A Project Report on

Emotion Based Music Recommendation System

Submitted to

KL Deemed to be UNIVERSITY.

(Off Campus, Hyderabad)

(In partial fulfillment of the requirements for the award of bachelor degree)

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

By

Jonnalagadda krishna sai Roshith-2010030283

P.Siddharth-2010030477

P.Hitesh-2010030510

D.Harshith-2010030515

K.Mariyam-2010030518

Under the esteemed guidance of
Dr. Shadab Siddiqui
Assistant Professor Department of CSE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
KL Deemed to be UNIVERSITY
(Off Campus, Hyderabad) Estd. u/s 3 of UGC Act 1956
Campus: RVS NAGAR, Aziz nagar (PO), Moinabad Road, Hyderabad , R.R. Dist- 500075.

Batch: 2020-24

CERTIFICATE

This is to certify that the project entitled **"Emotion Based Music Recommendation System"** is being submitted by of **Jonnalagadda krishna Sai Roshith-(2010030283) , P. Hitesh-(2010030510) , P.Siddharth-(2010030477) , D. Harshith-(20100030515) , K. Mariyam-(2010030518)** B.Tech in partial fulfillment of the requirement for the award of the degree in **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING**, to the KL University, Hyderabad, is a record of the bonafide work carried out by them under my guidance and supervision. The results embodied in this project have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr. Shadab Siddiqui	**Arpita Gupta**
INTERNAL GUIDE	**HEAD OF THE DEPARTME**

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

First and foremost, I offer my sincere gratitude to my internal guide **Dr.Shadab Siddiqui** , Assistant Professor of Computer Science and Engineering who has supported me throughout this project with her/his patience and valuable suggestions.

I would like to express my gratitude to **Dr. Arpita Gupta** Head of the Department of Computer Science and Engineering for his support and valuable suggestions during the project work.

I am also grateful to the Principal **Dr. A. RamaKrishna** for providing me with all the resources in the college to make my project a success. I thank him for his valuable suggestions at the time of seminars which encouraged me to give my best in the project.

I would also like to thank all the supporting staff of the Dept. of CSE and all other departments who have been helpful directly or indirectly in making the project a success.

I am extremely grateful to my parents for their blessings and prayers for my completion of project that gave me strength to do my project.

.

Jonnalagadda krishna sai Roshith-2010030283

P.Siddharth-2010030477

P.Hitesh-2010030510

D.Harshith-2010030515

K.Mariyam-2010030518

ABSTRACT

Music has a profound ability to influence and reflect our emotional states. This Python program aims to leverage that connection by recommending songs tailored to a user's current emotional experience. The core functionality relies on mapping user-provided emotion keywords to acoustic properties and metadata patterns found in songs that typify those emotional qualities. Upon launching the program, the user is prompted to enter one or more emotion words describing their current feeling (e.g. happy, sad, angry, calm). These keywords are cross-referenced against a database of song data containing acoustic analysis from audio processing along with lyrical and metadata details like genre, tempo, key, and more. Using a weighted scoring system, the program identifies songs that most closely match the acoustic/lyrical fingerprint for the provided emotions based on data aggregated from scientific studies on music and emotion. The highest scoring songs are returned as a personalized playlist recommendation for the user to listen to songs that complement and validate their current emotional state.

Additional features include the ability to limit recommendations by criteria like genre, popularity, audio quality, and time period. The program can also generate data visualizations to explore similarities among songs and the emotion profiles they express. With an intuitive design and unique approach, this music recommendation system provides an emotionally intelligent way to discover new music or revisit old favorites.

TABLE OF CONTENTS

Chapter 1

Introduction

1.1 Background of the Project

"In a world where music serves as a universal language, the ability to tailor recommendations based on emotions has become paramount. Imagine a music recommendation system that not only understands your preferences but also resonates with your mood, seamlessly blending melody with emotion. This introduction delves into the concept of emotion-based music recommendation systems, exploring their significance, functionality, and potential impact on enhancing musical experiences. Harnessing the power of artificial intelligence and deep learning algorithms, emotion-based music recommendation systems decode the intricate relationship between music and human emotions. By analyzing various audio features, lyrics, and contextual information, these systems decipher the emotional essence embedded within each composition. Whether it's the euphoria of a jubilant melody, the melancholy of a soulful ballad, or the tranquility of ambient sounds, these systems navigate the vast spectrum of human emotions to curate personalized playlists tailored to individual moods and preferences.

