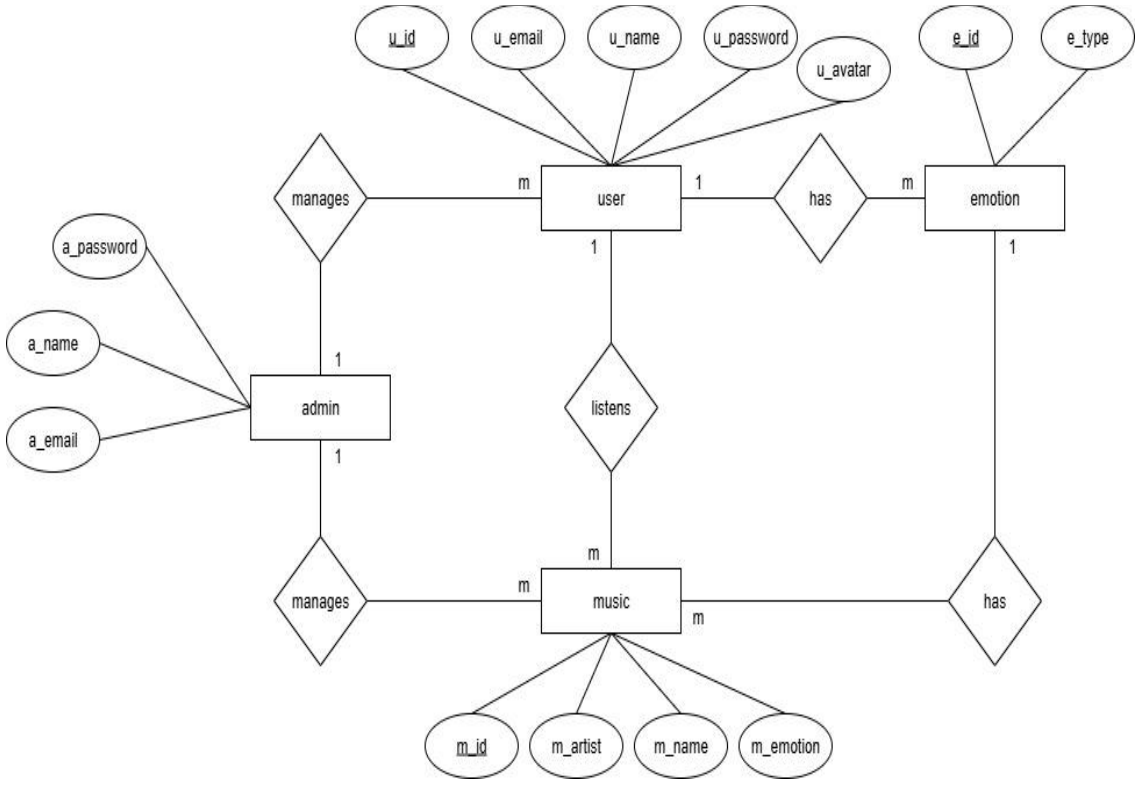


Figure 3.2 ER Diagram

A music streaming or recommendation system with four primary entities—User, Admin, Music, and Emotion is depicted in the following Entity-Relationship (ER) diagram. A user's ID, name, email, password, and avatar are among their properties. Multiple music tracks are available for users to listen to; this is represented by a many-to-many relationship known as "listens." Additionally, each user is linked to a single emotion), but numerous users may share the same emotion, creating a many-to-one relationship with the Emotion object.

ID, name, artist, and the emotion it attracts represent some of the attributes that are present in the Music entity. One emotion can be connected to a single musical track, and each emotion can be connected to several tracks—an additional many-to-one relationship. The system keeps track of the different music tracks that users listen to.

Music and users are managed by administrators. Although each administrator has the ability to oversee numerous users and music tracks, only one administrator is in charge of each user and music track, creating a one-to-many relationship in both situations. Admins themselves have personal information including their name, email address, and password.

All things considered, the figure shows how the system connects users, music, administrators, and emotions to produce a framework appropriate for administrative supervision and emotion-based music recommendations.

3.1.4 Data Flow Diagram (DFD)

Data flow diagram DFD is a graphical representaion of the flow of data in an information system. It is capable of depiction incoming data flow, outgoing data flow and stored data. The DFD does not mention anything about how data flows through the system

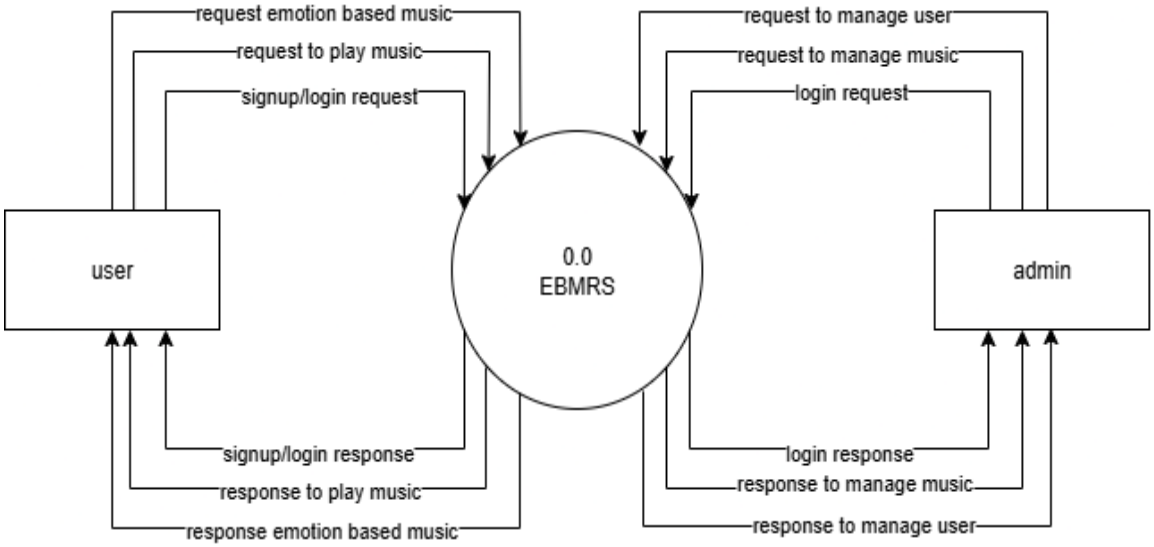


Figure 3.3 Level 0 DFD

An Emotion-Based Music Recommendation System (EBMRS) 0-level Data Flow Diagram (DFD) is displayed. It shows how data moves between users, administrators, and the system (represented by process 0.0 EBMRS). Users engage with the system by submitting requests to join up or log in, play music, and receive emotion-based music recommendations on the left side. After processing these requests, the system replies to the users with the relevant information, such as a login confirmation, the requested music, or music that is tailored to the user's emotional condition.

Admins utilize login requests and requests to manage user and music data on the right side of the figure. Following processing, responses pertaining to user or music management tasks and login confirmations are returned by the system. This DFD shows how data enters and exits the system for several basic functionality, outlining the high-level interactions between the EBMRS and external entities (users and administrators). It provides a basis for comprehending the main functions and functions of the system.

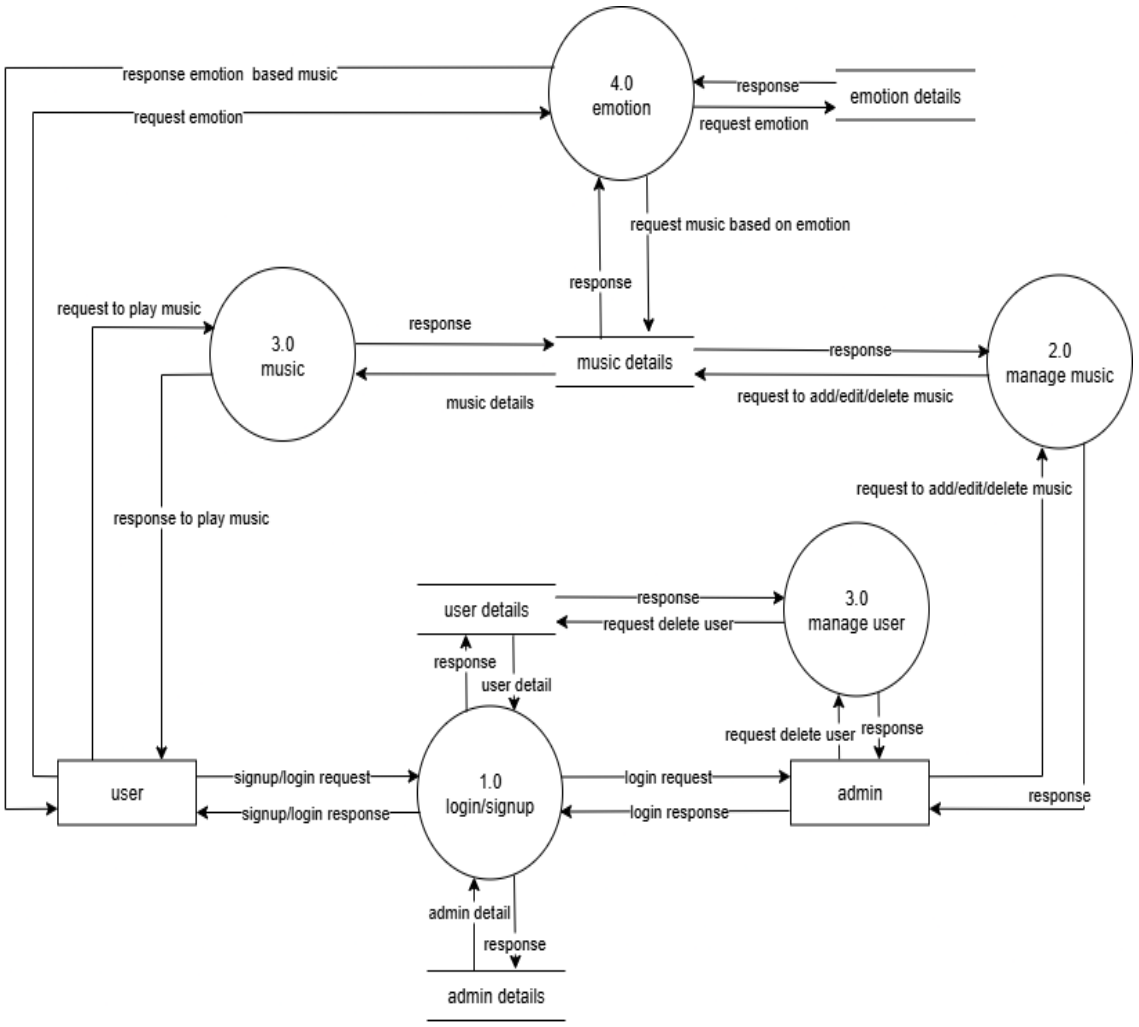


Figure 3.4 Level 1 DFD

The Emotion-Based Music Recommendation System (EBMRS) Level-1 Data Flow Diagram (DFD) illustrates the internal procedures and data flow between users, administrators, and system modules. Users and administrators provide their login credentials and receive authentication replies in the Login/Signup module (1.0), which starts the process. After logging in, users can ask the Music module (3.0) to play music or the Emotion module (4.0) to play music that is dependent on their emotions. Depending on the user's emotional state, this module retrieves appropriate music recommendations by interacting with external emotion data sources.

