## PROBLEM BASED ASSIGNMENT IV YEAR / I SEMESTER

ACADEMIC YEAR: 2024 -25

### Group Number-8

**Music connect**

**The only site for your music**

**By**

24QM5A6602 K.Sai Ram Yadav

24QM5A6603 K.Vishnu Teja

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (AI&ML) KG REDDY COLLEGE OF ENGINEERING AND TECHNOLOGY**

**Chilkur Village, Moinabad, Hyderabad, Telangana 500075 NOVEMBER 2024**

# CONTENT

* Front Page
* Abbreviation
* Problem Statement
* Purpose/Need of the Problem
* Hardware & Software Requirements
* Application of Technology
* Coding
* Results (Output Screenshots)
* Conclusion
* References

# ABBREVIATION

* **PBA:** Project-Based Assessment
* **DL:** deep Learning
* **Pred:** predicted
* **SVM:** Support Vector Machine
* **LR:** Logistic Regression
* **GB:** Gradient Boosting
* **KNN:** k-Nearest Neighbors
* **ML:** Machine Learning
* **IDE:** Integrated Development Environment
* **CPU:** Central Processing Unit
* **RAM:** Random Access Memory
* **SSD:** Solid State Drive
* **API:** Application Programming Interface
* **ROC:** Receiver Operating Characteristic
* **AUC:** Area Under the Curve
* **CV:** Cross-Validation
* **TPU:** Tensor Processing Unit
* **GPU:** Graphics Processing Unit

### PROBLEM STATEMENT

This website serves as a unique platform for individuals with skills in singing and playing various musical instruments to connect, collaborate, and showcase their talents. The primary purpose of the website is to create a supportive community where musicians and singers can discover each other and form groups or bands, allowing their collective creativity to flourish.

**Key Features:**

1. **Talent Discovery and Networking**:
   * The website enables users to create profiles showcasing their skills, such as singing or playing specific musical instruments.
   * Users can search for other talented individuals based on their location, skillset, or genre preferences, fostering connections and collaborations.
2. **Event Updates**:
   * A dedicated section for upcoming events, concerts, auditions, and workshops ensures that users are always informed about opportunities to perform or improve their skills.
   * Regular updates keep users engaged and encourage active participation in the music community.
3. **Encouragement for Hidden Talent**:
   * Many talented individuals hesitate to showcase their abilities due to a lack of encouragement or a proper platform. This website aims to bridge that gap by creating a positive and inclusive environment for all users.
   * By offering a space for feedback, mentorship, and recognition, the website inspires users to pursue their passions.
4. **Building a Creative Environment**:
   * The platform fosters an environment of growth and creativity, which is essential for nurturing emerging talent.
   * It provides resources and tools to help users collaborate, such as chat features, virtual jam sessions, and shared project spaces.
5. **Recruitment Opportunities**:
   * The website acts as a bridge between musicians and those seeking talent, such as event organizers, producers, or band managers.
   * Employers can browse profiles, view portfolios, and contact musicians or singers directly through the platform, creating valuable opportunities for users.
6. **Potential for Further Development**:
   * Future enhancements could include advanced features such as live-streaming performances, AI-based matching for collaborations, and personalized recommendations for gigs or training sessions.
   * By expanding its user base and functionality, the platform could become a central hub for the global music community.

# PURPOSE/NEED OF THE PROBLEM

**MusicConnect: Simplifying Life for Musicians**

MusicConnect is a revolutionary platform designed to address the common struggles faced by musicians. For those who are tired of the challenges involved in finding suitable bandmates, booking gigs, or building meaningful connections within their local music community, MusicConnect provides a dedicated and tailored solution. Unlike general social media platforms, which offer limited and unfocused tools for musicians, MusicConnect is purpose-built to cater specifically to the needs of the music community.