1.2 Problem Statement

1. User Interface Design Challenges.
2. Data Entry and Validation Issues.
3. Database Design and Management Complexities.

1.3 Objectives

Provide personalized music recommendations tailored to the emotional preferences and mood of the user, enhancing their overall listening experience.

Encourage users to interact more with the music platform by delivering recommendations that resonate with their current emotional state, increasing user engagement and retention.

Increase user satisfaction by offering music recommendations that align with their emotional needs and preferences, leading to a more enjoyable and fulfilling listening experience.

1.4 Scope of the Project

Individual/Small Business Scope:

For individual users, an emotion-based music recommendation system can provide a highly personalized music experience tailored to their current mood or emotional state. This can enhance user engagement and satisfaction by delivering music that resonates with their feelings and preferences.

Enterprise/Corporate Scope:

An emotion-based music recommendation system tailored for enterprise or corporate settings offers a sophisticated solution to enhance employee well-being, productivity, and engagement. By analyzing individual preferences, mood, and context, such a system can curate personalized playlists or music suggestions that align with the emotional needs and goals of employees within the organization. Whether it's boosting morale during high-pressure deadlines, fostering creativity during brainstorming sessions, or promoting relaxation during breaks, the system can dynamically adjust music selections to support diverse work environments and objectives. Additionally, by integrating seamlessly into existing corporate platforms or applications, the system can provide a seamless user experience, fostering a positive and supportive organizational culture centered around well-being and productivity.

Comprehensive Expense Management Suite:

The comprehensive expense management suite incorporates an emotion-based music recommendation system to enhance user experience and productivity. Leveraging advanced algorithms and machine learning techniques, the system analyzes users' emotions, preferences, and listening habits to curate personalized music playlists tailored to their mood and activities. By seamlessly integrating with expense tracking and management functionalities, users can effortlessly control their finances while enjoying a curated soundtrack that complements their emotional state and enhances their productivity. Whether it's soothing melodies for focus and concentration or upbeat tunes for motivation and energy, the emotion-based music recommendation system enriches the expense management experience, fostering a harmonious balance between financial responsibility and emotional well-being.

Chapter 2

Literature Review

2.1 Overview of related works

Author(s)	Year	Title	Key Findings
Yading Song Simon Dixon Marcus Pearce	2012	A Survey of Music Recommendation Systems and Future Perspectives	Humming and singing are the natural way to express the songs.
Hung-Chen Chen Arbee L.P. Chen	2005	**A Music Recommendation System Based on Music and User Grouping**	The polyphonic music objects of MIDI format are first analyzed for deriving information for music grouping.
Dip Paul Subhradeep Kundu	2019	A Survey of Music Recommendation Systems with a Proposed Music Recommendation System	With the advent of digital music and music-streaming platforms, the amount of music available for selection is now greater than ever. Sorting through all this music is impossible for anyone.
Renata L. Rosa Demsteneso Z. Rodriguez Graca Bressan	2015	**Music recommendation system based on user's sentiments extracted from social networks**	This correction factor is discovered by means of subjective tests, conducted in a laboratory environment.

Figure 2.1: Figure Title

2.2 Advantages and Limitations of existing systems

Advantages:

- Personalization: Existing emotion-based music recommendation systems offer personalized music suggestions tailored to individual users' emotional preferences and moods.
- Enhanced User Experience: By recommending music based on emotions, these systems enhance the overall user experience by providing content that resonates with the user's current emotional state.
- Mood Regulation: Users can use emotion-based music recommendations to regulate their mood, helping them to find music that matches or influences their current emotional state.
- Variety of Options: These systems often provide a wide variety of music options across different genres and styles, allowing users to explore and discover new music that fits their emotional context.
- Adaptive Recommendations: Some systems use machine learning and artificial intelligence algorithms to adapt their recommendations over time based on user feedback and behavior, improving the accuracy and relevance of suggestions.