Admins use the Manage Music (2.0) and Manage User (3.0) modules to interact with the system after login in. In addition to managing user accounts, including deleting users, administrators have the ability to add, edit, and remove music. Music and user information are sent and received in accordance with these management activities. In addition to playing music upon user request, the Music module retrieves or updates music information in close

coordination with the Manage Music and Emotion modules. To retrieve information about emotions and play appropriate music, the Emotion module establishes connections with outside data sources.

Login/signup, music handling, user and admin management, and emotion-based suggestions are the main components of the EBMRS system that are broken down in this Level-1 DFD. It also shows how data flows between these components and how users and admins interact with each one.

3.2 System Design

To realize the different functional requirement of the system in graphical form, different design diagram of the system has been prepared which are as follow:

3.2.1 Architectural design

For the system, three tier architecture is used which includes user interface, web server and database. In architectural design, basic structure of the system is shown:

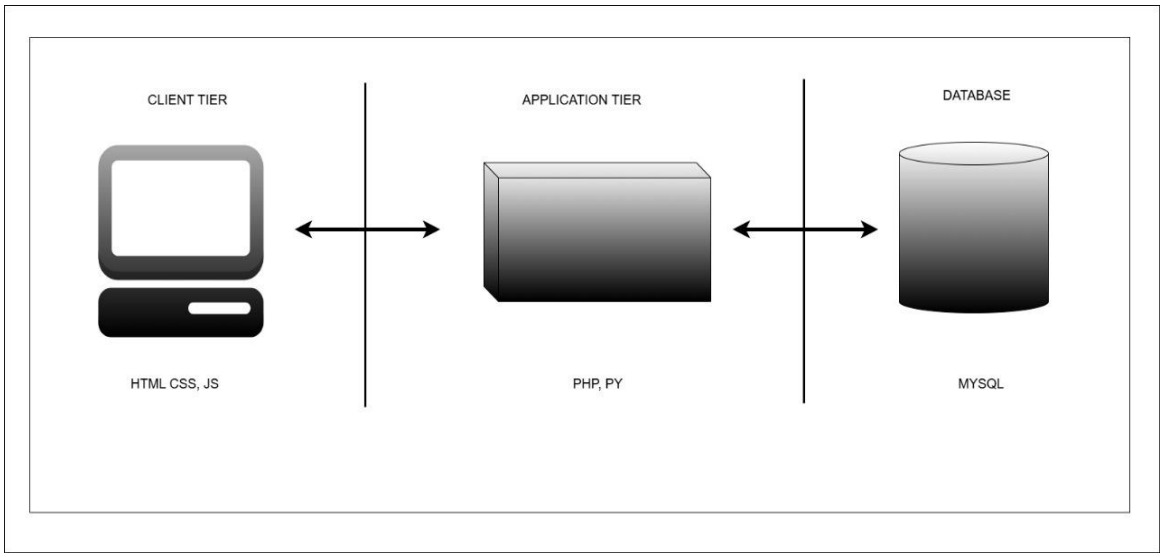


Figure 3.5 Architectural Design

3.2.2 System Flow Chart

A flowchart is a diagram of a process's individual steps shown in chronological sequence. It is a general-purpose tool that can be customized for a broad range of uses. It may be applied to explain a project plan, an administrative or service process, or a manufacturing process, among other processes. Here, the process to use website is shown step by step and determine what user and admin are capable to do.

The EBMRS flowchart illustrates each phase of the EBMRS website's activity, with each user having a separate role in running the system. The user flow on the EBMRS platform can be seen in this flowchart. Upon accessing the website, users will be asked to login. The system checks to see if the email address provided is valid. If so, it just goes on with the procedure; if not, it sends the user straight to the registration page where they must provide the necessary information. Following that, the system verifies if the individual is an administrator or a user in order to assign them the correct position.

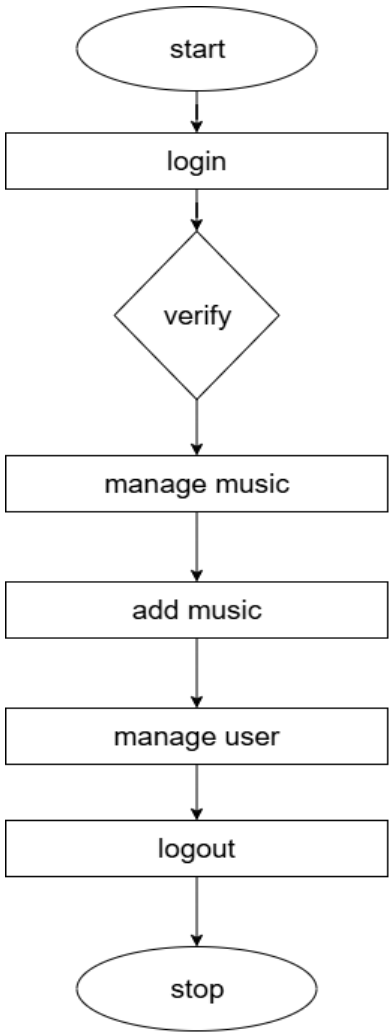


Figure 3.6 Admin Flow Chart

The flow diagram shows the steps taken in the admin area of a music management system. It begins with the "Start" node, which indicates the start of the administrative session. The initial step for the administrator is to log in using proper credentials. After logging in, the system checks the credentials using a decision-making process. If the verification fails, the flow returns to the login screen and prompts the administrator to try again. Upon successful

verification, the administrator gains access to the system's key functionalities. The administrator can manage existing music, which includes viewing, modifying, and deleting tracks. Additionally, the administrator can add new music to the system by uploading audio files and inserting essential metadata. The following step enables the administrator to manage user accounts, such as accessing user information or conducting administrative tasks on user profiles. When all tasks are done, the administrator logs out, signaling the end of the session, which is represented by the "Stop" node. This flow enables secure access control and orderly management of both music content and user data.

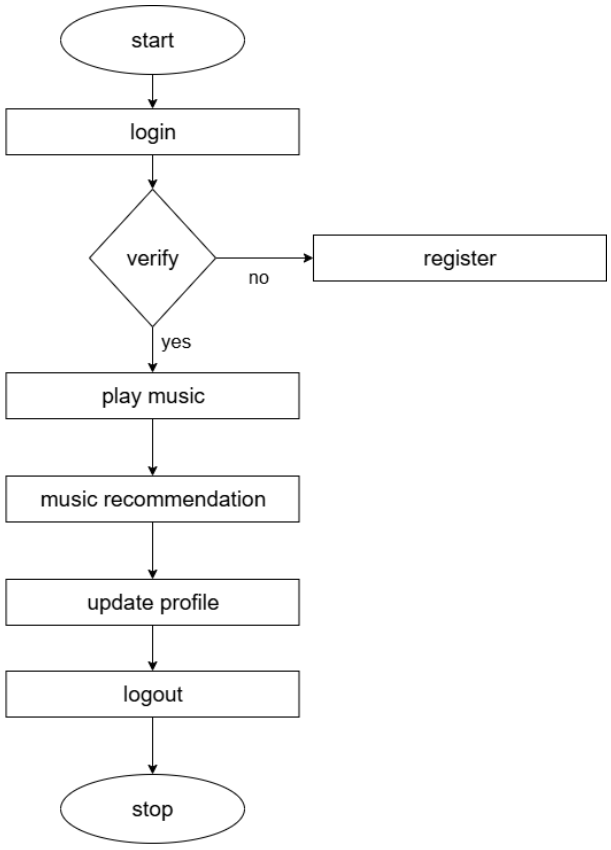


Figure 3.7 User Flow Diagram

The flow diagram depicts the User Flow of a music player system, outlining the processes that a user takes when interacting with the application. It begins with the "Start" point, which represents the start of the user's session. The initial action is "Login", in which the user inputs their credentials. A "Verify" decision block follows, which validates the login credentials.

If the login details are incorrect "No", the system returns the user to the "Register" page, where they can establish a new account. After successful registration, the user is likely to

be returned to the login screen (implied). If the login verification is successful "Yes", the user can use the application's major functions.

The first function offered is "Play Music," which allows users to listen to music within the app. Following that, the user can access "Music Recommendation," in which the system proposes music based on the user's tastes or previous activity. The user can then select ”Update Profile" to change personal information like their name and avatar.

When the user is finished using the system, they click "Logout" to securely end the session, which leads to the final "Stop" node, indicating that the interaction is complete.

Overall, this flow enables correct user authentication and provides users with access to critical services such as music recommendation, and profile management, all within a safe and controlled process.

3.2.3 Database Schema Design

The figure below is the database schema design of EBMRS. Database schema design is used to show the basic structure of the system. In this system, there are four tables in the database each of them has their own fields where their id is used in another table it becomes foreign key and foreign key are connected to another table with a line. There is data type of each entity and the foreign key in schema is represented by the arrow as shown in the diagram.