**Features for Musicians:**

1. **Custom Profiles for Showcasing Talent**:  
   On MusicConnect, every musician gets their own personalized profile. This profile acts as a digital resume where users can showcase their expertise. Whether you are a vocalist, a guitarist, a drummer, or a pianist, you can highlight your skills, your favorite songs or background music (BGM), and the instruments you love to play. This helps potential collaborators or recruiters understand your style and preferences at a glance.
2. **Focused Networking**:  
   Unlike conventional social media platforms, which are not specifically designed for musicians, MusicConnect provides an exclusive space where musicians can connect, collaborate, and grow together. The platform eliminates distractions and noise, allowing users to focus solely on building relationships and finding opportunities within the music industry.

**Support for Event Organizers:**

MusicConnect is not only beneficial for individual musicians but also serves as a valuable resource for event organizers. Organizers looking for talent—be it a solo performer, a duo, or an entire band—can browse the platform’s extensive database of musicians. Instead of relying on time-consuming searches or word-of-mouth recommendations, they can directly visit MusicConnect to find and recruit artists for concerts, shows, corporate events, weddings, and other performances.

**Why MusicConnect is Unique:**

The platform bridges the gap between talent and opportunity, creating a thriving ecosystem for the music industry. By focusing entirely on the needs of musicians and those who work with them, MusicConnect simplifies the process of finding collaborators and gigs, helping artists focus on what they do best—creating music.

# HARDWARE & SOFTWARE REQUIREMENTS

##### Hardware Requirements

1. **CPU:**

**Type:** Intel Core i5 or higher (or equivalent AMD processors

**Recommendation:** A multi-core processor is preferable to efficiently handle data processing and model training tasks.

##### RAM:

**Minimum:** 8 GB

**Recommended:** 16 GB or higher

**Justification:** Sufficient RAM is essential for running data-intensive applications, especially during the training of machine learning models.

##### Storage:

**Minimum:** 500 GB SSD

**Recommendation:** Use SSDs for faster read/write speeds which can significantly improve performance, especially when handling large datasets.

##### Graphics Card (Optional):

**Type:** NVIDIA GeForce GTX 1060 or higher (or equivalent)

**Justification:** A dedicated GPU can accelerate model training, particularly for complex models like neural networks.

##### Software Requirements

1. **Operating System:**

**Type:** Windows, macOS, or Linux (Ubuntu recommended)

**Justification:** The operating system should support Python and the necessary libraries without compatibility issues.

##### Programming Language:

**Language:** Python 3.9 or higher

**Justification:** Python is widely used in data science and machine learning due to its simplicity and the availability of numerous libraries.

##### Libraries and Frameworks:

**Numpy:** For numerical computations and handling large datasets.

**Pandas:** For data manipulation and analysis, particularly for structured data.

**Scikit-learn:** For implementing various machine learning algorithms and model evaluation.

**Matplotlib & Seaborn:** For data visualization to better understand the dataset and results.

##### Google Generative AI SDK:

**Package:** google-generativeai

**Justification:** To utilize the Gemini API for generating personalized lifestyle suggestions based on the user's health data.

##### Integrated Development Environment (IDE):

**IDE Options:**

**Jupyter Notebook:** Ideal for interactive data analysis and visualization.

**Google Colab:** A cloud-based option that allows for collaborative work and easy access to GPU resources.

* + **PyCharm:** A powerful IDE for larger projects that require advanced features.

##### Data Management Tools:

**Database System:**

**Type:** SQLite or PostgreSQL for managing user data and predictions if needed.

**Justification:** Efficient for handling relational data and offers robust querying capabilities.

### APPLICATION OF TECHNOLOGY

Here are the key applications of MusicConnect:

1. Musical Collaboration
   * Band Formation: Musicians can find members to form new bands
   * Temporary Replacements: Find substitute players for gigs or recordings
   * Project Collaboration: Connect for specific musical projects or sessions
   * Skill Exchange: Musicians can teach or learn from each other
2. Event Management
   * Concert Promotion: Advertise upcoming performances
   * Venue Booking: Connect venues with performing artists
   * Festival Organization: Coordinate multiple artists for events
   * Workshop Announcements: Promote music workshops and classes
3. Community Building
   * Local Scene Development: Strengthen local music communities
   * Genre-Specific Groups: Connect musicians within same genres
   * Mentorship: Experienced musicians can guide newcomers
   * Jam Session Organization: Arrange casual playing sessions
4. Professional Networking
   * Career Opportunities: Find paid gig opportunities
   * Session Musicians: Connect with artists needing studio musicians
   * Industry Contacts: Network with producers, managers, and venues
   * Portfolio Showcase: Display musical achievements and skills
5. Educational Purposes
   * Music Schools: Connect students with teachers
   * Practice Partners: Find others to practice with
   * Skill Development: Share and learn new techniques
   * Workshop Organization: Arrange learning sessions
6. Equipment Sharing/Exchange
   * Gear Recommendations: Share equipment experiences
   * Studio Access: Connect with recording facilities
   * Equipment Borrowing: Share or borrow instruments
   * Practice Space Finding: Locate rehearsal spaces
7. Performance Opportunities
   * Open Mic Discovery: Find venues hosting open mics
   * Competition Information: Share details about music competitions
   * Audition Notices: Post and find audition opportunities
   * Busking Locations: Share information about street performance spots
8. Music Business Development
   * Band Promotion: Platform for promoting music groups
   * Audience Building: Connect with potential fans
   * Merchandise Sales: Promote band merchandise
   * Booking Management: Handle performance scheduling
9. Cultural Exchange
   * Cross-Genre Collaboration: Mix different musical styles
   * Cultural Events: Organize multicultural music events
   * International Connections: Connect with musicians globally
   * Traditional Music Preservation: Share cultural music knowledge
10. Support Services
    * Technical Support: Find sound engineers, technicians
    * Music Production: Connect with producers and studios
    * Marketing Services: Find promotion and marketing help
    * Legal Advice: Share music industry legal resources

# CODING

The following algorithms are used for prediction:

* User Authentication
* Data Management
* Content Display
* Error Management
* Session Control