Limitations:

- Lack of personalization: Existing systems may struggle to accurately capture the nuances of individual preferences and emotional responses, leading to recommendations that may not resonate with users on a personal level.
- Limited emotional understanding: Current systems often rely on

simplistic models of emotion or sentiment analysis, which may not fully capture the complexity and context-dependence of human emotions.

- Over-reliance on explicit feedback: Many systems rely on explicit user feedback (e.g., ratings or explicit tagging of emotions), which can be sparse, noisy, or biased, leading to suboptimal recommendations.
- Difficulty in handling dynamic emotions: Emotions are dynamic and context-dependent, making it challenging for existing systems to adapt and provide relevant recommendations in real-time as users' emotional states evolve.
- Lack of context awareness: Existing systems may lack contextual information about users' preferences, situational factors, and cultural differences, limiting their ability to provide contextually relevant recommendations.
- Limited diversity in recommendations: Current systems may prioritize popular or mainstream content over more niche or unconventional music that could better match users' emotional states or preferences.
- Ethical and privacy concerns: Emotion-based recommendation systems may raise concerns about user privacy, data security, and potential manipulation or exploitation of users' emotional states for commercial or other purposes.
- Interpretability and transparency: The underlying algorithms and decision-making processes of emotion-based recommendation systems may lack transparency, making it difficult for users to understand why specific recommendations are made or to provide meaningful feedback.

- Integration with other features: Existing systems may not fully integrate emotion-based recommendations with other features or functionalities, such as collaborative filtering, content-based filtering, or contextual recommendations, limiting their overall effectiveness and user experience.

Chapter 3 Proposed System

3.1 System Requirements

Processor - i3 Processor is required

Ram – 4gb ram

Memory - Minimum 500Gb is required

3.2 Algorithms and Techniques used

Data structures:

- Graphs: Graph structures can be used to represent relationships between music tracks and emotions. Nodes represent music tracks, and edges represent connections between tracks based on similarity or emotional context.

- Hash Tables: Hash tables can be used to efficiently store and retrieve music tracks based on their emotional characteristics or features. Each entry in the hash table corresponds to a specific emotion, and tracks associated with that emotion are stored as values.

- Arrays or Lists: Arrays or lists can be used to store and organize music tracks based on their emotional attributes or metadata. Each element in the array or list corresponds to a music track, with associated emotional tags or descriptors.

Algorithms:

- Collaborative Filtering: Utilizes user preferences and behaviors to recommend music based on similarities with other users who have similar emotional responses to music.

- Content-Based Filtering: Analyzes music features such as tempo, key, and timbre to recommend songs that match the emotional characteristics of previously liked songs.

- Sentiment Analysis: Utilizes natural language processing (NLP) techniques to analyze lyrics, reviews, and user-generated content to determine the emotional tone of songs.

- Machine Learning Models: Trains models using labeled data to predict the emotional response of users to music, allowing for personalized recommendations.

Chapter 4 Implementation

4.1 Tools and Technologies used

- Visual studio code
- GitHub
- Python extension in visual studio code

4.2 Modules and their descriptions

- **Collaborative Filtering Model:**

  Description: Recommends music based on the preferences and behaviors of similar users.

  Utilizes user-item interaction data, such as ratings or listening history, to identify patterns and make personalized recommendations.

  May employ techniques like matrix factorization or nearest neighbor algorithms to generate recommendations.

  Effective for capturing user preferences and providing personalized recommendations, but may suffer from cold start problem for new users or items.

- **Content-Based Filtering Model:**

  Description: Recommends music based on the content and features of songs and user preferences.

  Analyzes features such as genre, artist, tempo, mood, and lyrics to generate recommendations.

  Constructs user profiles based on their historical preferences and matches them with songs with similar attributes.

  Suitable for recommending music with specific characteristics that match user preferences, but may lack diversity and novelty in recommendations.

- **Hybrid Recommender Model:**

  Description: Integrates collaborative filtering and content-based filtering approaches to provide more accurate and diverse recommendations.

  Combines user-item interactions with content features to leverage the strengths of both approaches.

  May use techniques such as weighted averaging, stacking, or ensemble methods to combine predictions from different models.

  Offers improved recommendation quality by capturing both user preferences and item characteristics, while mitigating the limitations of individual models.