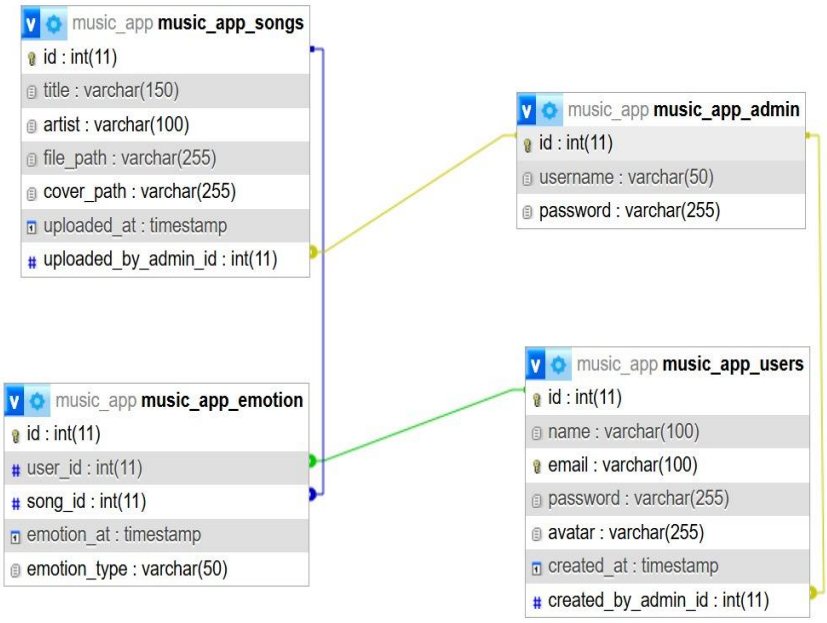


Figure 3.8 Database Schema

3.2.4 Wireframing

Wireframing is the process of developing a visual blueprint or guide for a software interface, app, or website. It helps in organizing the structure, functionality, and layout earlier than the actual design and development process