##### Whole Code:

# # Import what we need from flask

# from flask import Flask, render\_template, request, redirect, url\_for, flash, session

# from datetime import datetime

# import json

# from werkzeug.security import generate\_password\_hash, check\_password\_hash

# # Create the Flask app

# app = Flask(\_\_name\_\_)

# # We need this for login to work

# app.secret\_key = 'your\_secret\_key'

# # Store user data (like a simple database)

# users = {}

# events = []

# band\_requests = []

# # Home page

# @app.route('/')

# def home():

# return render\_template('index.html')

# # Login page

# @app.route('/login', methods=['GET', 'POST'])

# def login():

# # If user submits the form

# if request.method == 'POST':

# # Get username and password from form

# username = request.form['username']

# password = request.form['password']

# 

# # Check if user exists and password is correct

# if username in users and check\_password\_hash(users[username]['password'], password):

# # Save username in session (remember who is logged in)

# session['username'] = username

# # Show success message

# flash('Login successful!')

# # Go to profile page

# return redirect(url\_for('profile', username=username))

# # If login fails, show error

# flash('Invalid username or password!')

# # Show login page

# return render\_template('login.html')

# # Logout

# @app.route('/logout')

# def logout():

# # Remove username from session (forget who was logged in)

# session.pop('username', None)

# # Show logout message

# flash('You have been logged out.')

# # Go back to home page

# return redirect(url\_for('home'))

# # Profile page

# @app.route('/profile/<username>')

# def profile(username):

# # Check if user exists

# if username not in users:

# flash('User not found!')

# return redirect(url\_for('home'))

# # Show profile page

# return render\_template('profile.html', user=users[username])

# # Register page

# @app.route('/register', methods=['GET', 'POST'])

# def register():

# # If user submits the form

# if request.method == 'POST':

# # Get form data

# username = request.form['username']

# password = request.form['password']

# 

# # Check if username already exists

# if username in users:

# flash('Username already exists!')

# return render\_template('register.html')

# 

# # Save new user data

# hashed\_password = generate\_password\_hash(password)

# users[username] = {

# 'username': username,

# 'password': hashed\_password,

# 'instruments': request.form.get('instruments', ''),

# 'experience': request.form.get('experience', ''),

# 'genre': request.form.get('genre', ''),

# 'bio': request.form.get('bio', '')

# }

# 

# # Show success message

# flash('Registration successful! Please login.')

# # Go to login page

# return redirect(url\_for('login'))

# 

# # Show register page

# return render\_template('register.html')

# # Events page

# @app.route('/events', methods=['GET', 'POST'])

# def events\_page():

# # If user submits new event

# if request.method == 'POST':

# # Get event details

# event = {

# 'title': request.form['title'],

# 'date': request.form['date'],

# 'location': request.form['location'],

# 'description': request.form['description']

# }

# # Add event to list

# events.append(event)

# flash('Event added successfully!')

# 

# # Show events page with list of events

# return render\_template('events.html', events=events)

# # Band requests page

# @app.route('/band-requests', methods=['GET', 'POST'])

# def band\_requests\_page():

# # If user submits new request

# if request.method == 'POST':

# # Get request details

# request\_data = {

# 'band\_name': request.form['band\_name'],

# 'instrument\_needed': request.form['instrument\_needed'],

# 'genre': request.form['genre'],

# 'description': request.form['description'],

# 'contact': request.form['contact']

# }

# # Add request to list

# band\_requests.append(request\_data)

# flash('Band request added successfully!')

# 

# # Show band requests page with list of requests

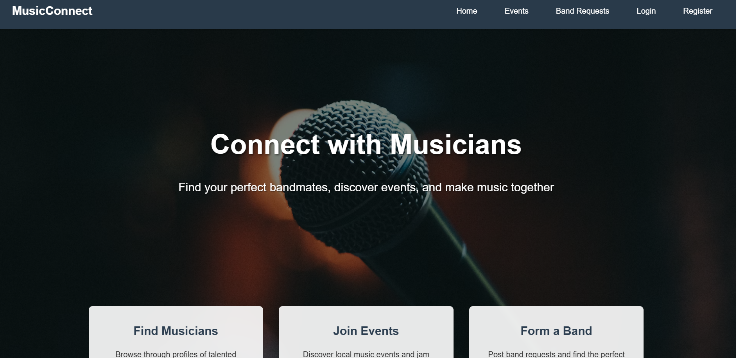
# return render\_template('band\_requests.html', requests=band\_requests)