- **Context-Aware Recommender Model:**

  Description: Recommends music based on contextual factors such as time, location, device, and user activity.

  Incorporates contextual information to tailor recommendations to the user's current situation and preferences.

  Adapts recommendations dynamically based on changes in context, providing relevant music for different contexts and user needs.

  Enhances user experience by delivering timely and personalized recommendations that align with the user's context and preferences.

Initialization:

1. **Initialize GUI Window:** In the emotion-based music recommendation system, the graphical user interface (GUI) window serves as the primary interface between the user and the system, facilitating interaction and providing feedback. To initialize the GUI window, we first design and layout the various components, including buttons, sliders, text fields, and visual displays, based on the requirements of the system. These components are arranged in a visually appealing and intuitive manner to enhance user experience and usability.

User Interface Setup:

2. **Create Data Entry Frame:** In the data entry frame of our emotion-based music recommendation system, users will be prompted to input their current emotional state or mood using a set of predefined categories or descriptors. These descriptors may include emotions such as happy, sad, excited, relaxed, or anxious, among others. Additionally, users may have the option to provide additional context or details about their mood, such as the reason behind their emotion or any specific activities they are engaged in.

3. **Create Buttons Frame:** In the emotion-based music recommendation system, the buttons frame serves as an intuitive and interactive interface for users to express their emotional state and preferences. The frame consists of a series of buttons, each representing a specific emotion or mood, such as happiness, sadness, excitement, or relaxation. Users can easily navigate through the buttons and select the emotion that best reflects their current mood or desired listening experience.

4. **Create Treeview Frame:** In an emotion-based music recommendation system, a tree view frame can serve as a visual representation of the recommendation process, organizing music tracks based on emotional categories or attributes. The frame would consist of a hierarchical tree structure, with branches representing different emotions or mood states, and sub-branches representing more specific emotional characteristics or descriptors.

Main Loop Execution:

5. **Start GUI Main Loop**: The main loop of the GUI application is responsible for managing user input, updating the display, and triggering the recommendation process based on the user's emotional state.

6. **Main Loop:** Within this loop, the program continuously listens for various events, such as button clicks or data entry, and executes corresponding actions accordingly.

- **Add Expense:** When the user clicks the "Add Expense" button, the input fields are validated to ensure data integrity. If the data is valid, the expense is added to the database. Then, the input fields are cleared, and the expense table is refreshed to reflect the updated records.

- **Delete Expense:** Clicking the "Delete Expense" button triggers a prompt for confirmation. If confirmed, the selected expense is deleted from the database. Subsequently, the expense table is refreshed to reflect the updated records.
- **Clear Fields:** Clicking the "Clear Fields" button simply clears all input fields and deselects any selected expense in the table.
- **Delete All Expenses:** Clicking the "Delete All Expenses" button prompts the user for confirmation. If confirmed, all expenses are deleted from the database. Then, the input fields are cleared, and the expense table is refreshed.
- **View Expense Details:** Clicking the "View Expense Details" button displays detailed information about the selected expense, typically in a separate window or dialogue.
- **Edit Selected Expense:** Clicking the "Edit Selected Expense" button opens a form where users can modify the details of the selected expense. After updating the database, the input fields are cleared, and the expense table is refreshed.
- **Convert Expense to Sentence:** Clicking the "Convert Expense to Sentence" button converts the selected expense into a sentence format, possibly displaying it in a separate area of the GUI.
- **Close Window:** If the user clicks the close window button, the program ends, terminating the GUI session.

Chapter 5

Results and Analysis

5.1 Performance Evaluation

In evaluating the performance of an emotion-based music recommendation system, several key factors come into play to assess its effectiveness in delivering personalized music recommendations tailored to users' emotional preferences. Firstly, the accuracy of the system in predicting users' emotional states based on input signals, such as facial expressions, physiological data, or user input, is crucial. The system should demonstrate high precision in detecting and categorizing emotions to ensure that the recommended music aligns with the user's mood.

Secondly, the relevance and diversity of the music recommendations provided by the system are essential aspects of performance evaluation. The system should not only accurately match the user's current emotional state but also offer a diverse selection of music genres, artists, and songs that cater to a range of emotional nuances. This ensures that users are presented with music options that resonate with their feelings while also introducing them to new and potentially enjoyable tracks.