User Dashboard

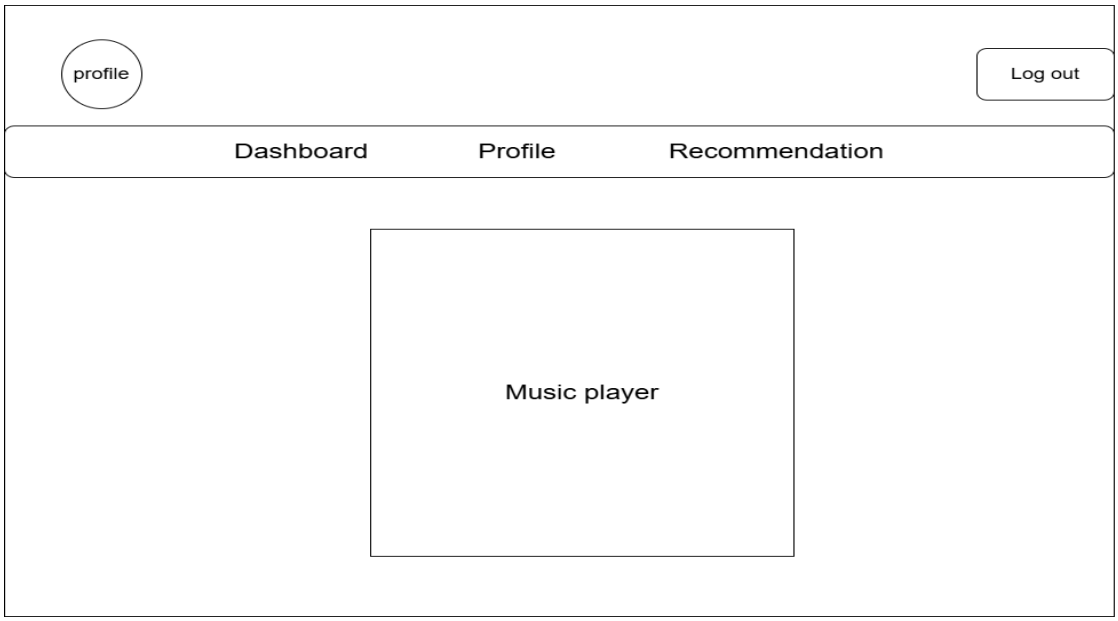


Figure 3.9 WireFraming Design of User Dashboard

Signup page

Figure 3.10 WireFraming Design of Signup page

Login page

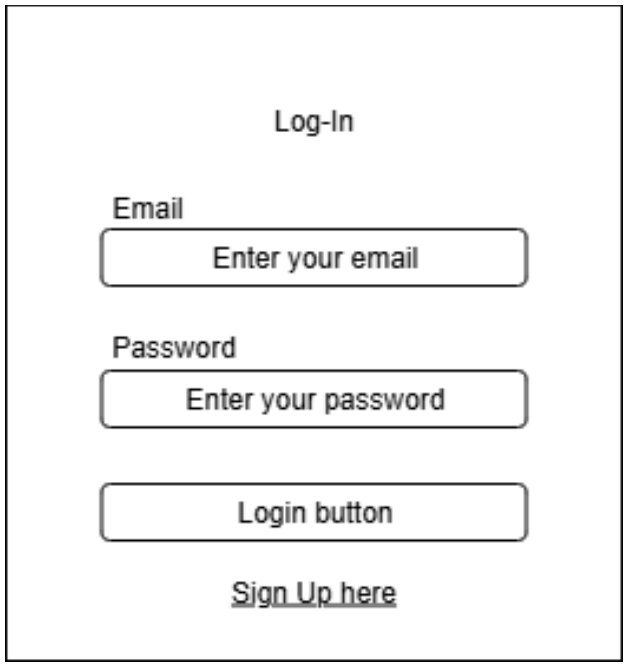


Figure 3.11 WireFraming Design of Login page

User profile page

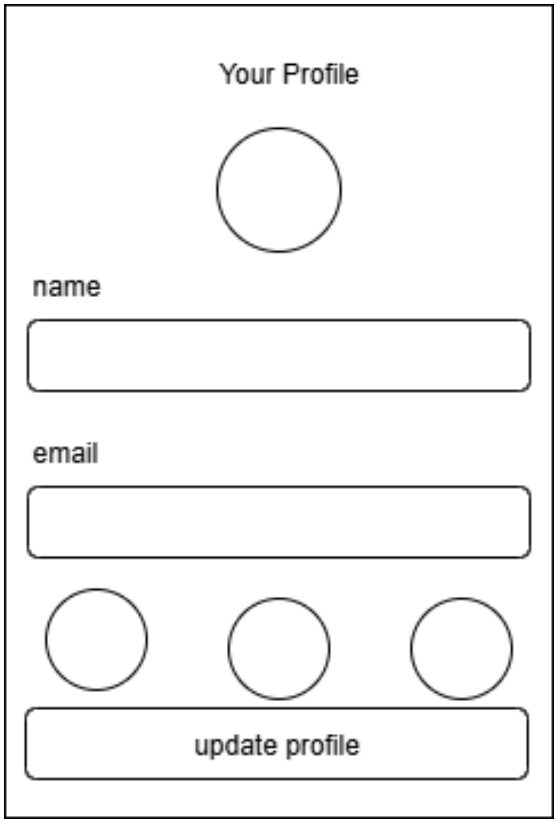


Figure 3.12 WireFraming Design of User Profile

Admin dashboard

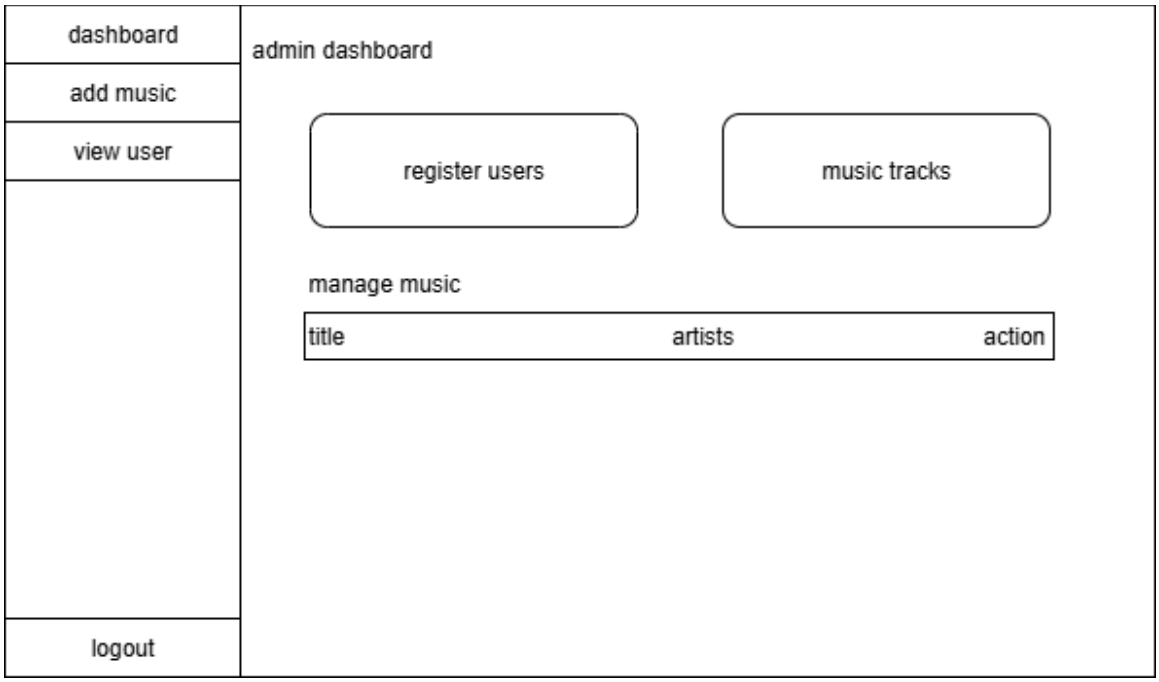


Figure 3.13 WireFraming Design of Admin Dashboard

Add music page

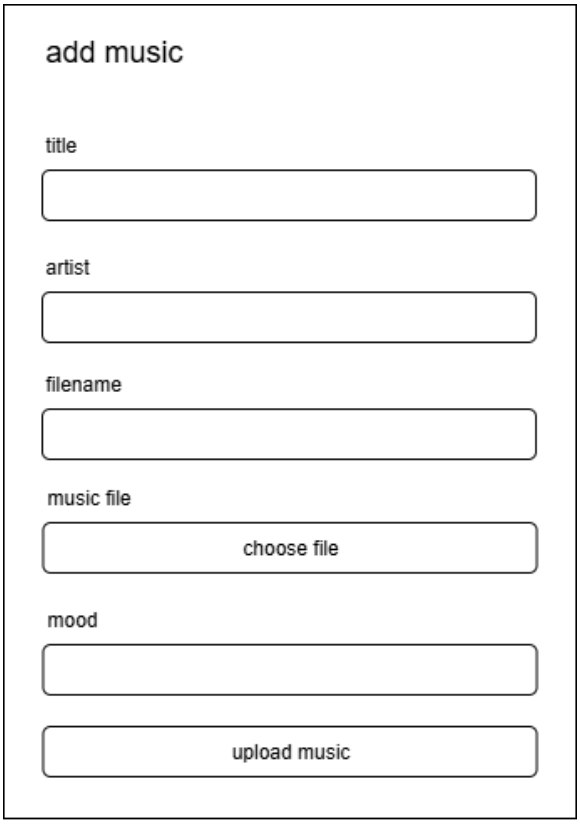


Figure 3.14 WireFraming Design of Add Music Page

Manage user page

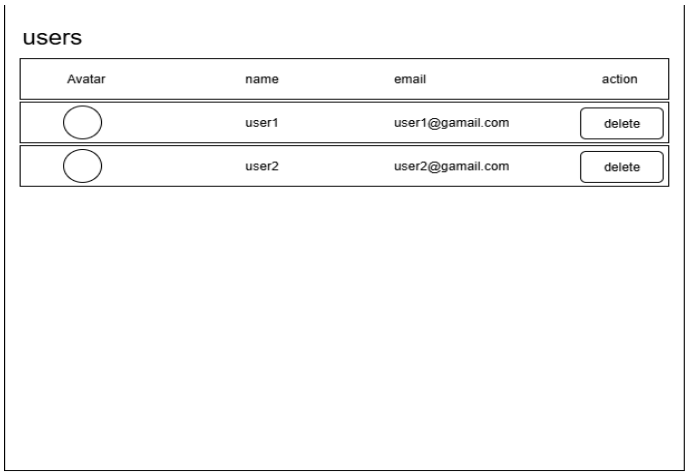


Figure 3.15 WireFraming Design of Manage User Page

3.2.5 Interface Design (UI Interface/Interface Structure Diagrams)

UI (User Interface) design is the process of developing visually appealing and user-friendly interfaces that enable people to easily interact with a system. It focuses on the layout, colors, buttons, icons, and overall structure to make navigating easy and intuitive. A strong user interface allows users to easily log in, play music, and update their profile. It assures uniformity across all pages, gives immediate feedback (such as loading indicators or problem notifications), and is well-suited to different devices. Overall, UI design seeks to improve the user experience by creating an interface that is simple, efficient, and fun to use.