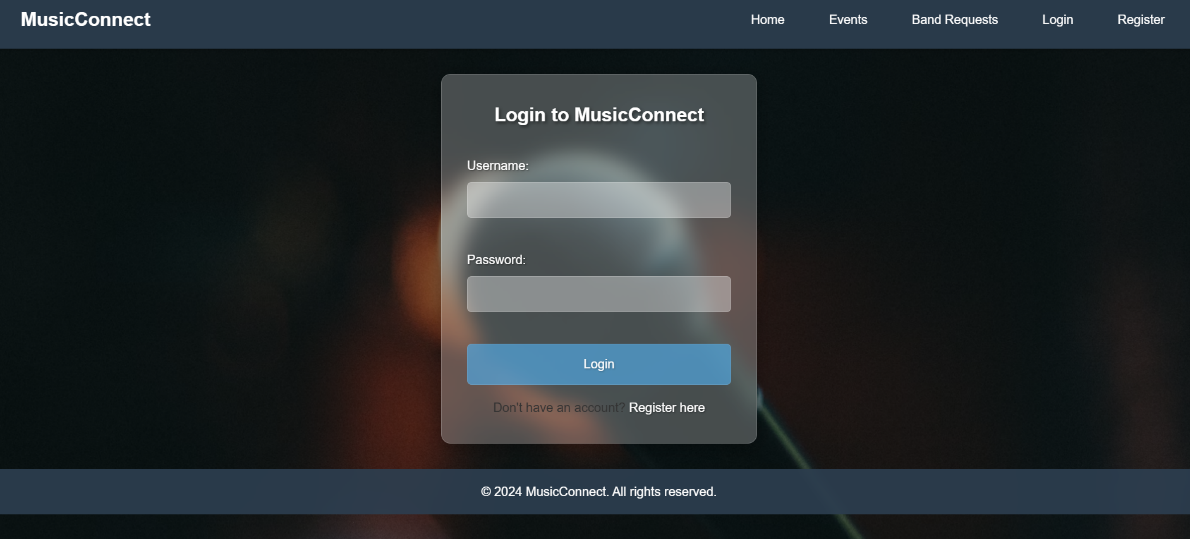
# # Start the website

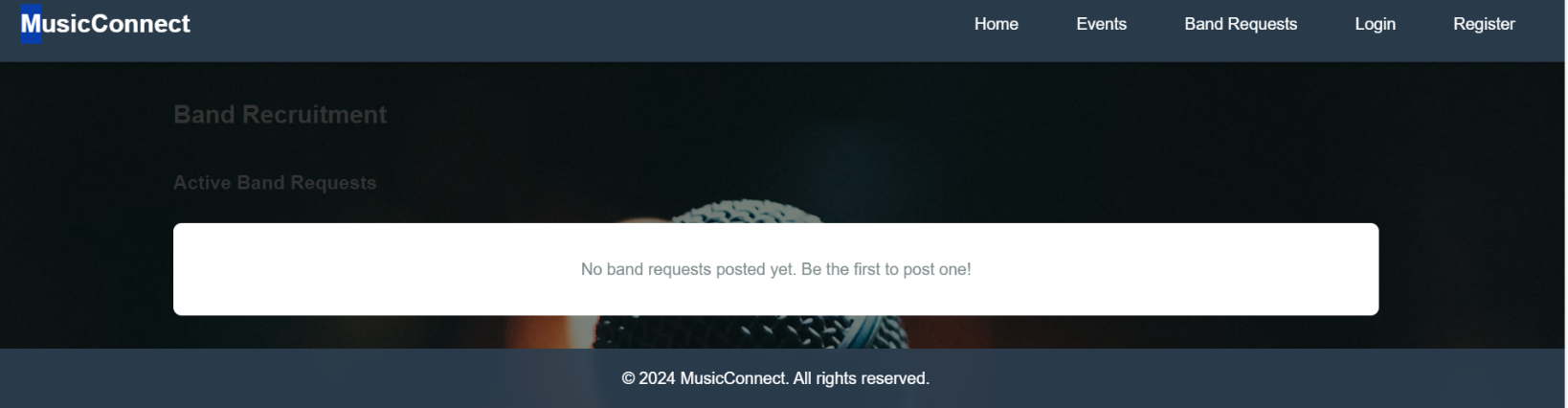
# if \_\_name\_\_ == '\_\_main\_\_':