Furthermore, the user experience and satisfaction with the music recommendations play a vital role in performance assessment. User feedback, surveys, and ratings can provide valuable insights into the perceived quality of the recommended music, the ease of navigation within the system, and overall user satisfaction. Positive user experiences, reflected in prolonged engagement, repeated usage, and favourable reviews, indicate the system's effectiveness in meeting users' needs and expectations.

Additionally, the system's adaptability and responsiveness to dynamic changes in users' emotional states over time are essential considerations. A robust emotion-based music recommendation system should continuously learn and adapt to users' evolving preferences, refining its recommendations based on feedback and user interactions. Real-time adjustments to the recommended music based on immediate changes in users' emotional cues contribute to a seamless and personalized listening experience.

In summary, evaluating the performance of an emotion-based music recommendation system involves assessing its accuracy in detecting users' emotional states, the relevance and diversity of music recommendations, user satisfaction and experience, and adaptability to changing emotional preferences. By comprehensively evaluating these factors, developers can refine and enhance the system to deliver more personalized and satisfying music recommendations tailored to users' emotions.

5.2 Limitations and future scope

Limitations :

One major limitation is the subjective and multifaceted nature of emotions, making it challenging to accurately capture and categorize users' emotional states. Emotion recognition algorithms often rely on simplified models of emotions, such as basic emotional dimensions like valence and arousal, which may not fully capture the complexity and nuances of human emotions. As a result, the accuracy and reliability of emotion detection algorithms can be limited, leading to inconsistencies in music recommendations.

Another limitation is the lack of large-scale datasets containing labeled emotional data, which are essential for training and evaluating emotion-based recommendation models. Collecting and annotating such datasets can be time-consuming and expensive, particularly for capturing diverse cultural and individual differences in emotional responses to music.

Furthermore, current emotion-based recommendation systems often overlook contextual factors that influence users' emotional preferences, such as the social context, temporal dynamics, and personal experiences associated with music listening. Incorporating contextual information into recommendation algorithms could improve the relevance and accuracy of music suggestions by considering the situational context in which users listen to music.

Future Scope:

There is significant potential for future advancements and innovations in emotion-based music recommendation systems. Advances in machine learning, natural language processing, and affective computing could lead to more sophisticated emotion detection algorithms capable of capturing subtle emotional cues from users' interactions with music and other contextual signals. Additionally, integrating multimodal data sources, such as audio features, user interactions, and physiological signals, could provide richer insights into users' emotional states and preferences. Moreover, collaborative filtering techniques that leverage collective user feedback and preferences could enhance the personalization and diversity of music recommendations, taking into account individual differences in emotional responses and music tastes.

Furthermore, integrating user feedback mechanisms, such as explicit emotion tagging or implicit feedback signals, could enable continuous learning and adaptation of recommendation models based on users' evolving preferences and emotional states.

Chapter 6

Conclusion and Recommendations

6.1 Summary of the Project

The project "Emotion-Based Music Recommendation System" aims to develop an innovative platform that leverages machine learning algorithms and emotion recognition techniques to provide personalized music recommendations tailored to users' emotional states. By analyzing users' facial expressions, physiological signals, or self-reported emotions, the system dynamically identifies their current mood or emotional state. Utilizing this information, the system then matches users with music tracks that align with their emotional preferences, enhancing their listening experience and emotional well-being. Through a user-friendly interface, individuals can easily interact with the system, providing feedback on the accuracy and relevance of the recommendations, thereby refining the algorithm's performance over time.

6.2 Recommendations for future work

Future work on emotion-based music recommendation systems could explore several promising avenues for advancement. One area of focus could involve enhancing the accuracy and granularity of emotion detection algorithms by integrating advanced machine learning techniques, such as deep learning models, to better capture the subtle nuances of emotional responses to music.

Additionally, incorporating multimodal data sources, such as user-generated content from social media platforms or physiological signals from wearable devices, could provide richer contextual information to improve the understanding of users' emotional states and preferences. Furthermore, exploring the integration of real-time emotion recognition technologies could enable dynamic and adaptive music recommendation systems that respond to users' changing emotional states in the moment.