User Dashboard

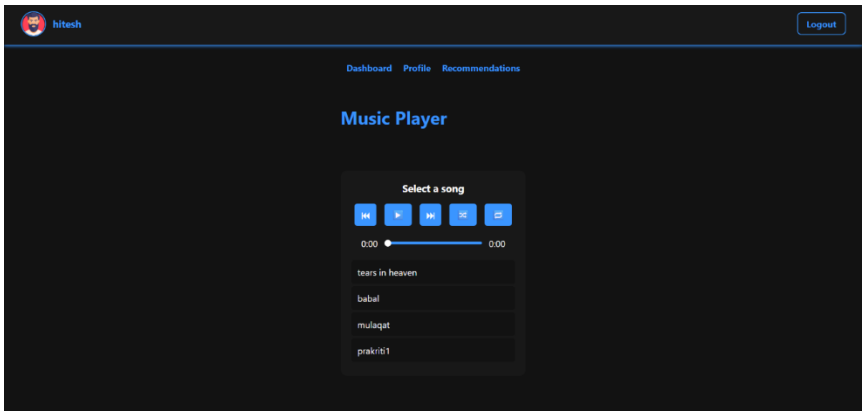


Figure 3.16 Interface Design of User Dashboard

Signup page

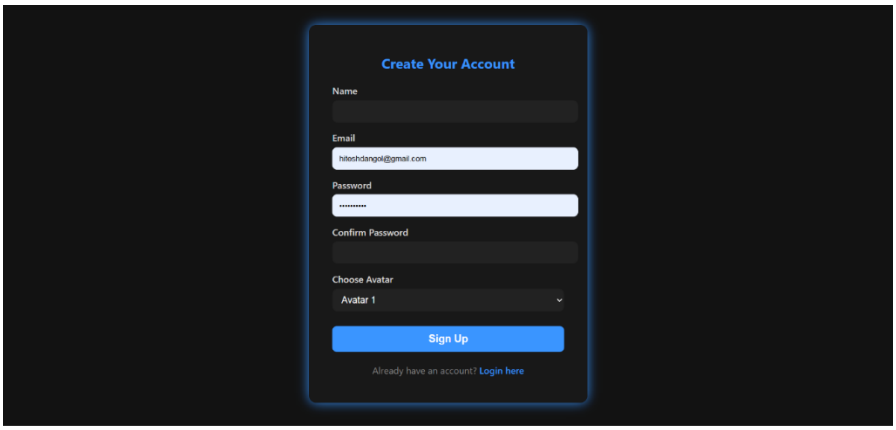


Figure 3.17 Interface Design of Signup Page

Login page

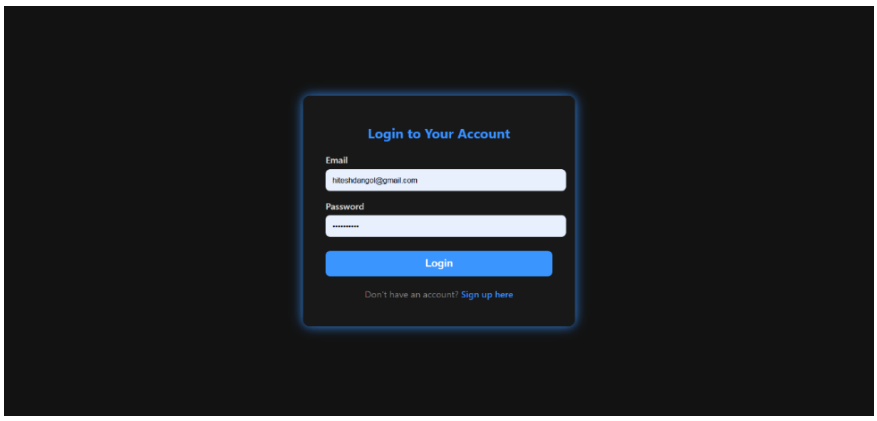


Figure 3.18 Interface Design of Login Page

User profile page

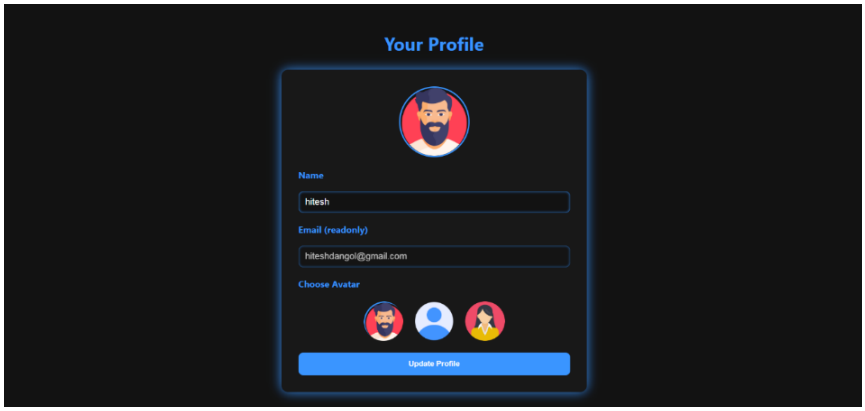


Figure 3.19 Interface Design of User Profile Page

Admin dashboard

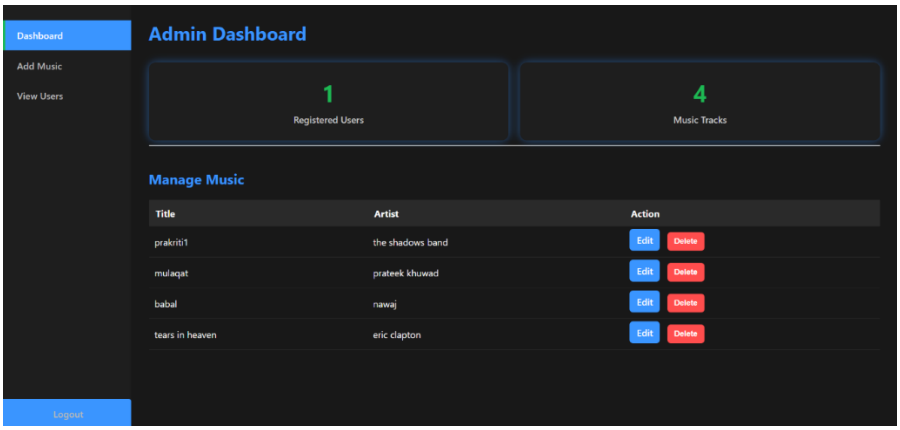


Figure 3.20 Interface Design of Admin Dashboard

Add music page

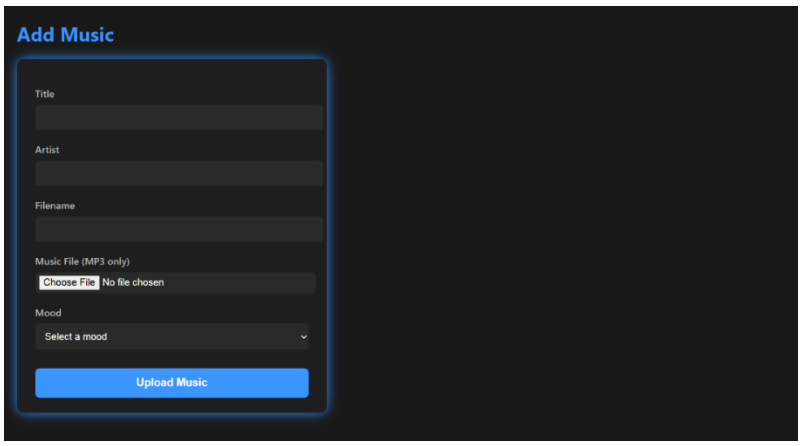


Figure 3.21 Interface Design of Add Music Page

Manage user page

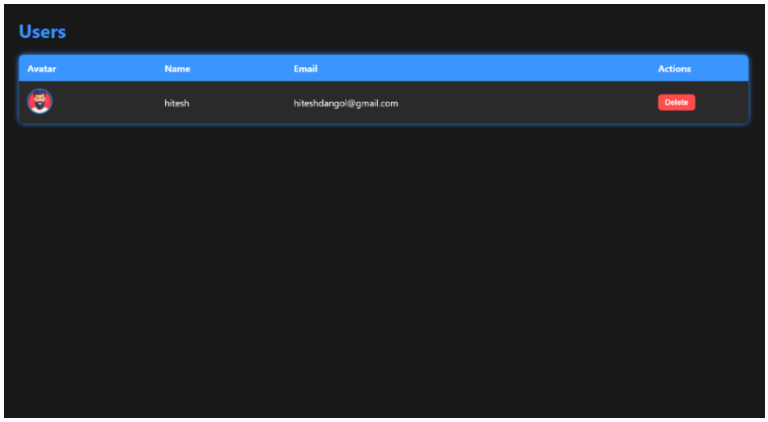


Figure 3.22 Interface Design of Manage User Page

3.3 Algorithm

Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is a powerful deep learning model designed Convolutional neural networks, or CNNs, are strong deep learning models made especially for visual vision analysis. By using a sequence of convolutional layers to automatically identify significant elements in pictures, it imitates the human visual brain. To capture local patterns like edges, corners, and textures, each convolutional layer applies a number of filters that move across the input picture. In order to provide non-linearity and allow the network to learn intricate patterns, these extracted features are then run through activation functions such as ReLU. The feature maps are then down sampled via pooling layers, which lower dimensionality while maintaining crucial information to enhance generalization and computational efficiency. The high-level characteristics that are produced after a number of convolution and pooling stages are compressed into a one-dimensional vector and then fed into fully connected layers that function similarly to a conventional neural network, learning to categorize the input into several groups in this example, various facial expressions. In order to choose the most likely emotional state, the output layer usually assigns probabilities to each emotion class using a soft max function. CNNs are very good at tasks like face emotion identification because of their spatial awareness and hierarchical feature learning, which allow systems to correctly read even the smallest changes in human emotions. The convolutional neural network is made of four main parts. Convolutional layers, Rectified Linear Unit (ReLU for short), Pooling layers, Fully connected layersThis section dives into the definition of each one of these components through the example of the following example of classification of a handwritten digit.

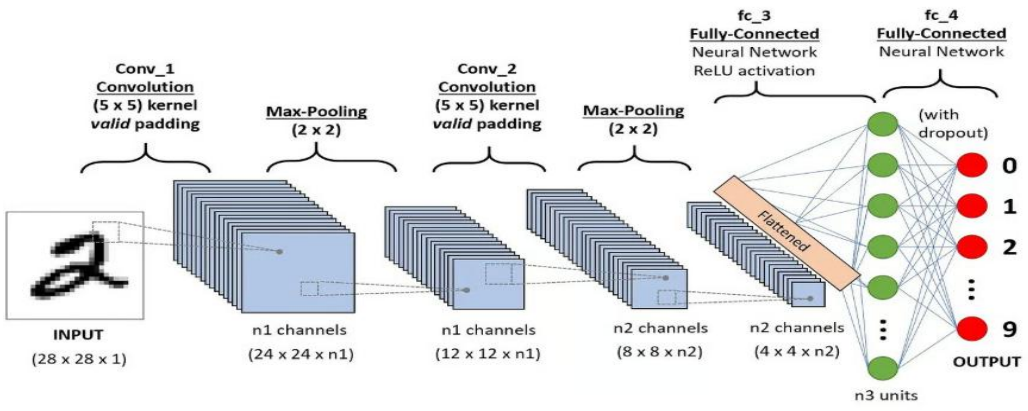


Figure 3.23 Architecture of the CNNs applied to digit recognition [12]

For a 2D input image I and a 2D kernel K, the convolution operation can be defined as:

$S(i,j)=(I*K)(i,j)=\sum_{m=0|M-1}\sum_{n=0|N-1} I(i+m,j+n)\cdot K(m,n)$

- $I(i,j)I(i,j)$: The pixel value at position $I(i,j)I(i,j)$ in the input image.
- $K(m,n)K(m,n)$: The weight of the kernel at position $(m,n)(m,n)$.
- $S(i,j)S(i,j)$: The output feature map at position $(i,j)(i,j)$. [12]

Chapter-4. Implementation and Testing

4.1 Implementation

4.1.1 Tools used (CASE tools, programming languages, database platforms)

Following are the tools and framework used for the accomplishment of this project:

Frontend

1. HTML (Hyper Text Markup Language)

In the EBMRS, **HTML** is used to create and structure the web interface. It organizes different sections such as headers, user prompts, music playlists, and controls using various tags and elements. HTML helps define the layout of the system's pages by including headings, paragraphs, buttons, links, and images, providing a clear and user-friendly experience for users interacting with the system

2. CSS (Cascading style sheet)

In EBMRS, css is used for designing different tags of html. It is also used to design different component by the help of class and id. Different css are used such as inline css, internal css, and external css to design this system. It is used for defining the styles for web pages. By using css, we can control the text color, font style, the spacing between paragraphs, sizing of columns, layout designs, and many more.

3. JavaScript

In EBMRS, JavaScript is used for client-side validation and to make dynamic, interactive and responsive web pages. It is used to add dynamic behavior to the webpage and add special effects to the webpage.

Backend

1. Python/PHP

In the Emotion-Based Music Recommendation System (EBMRS), Python and PHP are used for backend development and creating dynamic web pages. They handle server-side scripting tasks such as establishing connectivity with the database, encrypting sensitive data, validating user inputs, and managing user authentication to control access to specific

pages like login and dashboard. Additionally, both languages are responsible for performing database operations, including adding, updating, and deleting records to maintain accurate and up-to-date information.

Specifically, Python plays a critical role in implementing facial emotion detection. It uses advanced machine learning and computer vision library such as MediaPipe to analyze real-time facial expressions captured from the webcam. Python processes these images and classifies the detected emotion, which is then sent to the PHP side through a local API. Based on the emotion received, PHP dynamically generates music recommendations for the user.

Server

1. Apache Server

In this system, apache server is used to run php files and creating fast and dynamic web pages.

Database

1. MySQL

MySQL is use for storing all the information required to the database in EBMRS. It is used for performing CRUD operation such as create, delete and update data from the database as requested by the user.

Designing Tools:

1. Draw.io

This is used to generate diagrams for system analysis and design of EBMRS. Diagrams were created using this tool in order to save time since all components are available with drag and drop functions.

Documentation Tools:

MS Word

This is used for writing and editing the documentation of EBMRS.

MS PowerPoint

This is used for making PowerPoint slide of EBMRS.

4.1.2 Implementation details of modules

Different modules of this system are shown below:

Login module

```
if ($_SERVER["REQUEST_METHOD"] === "POST") {
    $email = $_POST["email"];
    $password = $_POST["password"];

    $stmt = $conn->prepare("SELECT id, password FROM users WHERE email = ?");
    $stmt->bind_param("s", $email);
    $stmt->execute();
    $stmt->store_result();

    if ($stmt->num_rows === 1) {
        $stmt->bind_result($id, $hashed);
        $stmt->fetch();
        if (password_verify($password, $hashed)) {
            $_SESSION["user_id"] = $id;
            header("Location: dashboard.php");
        }
    }
}
```

Listing 4.1 Code Snippet of Login Module

This code to check if a user's email and password are correct and log them in by starting a session

Signup module

```
if ($_SERVER["REQUEST_METHOD"] === "POST") {
    $name = $_POST["name"];
    $email = $_POST["email"];
    $password = password_hash($_POST["password"], PASSWORD_DEFAULT);

    $sql = "INSERT INTO users (name, email, password) VALUES (?, ?, ?)";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("sss", $name, $email, $password);
    $stmt->execute();

    header("Location: login.php");
    exit();
}
```

Listing 4.2 Code Snippet of Signup module

This code is used to register a new user by saving their name, email, and securely hashed password into the database.

Music player module

```
$result = $conn->query("SELECT * FROM music");
while ($row = $result->fetch_assoc()) {
    echo '<div>';
    echo '<strong>' . $row['title'] . '</strong> by ' . $row['artist'];
    echo '<audio controls>';
    echo '<source src="uploads/' . $row['filename'] . '" type="audio/mpeg">';
    echo '</audio>';
    echo '</div>';
}
```

Listing 4.3 Code Snipped of Music player module

This code retrieves music data from a database and displays it as an HTML list of playable audio tracks.

Add music module

```
// admin_upload.php
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $title = $_POST['title'];
    $artist = $_POST['artist'];
    $genre = $_POST['genre'];
    $file = $_FILES['music_file'];

    $filename = time() . "_" . basename($file['name']);
    $target = "uploads/" . $filename;

    if (move_uploaded_file($file['tmp_name'], $target)) {
        $sql = "INSERT INTO music (title, artist, genre, filename) VALUES (?, ?, ?, ?)";
        $stmt = $conn->prepare($sql);
        $stmt->bind_param("ssss", $title, $artist, $genre, $filename);
        $stmt->execute();
        echo "Music uploaded successfully!";
    } else {
        echo "Upload failed.";
    }
}
```

Listing 4.4 Code Snippet to Add Music Module

This code processes uploaded music files, saves them to a designated directory, and inserts their details (title, artist, genre, filename) into a database.