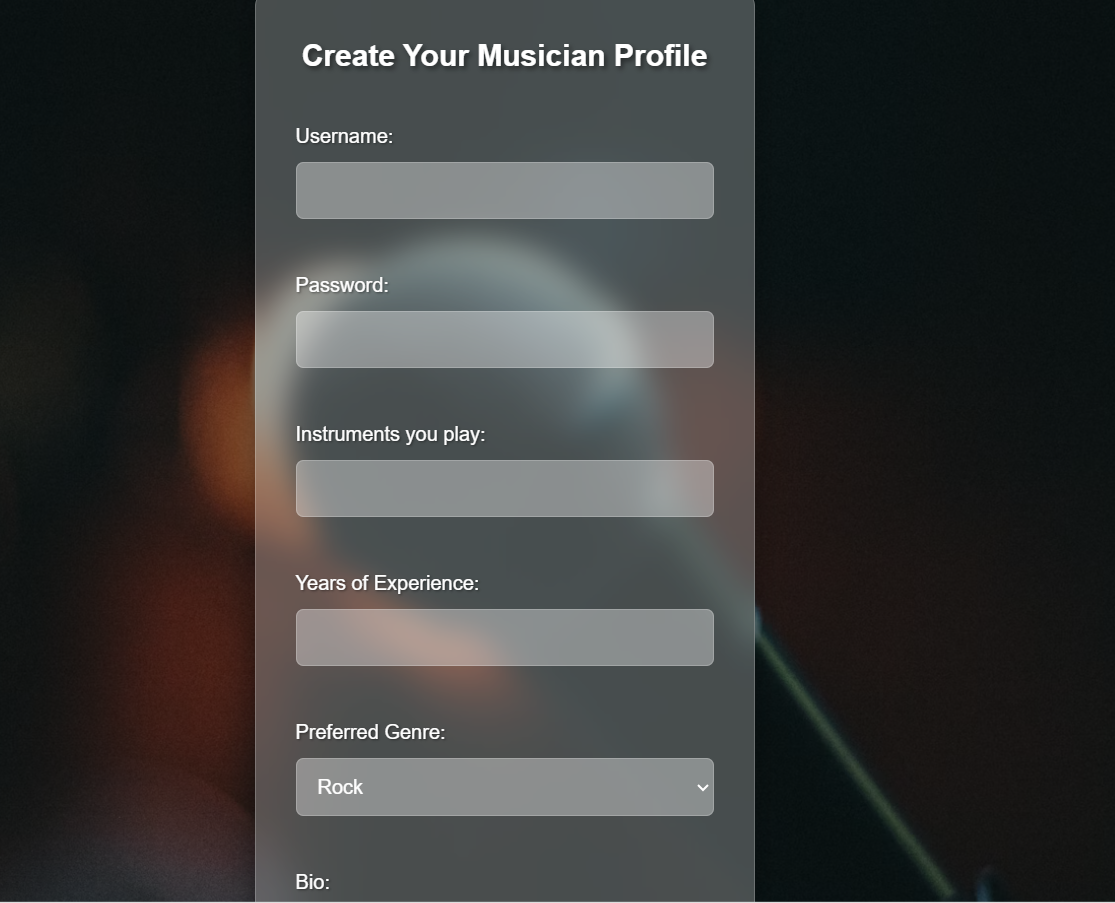
# app.run(debug=True)

# RESULT









# CONCLUSION

We developed a website tailored specifically for musicians to connect, collaborate, and form bands, offering a centralized platform to enhance their creative opportunities. Beyond just musicians, this platform is also designed for event managers or organizers, enabling them to easily discover and hire bands to perform at their shows or events. By bridging this gap between artists and event organizers, the website creates a wealth of opportunities for musicians to showcase their talents and expand their reach.

The platform is built using a combination of **HTML** for its structure and design and **Python** for backend functionalities, ensuring both ease of use and robust performance. Its user-friendly interface includes a straightforward registration process for anyone interested in joining. Registered users can create detailed profiles to showcase their skills, genres, and experience levels, making it easier for others to find suitable collaborators or performers.

Additionally, the website features a section dedicated to **upcoming shows and events**, providing users with real-time updates on performance opportunities or events in their area. This functionality not only helps musicians find gigs but also strengthens the local music community by promoting live music and fostering connections.

By simplifying the process of forming bands and booking performances, this website acts as a vital tool for empowering musicians, event organizers, and the broader music scene.

## REFERENCES

**Vampr:-<https://www.vampr.me/>**

**MusicianFinder:-<https://musicianfinder.com/>**

**Soundbetter:-**[**https://soundbetter.com/**](https://soundbetter.com/)