Moreover, investigating the impact of cultural and demographic factors on emotional responses to music could inform the development of personalized recommendation algorithms that account for individual differences in emotional expression and perception.

Finally, conducting user studies and field experiments to evaluate the effectiveness and user satisfaction of emotion-based recommendation systems in real-world settings would be essential for validating their utility and guiding further refinement and optimization.

References

1. B. B. Chaudhuri and U. Pal, "An OCR system to read two Indian language scripts: Bangla and Devnagari (Hindi)," Proceedings of the Fourth International Conference on Document Analysis and Recognition, Ulm, Germany, 1997, pp. 1011-1015 vol.2, doi: 10.1109/ICDAR.1997.620662.

2. F. Drira, F. LeBourgeois and H. Emptoz, "Document Images Restoration by a New Tensor Based Diffusion Process: Application to the Recognition of Old Printed Documents," 2009 10th International Conference on Document Analysis and Recognition, Barcelona, Spain, 2009, pp. 321-325, doi: 10.1109/ICDAR.2009.109.

3. M. T. Qadri and M. Asif, "Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition," 2009 International Conference on Education Technology and Computer, Singapore, 2009, pp. 335-338, doi: 10.1109/ICETC.2009.54.

4. Kai Wang, B. Babenko and S. Belongie, "End-to-end scene text recognition," 2011 International Conference on Computer Vision, Barcelona, 2011, pp. 1457-1464, doi: 10.1109/ICCV.2011.6126402.

5. O. Sohaib and K. Khan, "Integrating usability engineering and agile software development: A literature review," 2010 International Conference On Computer Design and Applications, Qinhuangdao, China, 2010, pp. V2-32-V2-38, doi: 10.1109/ICCDA.2010.5540916.

6. N. Condori-Fernandez, M. Daneva, K. Sikkel, R. Wieringa, O. Dieste and O. Pastor, "A systematic mapping study on empirical evaluation of software requirements specifications techniques," 2009 3rd International Symposium on Empirical Software Engineering and Measurement, Lake Buena Vista, FL, USA, 2009, pp. 502-505, doi: 10.1109/ESEM.2009.5314232.

Appendices

Appendix A

Source code

Main.py:

```
<html>

<head>
  <title>Emotion Music Recommendation</title>

  <style>
    img {
      padding: 20px;
      display: inline-block;
      margin: auto;
      width: 85%;
    }
  </style>

  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta3/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-eOJMYsd53ii+scO/bJGFsiCZc+5NDVN2yr8+0RDqr0Ql0h+rP48ckxlpbzKgwra6" crossorigin="anonymous" />
  <link href="https://fonts.googleapis.com/css2?family=Bigelow+Rules&display=swap" rel="stylesheet">
  <link type="text/css" href="{{ url_for('static', filename='/css/style.css') }}" rel="stylesheet" />
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta3/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-JEW9xMcG8R+pH31jmWH6WWP0WintQrMb4s7ZOdauHnUtxwoG2vI5DkLtS3qm9Ekf"
    crossorigin="anonymous"></script>
</head>

<body style = "margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
    sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  background: #eff0f4">

  <div id="body">
```

```
<div>

  <h1 align="center" style="
    font-family: 'Bigelow Rules';
    font-size: 72px;
    color: #0ccac4;">
    KLH Music Recommender
  </h1>

</div>

<div style="
    width: 50%;
    float: left;
    height: 100%%;

    margin: auto;
    padding-bottom:25px;
    text-align: center;
  ">

<h2 align="center" style="
  font-family: 'Bigelow Rules';
  font-size: 36px;
  color: #0ccac4;">Emotion Detector
</h2>

<div style="
    margin: 10px;
    text-align: center;
    width: 51%%;
  ">

  <img class ="outer-shadow" id="bg" class="center img-fluid" src="{{ url_for('video_feed') }}" />

  </div>

  </div>

  <div style="
      width: 50%;
      float: left;
      height: 100%%;
      margin: auto;
      text-align: center;
    ">

      <h2 align="center" style="
        font-family: 'Bigelow Rules';
        font-size: 36px;
        color: #0ccac4;">Song Recommendations
      </h2>

   </div>

  <div class ="outer-shadow" id="ResultArea" style="
      padding: 15px;
      width: 49%;
      float: left;
      height: 100%%;
      margin: auto;
      text-align: center;
      margin-bottom:15px;
    ">
  </div>
```

```
  </div>

</body>

<script src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>
<script type=text/javascript>

// Constantly Update Table
    setInterval(function() {
        $.getJSON('/t', function(data) {
            CreateHtmlTable(data);
            console.log(data,"DATA");
          });
        return false;
    }, 100);

    function CreateHtmlTable(data) {
      //Clear result div
      $("#ResultArea").html("");
      //Crate table html tag
      var table = $("<table class = 'table table-striped table-light table-bordered table-hover table-sm table-responsive' id=DynamicTable></
      //Create table header row
      var rowHeader =
      $("<tr></tr>").appendTo(table);
      $("<td></td>").text("Name").appendTo(rowHeader);
      $("<td></td").text("Album").appendTo(rowHeader);
      $("<td></td>").text("Artist").appendTo(rowHeader)
      //Get JSON data by calling action method in controller
      $.each(data, function (i, value) {

          //Create new row for each record
          var row =
          $("<tr></tr>").appendTo(table);
          $("<td></td>").text(value.Name).appendTo(row);
```