4.2 Testing

Software testing is an investigation conducted to provide stakeholders with the information about the quality of the product or services under test. It is the process of executing software with the intent of finding bugs and to ensure that it satisfies the specified requirements

After the development of the system, we have tested whether it meets the requirement or not in different phases. The functionality has also been tested taking some test cases.

4.2.1 Test Case for Unit testing

Unit testing in the Library Management System is a crucial step in ensuring the reliability and functionality of individual modules. It involves designing and executing test cases

that cover various scenarios, such as user authentication and note management. By conducting unit tests, we verify that each module performs as expected. Through comprehensive unit testing, we thoroughly examine the system's components, enabling us to identify and address any bugs or issues.

Table 4.1 User testing

S.N	Test Name	Input	Expected Output
1	Enter valid email and password	Email=hitesh@gmail.com and password=hitesh123	Enter in home page
2	Play music	Click to play button	Starts to play music
3	Next music	Click next button	Plays next button
4	Previous music	Click previous button	Plays previous music
5	Repeat and shuffle music	Click repeat and shuffle button	Plays one music repeatedly or shuffles all the music
6	Change user name	Pervious name = hitesh Changed name = Hitesh 1	Successfully changed
7	Change user profile pic	Pervious pp= avatar 1 changed pp = avatar 2	Successfully changed
8	Emotion based music recommendation system	Click to recommendation	Goes to recommendation
9	Recommend music	Click recommend music	Displays recommended music according to the emotion
10	Logout	Click to logout button	Logout

Admin Testing

Table 4.2 Admin testing

S.N	Test Name	Input	Expected Output

1	Enter valid email and password	Email=admin@admin.com and password=admin123	Enter in home page
2	manage music	Click to add music	Displays form to add music
3	Add music button	Click add music button	Fill all the valid instruction and music added
4	Manage user	Click to view user	Displays users list
5	Delete user	Click on delete button	User deleted
6	Edit music	Click to edit button on the dashboard and edit the field	Edited successfully
7	Delete music	Click on delete music button	Deleted successfully
8	Logout	Click on logout button	Logout goes to login page

4.2.2 Test Case for System testing

System testing plays a crucial role in ensuring the smooth integration and functionality of all modules and components within the Library Management System. This testing phase involves validating the system as a whole to ensure functional cooperation among its various parts and modules.

Table 4.3 System testing

S.N	**Test Case**	**Expected Result**	**Test Result**
1	On Click of Add music with all valid inputs	A New music was added	Successful
2	On clicking on Profile Button	User Profile Displayed	Successful
3	On clicking Delete button on user	User deleted	Successful

4	On clicking music recommender	Music recommended according to music	Successful

4.3 Result and Analysis

The Emotion-Based Music Recommendation System (EBMRS) aims to improve the user experience. Core features including user registration, login, music listening, and emotion-based song recommendations are all efficiently supported by the system. While administrators control users and music content via a specialized interface, users have the option to choose music directly or get recommendations depending on their mood. The system's modular architecture, which includes distinct parts for user management, emotion processing, music management, and login/signup, guarantees scalability, ease of maintenance, and obvious functionality. The system's capacity to customize music recommendations is one of its main advantages, as it can boost user engagement and happiness. However, the precision of emotion input or detection has a significant impact on the system's efficacy. The relevancy of music recommendations may decrease if emotions are not accurately identified. Although the fundamental user authentication system works, further security features that are essential for practical implementation are not depicted in the illustrations. All things considered, EBMRS effectively incorporates emotional context into the music selection procedure, resulting in a more tailored and significant listening experience. The system has great potential for wider adoption and effect with future improvements including real-time emotion recognition and enhanced security.

Chapter-5. Conclusion and Future Recommendation

5.1 Conclusion

To sum up, the Emotion-Based Music Recommendation System has the power to revolutionize how people listen to music by making it more responsive, easy to use and emotionally convincing. With ongoing developments in AI, machine learning, and user data analytics, this system has, the potential to be very important for applications related to mental health, stress reduction, and entertainment.

5.2 Lesson learnt/ Outcome

Each project promotes education and the growth of knowledge in a variety of fields. Strong problem-solving skills, independent thought, proper rule application and improved writing and communication talents have all been facilitated by this project.

- **Problem Solving Skills:**

One major challenge faced while working on the music recommendation system involved organizing and structuring the music database for easy access. Effective database management techniques were implemented to ensure quick retrieval of songs based on user emotions. Additionally, issues related to website responsiveness were addressed, and the search functionality was improved to enhance the overall user experience.

- **Writing Skills:**

This project enhanced the ability to create structured documentation, including system proposals, database schema designs, and workflow diagrams. It also improved content writing skills by developing user-friendly descriptions for music categories and crafting instructional guides to help users navigate the music recommendation website.

- **Time Management:**

Managing different aspects of the music recommendation system required careful planning. Prioritizing tasks like front-end design, music data entry, and testing ensured smooth project completion. Setting clear deadlines and dividing work efficiently led to the successful development of a functional and user-friendly platform.

5.3 Future Recommendations

The future recommendation for EBMRS are listed below:

- **Expand emotion category:**

At the moment, the system might rely on sentiments like happiness, sadness, or anger. Recommending music would be more accurate and tailored if the emotion categories were expanded to encompass more complex emotions, such as enthusiasm, anxiety, relaxation, or nostalgia. By doing this, the system's emotional intelligence would be improved and appropriate tunes would be more appropriately matched to users' moods.

- **Real-time continuous emotion detection**

Real-time continuous emotion recognition would enable the system to track emotional shifts while music is playing, as opposed to sensing a user's emotion just once each session. This would allow the playlist to be dynamically adjusted, making the listening experience more responsive and engaging as the system adjusts to users' changing moods.

- **Integration with music streaming platform like Spotify**

The system may access a wider music database, user playlists, and listening history by integrating with well-known services like Apple Music or Spotify. This would enhance the simplicity and accuracy of recommendations, enabling consumers to take advantage of tailored, emotion-based recommendations on the platforms they currently utilize.

- **Add other ways of emotion detection like voice tone**

Emotions can now be entered by facial expressions. Another layer of emotion identification can be added by including voice tone analysis, in which the system uses vocal signals or speech patterns to identify emotion. This would increase the system's adaptability and accessibility, particularly for visually impaired or voice-activated users.

References

[1] A. P. Rajesh Wadhvani, "A Comparative Study of Music Recommendation Systems," *2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS),* 2018.

[2] M. C. D. S. M. S. Sriraj Katkuri, "Emotion Based Music Recommendation System," *International Journal of Engineering Research & Technology (IJERT),* 2023.

[3] S. V. G. P. V. L. M. S. K. s. Tadi Lahari, "EMOTION BASED MUSIC RECOMMENDATION SYSTEM," *INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS),* 2024.

[4] M. M. S. L. R. R. V. A. Phaneendra, "AN EMOTION BASED MUSIC RECOMMENDATION SYSTEM," *International Research Journal of Modernization in Engineering Technology and Science,* 2022.

[5] N. D. M.Sree Vani, "EMOTION BASED MUSIC RECOMMENDATION SYSTEM," *INTERNATIONAL JOURNAL OF CREATIVE RESEARCH AND THOUGHTS,* 2022.

[6] ,. S. L. S. P. D. B. Shubham Kulkarni, "FACIAL EMOTION BASED MUSIC RECOMMENDATION SYSTEM," *INTERNATION JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT IJNRD,* 2023.

[7] ,. K. K. T. B. R. W. A. B. S. P. D. H. G. Yash Sahare, "Emotion Based Music Recommendation System," *International Journal of Research Publication and Reviews,* 2023.

[8] C. G. P. r. Dr.Sunil Bhutada, "EMOTION BASED MUSIC RECOMMENDATION SYSTEM," *Journal of Emerging Technologies and Innovative Research (JETIR).*

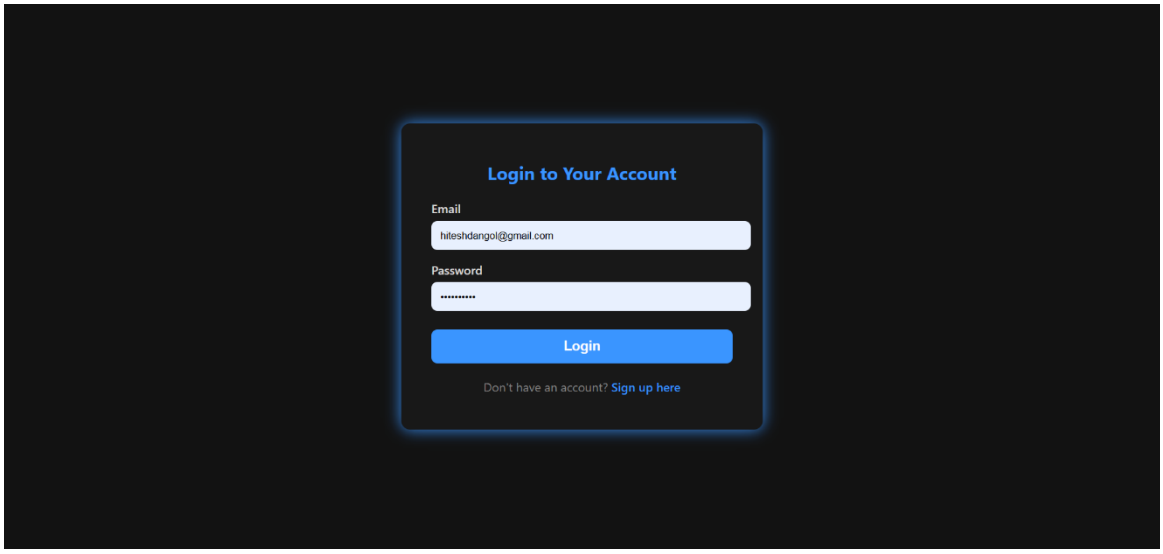
[9] S. U. O. M. A. N. A. P. Arti Bang, "Emotion Based Music Recommendation System," *Journal of Propulsion Technology,* 2024.