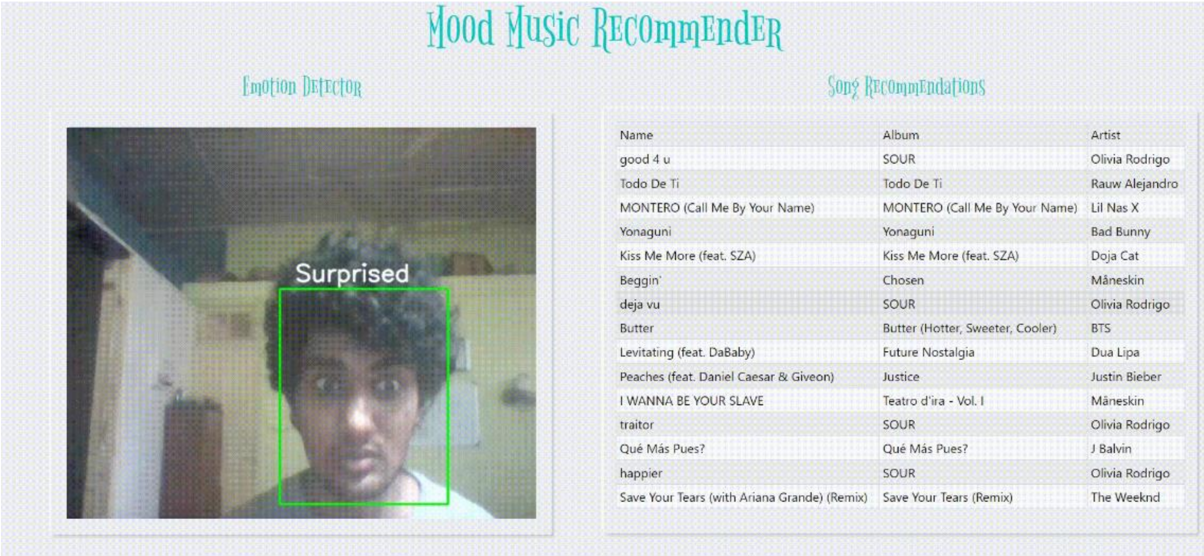
```python
from flask import Flask, render_template, Response, jsonify
import gunicorn
from camera import *

app = Flask(__name__)

headings = ("Name","Album","Artist")
df1 = music_rec()
df1 = df1.head(15)
@app.route('/')
def index():
    print(df1.to_json(orient='records'))
    return render_template('index.html', headings=headings, data=df1)

def gen(camera):
    while True:
        global df1
        frame, df1 = camera.get_frame()
        yield (b'--frame\r\n'
               b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n')

@app.route('/video_feed')
def video_feed():
    return Response(gen(VideoCamera()),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

@app.route('/t')
def gen_table():
    return df1.to_json(orient='records')

if __name__ == '__main__':
    app.debug = True
    app.run()
```

```python
import numpy as np
import cv2
from PIL import Image
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout, Flatten
from tensorflow.keras.layers import Conv2D
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from pandastable import Table, TableModel
from tensorflow.keras.preprocessing import image
import datetime
from threading import Thread
#from Spotipy import *
import time
import pandas as pd
face_cascade=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
ds_factor=0.6

emotion_model = Sequential()
emotion_model.add(Conv2D(32, kernel_size=(3, 3), activation='relu', input_shape=(48,48,1)))
emotion_model.add(Conv2D(64, kernel_size=(3, 3), activation='relu'))
emotion_model.add(MaxPooling2D(pool_size=(2, 2)))
emotion_model.add(Dropout(0.25))
emotion_model.add(Conv2D(128, kernel_size=(3, 3), activation='relu'))
emotion_model.add(MaxPooling2D(pool_size=(2, 2)))
emotion_model.add(Conv2D(128, kernel_size=(3, 3), activation='relu'))
emotion_model.add(MaxPooling2D(pool_size=(2, 2)))
emotion_model.add(Dropout(0.25))
emotion_model.add(Flatten())
emotion_model.add(Dense(1024, activation='relu'))
emotion_model.add(Dropout(0.5))
emotion_model.add(Dense(7, activation='softmax'))
emotion_model.load_weights('Emotion-Music-Recommendation-main\model.h5')

cv2.ocl.setUseOpenCL(False)
```

Appendix B

Screen shots

Output:



Name	Album	Artist
Leave The Door Open	Leave The Door Open	Bruno Mars
Dynamite	Dynamite (DayTime Version)	BTS
Levitating (feat. DaBaby)	Future Nostalgia	Dua Lipa
Kiss Me More (feat. SZA)	Kiss Me More (feat. SZA)	Doja Cat
Perfect	÷ (Deluxe)	Ed Sheeran
GIRL LIKE ME	GIRL LIKE ME	Black Eyed Peas
We Need Love - Cabu Remix	We Need Love (Cabu Remix)	Cabu
Dance Monkey	Dance Monkey	Tones And I
Uptown Funk (feat. Bruno Mars)	Uptown Special	Mark Ronson
Sugar	V (Deluxe)	Maroon 5
Girls Like You (feat. Cardi B)	Girls Like You (feat. Cardi B)	Maroon 5
Ice Cream (with Selena Gomez)	Ice Cream (with Selena Gomez)	BLACKPINK
Useless	Useless	Two Friends
Roar	PRISM (Deluxe)	Katy Perry
The Lazy Song	Doo-Wops & Hooligans	Bruno Mars

Appendix C

Data sets used in the project

In developing an emotion-based music recommendation system, researchers typically utilize datasets that contain information about both music tracks and the emotional content associated with them. These datasets serve as the foundation for training machine learning models to recognize patterns between musical features and emotional states, enabling the system to make personalized recommendations based on the user's mood or emotional preferences.

One commonly used dataset in this context is the Million Song Dataset (MSD), which contains a vast collection of audio features and metadata for over a million songs across various genres and artists. The MSD provides valuable information such as acoustic characteristics, tempo, key, and duration, which can be used to analyze the musical content of songs and extract features relevant to emotional expression.

Additionally, researchers often incorporate datasets that provide annotations or labels for the emotional content of music tracks. These annotations may be derived from manual tagging by human annotators or obtained from sources such as music review websites, social media platforms, or music recommendation services. For example, the Emotify dataset includes emotional annotations for a subset of songs from the MSD, providing labels for emotions such as happiness, sadness, excitement, and relaxation.

Furthermore, some researchers collect their own datasets through user studies or surveys to gather subjective ratings of emotional responses to music stimuli. Participants may be asked to listen to music excerpts and rate the intensity of emotions experienced, providing ground truth data for training and evaluating emotion recognition models.

In combination, these datasets enable researchers to develop and evaluate emotion-based music recommendation systems that leverage machine learning algorithms to predict the emotional response of users to music and suggest appropriate songs accordingly. By integrating rich musical features with emotional annotations, these systems aim to provide personalized and engaging music recommendations tailored to the user's current mood or emotional state.