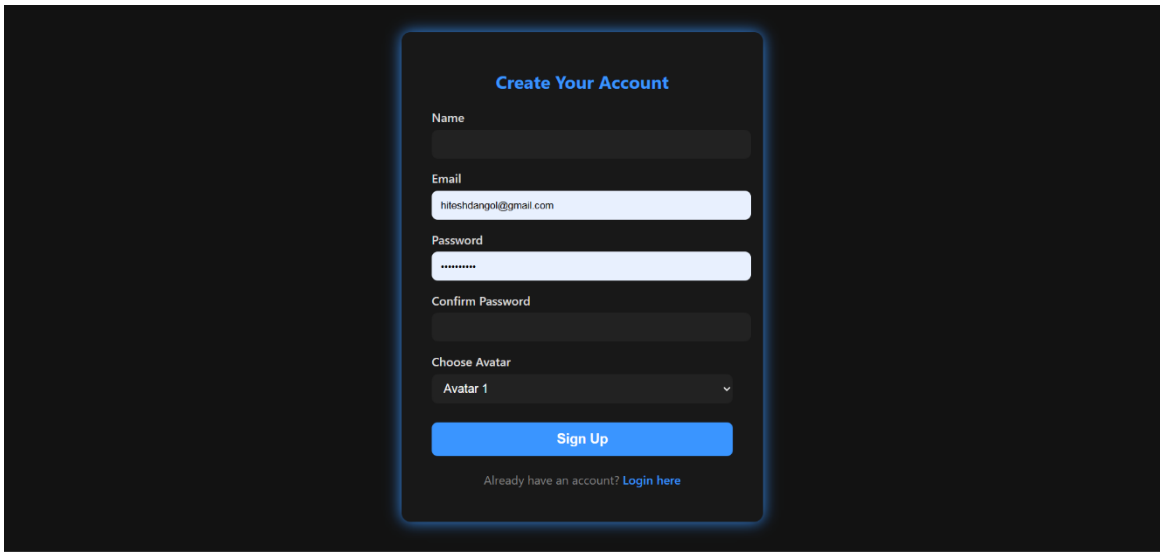
[10] I. S. P. K. A. S. M. A. M. C. I. o. T. C. Punidha A, "Human Emotion Detection using Machine Learning Techniques," *SSRN Electronic Journal,* 2020.

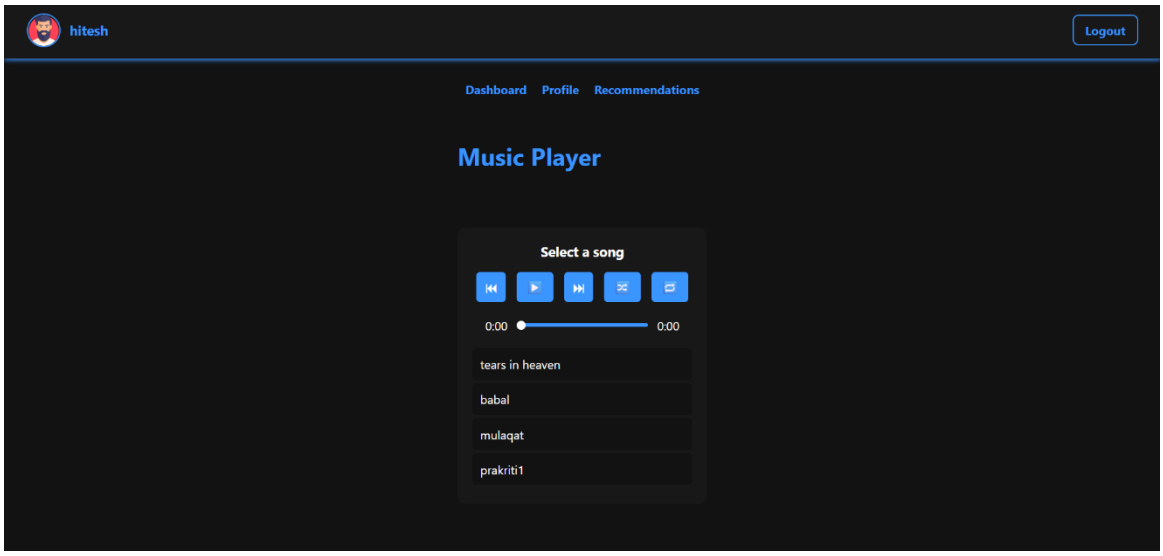
[11] C. T. Z. Y. W. Y. Li Qinjun*, "Facial Recognition Technology: A Comprehensive Overview," *Academic Journal of Computing & Information Science.*

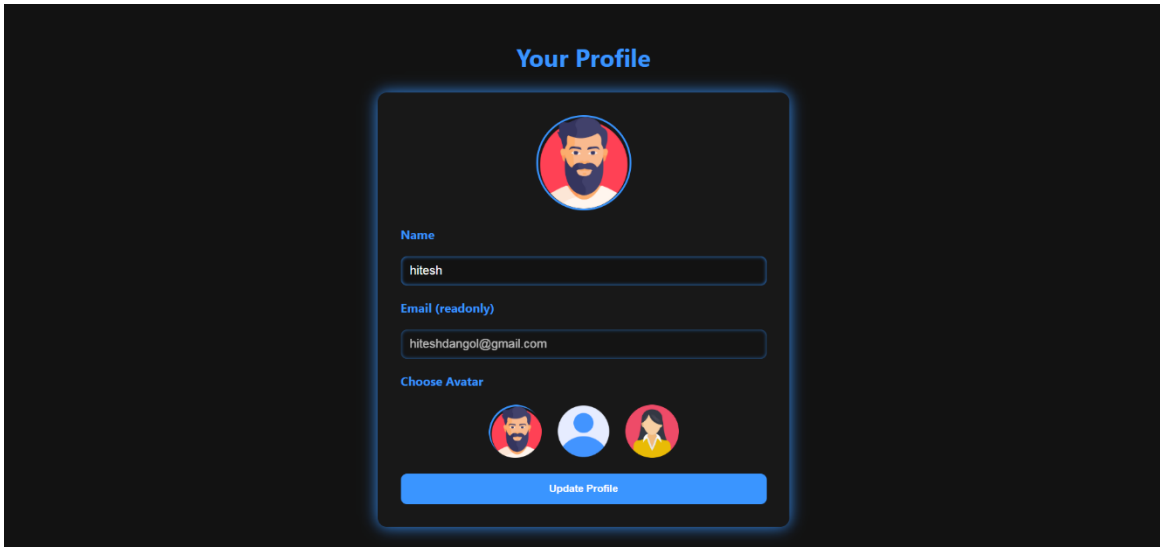
[12] Z. Keita, "Datacamp," 14 November 2023. [Online]. Available: https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns.

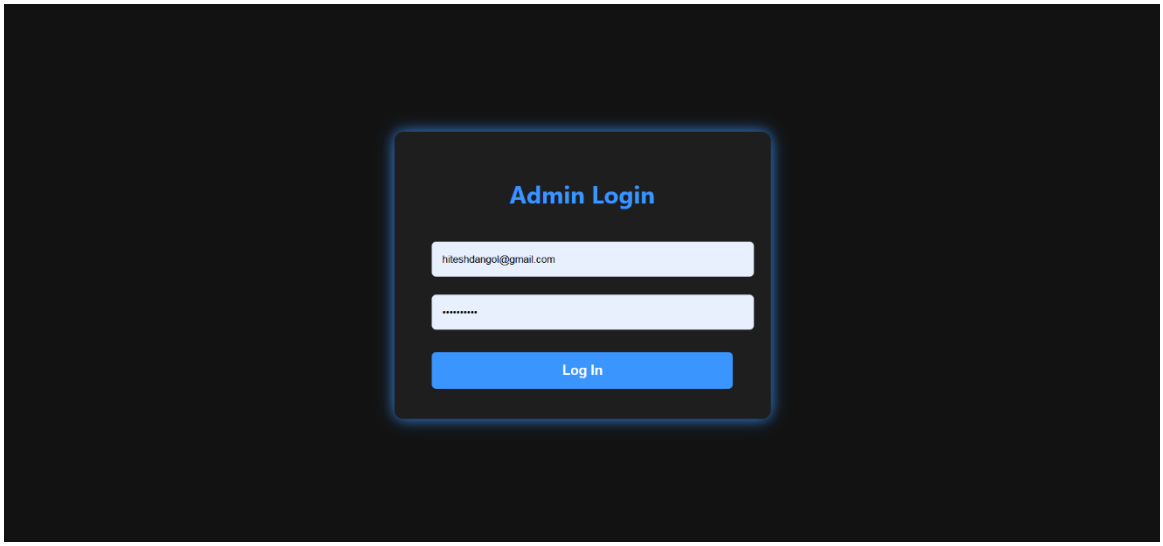
Appendices

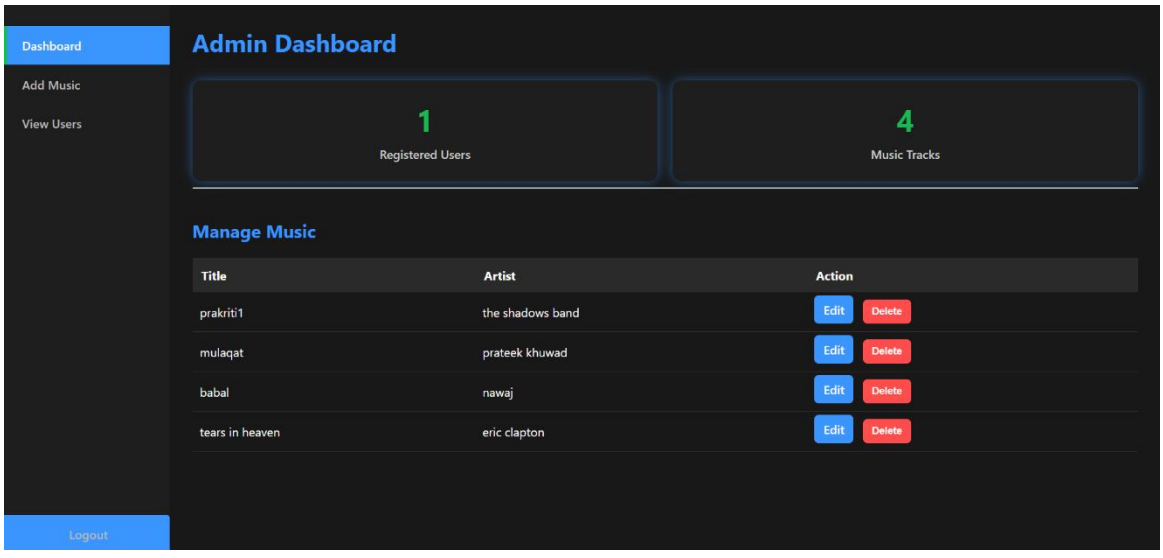

Login to Your Account
Email
hiteshdangol@gmail.com
Password
Login
Don't have an account? Sign up here


Create Your Account
Name
Email
hiteshdangol@gmail.com
Password
Confirm Password
Choose Avatar
Avatar 1
Sign Up
Already have an account? Login here


hitesh
Logout
Dashboard
Profile
Recommendations
Music Player
Select a song
0:00
0:00
tears in heaven
babal
mulaqat
prakriti1


Your Profile
Name
hitesh
Email (readonly)
hiteshdangol@gmail.com
Choose Avatar
Update Profile


Admin Login
hiteshdangol@gmail.com
Log In


Dashboard
Add Music
View Users
Logout
Admin Dashboard
1
Registered Users
4
Music Tracks
Manage Music
Title
Artist
Action
prakriti1
the shadows band
Edit
Delete
mulaqat
prateek khuwad
Edit
Delete
babal
nawaj
Edit
Delete
tears in heaven
eric clapton
Edit
Delete

Add Music

Title

Artist

Filename

Music File (MP3 only)

Choose File No file chosen

Mood

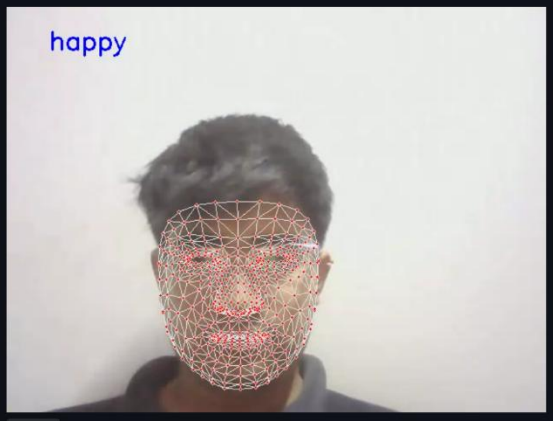
Select a mood

Upload Music

Users

Avatar	Name	Email	Actions
	hitesh	hiteshdangol@gmail.com	Delete

Emotion Based Music Recommender



STOP

Recommend me songs

Source Code

```
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}

require_once '../includes/db.php';

$userName = $_SESSION['user_name'];
$userAvatar = $_SESSION['user_avatar'];

// Fetch songs from DB
$songs = [];
$result = $conn->query("SELECT id, title, file_path FROM songs");
while ($row = $result->fetch_assoc()) {
    $songs[] = $row;
}
include '../includes/music-player.php';

?>

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<title>Dashboard</title>
<style>
  /* Reset */
  * {
    box-sizing: border-box;
  }

  body {
    margin: 0;
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background: #121212;
    color: white;
    display: flex;
    flex-direction: column;
    min-height: 100vh;
  }

  header {
    background-color: #181818;
    padding: 15px 30px;
    display: flex;
    align-items: center;
    justify-content: space-between;
    box-shadow: 0 2px 6px #3A95FF;
  }

  .user-info {
    display: flex;
    align-items: center;
    gap: 12px;
  }

  .user-info img.avatar {
    width: 45px;
    height: 45px;
    border-radius: 50%;
    border: 2px solid #3A95FF;
    object-fit: cover;
  }

  .user-info span {
    font-weight: 700;
    font-size: 18px;
    color: #3A95FF;
  }
```

```
a.logout {
  color: #3A95FF;
  font-weight: 700;
  text-decoration: none;
  border: 2px solid #3A95FF;
  padding: 8px 15px;
  border-radius: 8px;
  transition: background-color 0.3s ease;
}

a.logout:hover {
  background-color: #3A95FF;
  color: #121212;
}

main {
  flex-grow: 1;
  padding: 30px;
  display: flex;
  flex-direction: column;
  max-width: 900px;
  margin: auto;
  gap: 40px;
}

h1 {
  color: #3A95FF;
  margin: 0;
}

/* Music player container */
.player {
  background-color: #181818;
  padding: 20px;
  border-radius: 12px;
  box-shadow: 0 0 20px #3A95FF;
  display: flex;
  flex-direction: column;
  align-items: center;
}

.player-info {
  text-align: center;
}

.player-info h2, .player-info h3 {
  margin: 5px 0;
  font-weight: 700;
}

.player-controls {
  margin-top: 15px;
  display: flex;
  justify-content: center;
  gap: 25px;
}

button.control-btn {
  background: none;
  border: none;
  color: #3A95FF;
  font-size: 24px;
  cursor: pointer;
  transition: color 0.3s ease;
}

button.control-btn:hover {
  color: #6ab0ff;
}

/* Songs list */
.songs-list {
  background-color: #181818;
  border-radius: 12px;
  padding: 20px;
```

```
  box-shadow: 0 0 20px #3A95FF;
  max-height: 350px;
  overflow-y: auto;
 }

 .songs-list table {
  width: 100%;
  border-collapse: collapse;
 }

 .songs-list th, .songs-list td {
  padding: 12px 15px;
  text-align: left;
  border-bottom: 1px solid #333;
 }

 .songs-list th {
  color: #3A95FF;
 }

 .like-btn {
  background: none;
  border: none;
  color: #888;
  font-size: 20px;
  cursor: pointer;
  transition: color 0.3s ease;
 }

 .like-btn.liked {
  color: #ff4c4c;
 }

 /* Nav links */
 .nav-links {
  display: flex;
  gap: 20px;
  justify-content: center;
  margin-bottom: 15px;
 }

 .nav-links a {
  color: #3A95FF;
  font-weight: 700;
  text-decoration: none;
  border-bottom: 2px solid transparent;
  transition: border-color 0.3s ease;
 }

 .nav-links a:hover {
  border-bottom-color: #3A95FF;
 }

</style>
</head>
<body>

<header>
  <div class="user-info">
    <img src="../avatars/<?= htmlspecialchars($userAvatar) ?>" alt="User Avatar" class="avatar" />
    <span><?= htmlspecialchars($userName) ?></span>
  </div>
  <a href="logout.php" class="logout">Logout</a>
</header>

<main>
  <div class="nav-links">
    <a href="dashboard.php">Dashboard</a>
    <a href="profile.php">Profile</a>
    <a href="../recommendations/">Recommendations</a>
  </div>

  <h1>Music Player</h1>

  <div class="player">
```

```
  <div class="player-info">
    <h2 id="song-title">Select a song</h2>
    <h3 id="song-artist"></h3>
  </div>
  <audio id="audio-player" preload="metadata"></audio>

  <div class="player-controls">
    <button id="btn-shuffle" class="control-btn" title="Shuffle">&#128256;</button>
    <button id="btn-prev" class="control-btn" title="Previous">&#9664;&#9664;</button>
    <button id="btn-play" class="control-btn" title="Play/Pause">&#9654;</button>
    <button id="btn-next" class="control-btn" title="Next">&#9654;&#9654;</button>
    <button id="btn-repeat" class="control-btn" title="Repeat">&#128257;</button>
  </div>
 </div>

 <div class="songs-list">
  <table>
   <thead>
    <tr>
     <th>Title</th>
     <th>Artist</th>
     <th>Like</th>
     <th>Play</th>
    </tr>
   </thead>
   <tbody>
    <?php foreach ($songs as $song): ?>
     <tr data-id="<?= $song['id'] ?>" data-filename="<?= htmlspecialchars($song['filename']) ?>" data-title="<?=
htmlspecialchars($song['title']) ?>" data-artist="<?= htmlspecialchars($song['artist']) ?>">
      <td><?= htmlspecialchars($song['title']) ?></td>
      <td><?= htmlspecialchars($song['artist']) ?></td>
      <td>
       <button class="like-btn" title="Like">&#10084;</button>
      </td>
      <td>
       <button class="control-btn play-song" title="Play">&#9654;</button>
      </td>
     </tr>
    <?php endforeach; ?>
   </tbody>
  </table>
 </div>
</main>

<script>
 // JavaScript for player functionality

 const audio = document.getElementById('audio-player');
 const songTitle = document.getElementById('song-title');
 const songArtist = document.getElementById('song-artist');
 const playBtn = document.getElementById('btn-play');
 const nextBtn = document.getElementById('btn-next');
 const prevBtn = document.getElementById('btn-prev');
 const shuffleBtn = document.getElementById('btn-shuffle');
 const repeatBtn = document.getElementById('btn-repeat');

 let songs = [...document.querySelectorAll('.songs-list tbody tr')];
 let currentIndex = -1;
 let isPlaying = false;
 let isShuffle = false;
 let isRepeat = false;

 function loadSong(index) {
  if(index < 0 || index >= songs.length) return;
  const song = songs[index];
  const filename = song.dataset.filename;
  const title = song.dataset.title;
  const artist = song.dataset.artist;
  audio.src = '../music/' + filename;
  songTitle.textContent = title;
  songArtist.textContent = artist;
  currentIndex = index;
 }

 function playSong() {
```

```
  if(currentIndex === -1) return;
  audio.play();
  isPlaying = true;
  playBtn.innerHTML = '&#10074;&#10074;'; // pause icon
}

function pauseSong() {
  audio.pause();
  isPlaying = false;
  playBtn.innerHTML = '&#9654;'; // play icon
}

playBtn.addEventListener
```