PROJECT REPORT
ON
Emotion Based Music Player

SUBMITTED BY

Faiz Ziauddin Ansari

SEAT NO: 701

PROJECT GUIDE

Ms. MANALI PATIL

BSC. (COMPUTER SCIENCE) SEM -VI
2021 – 2022

CONDUCTED

AT

CHIKITSAK SAMUHA'S

S. S. & L.S. PATKAR COLLEGE OF ARTS & SCIENCE

AND

V. P. VARDE COLLEGE OF COMMERCE & ECONOMICS

GOREGAON (W). MUMBAI -400062

CHIKITSAK SAMUHA'S

S.S. & L.S. PATKAR COLLEGE OF ARTS & SCIENCE
AND
V. P. VARDE COLLEGE OF COMMERCE & ECONOMICS

Accredited A Grade by NAAC

S.V.Road, Goregaon (West), Mumbai - 400 062. Tel.: 91-022-28723731/28781188 Fax: 91-022-2874 4755
Website : www.patkarvardecollege.edu.in
E-mail : principal@patkarvardecollege.edu.in info@patkarvardecollege.edu.in

Self-Financed Courses

Project Certificate

This is to certify that Mr. /Ms.___________**Faiz Ziauddin Ansari**__________ of T.Y.B.Sc. Computer Science with University Seat no _**701**_ has completed his project titled ____**Emotion Based Music Player**______under the guidance of Project Guide Ms.____**Manali Patil**_______ as laid by University of Mumbai in the college during the year 2021-22.

B.Sc. Computer science

Project Guide **Co-ordinator**

Ms Manali Patil **Mrs Namrata Shinde**

External Examiner

Date:

ACKNOWLEDGEMENT

I take this opportunity of submitting this report to express my profound gratitude to the management of "S.S. & L.S. Patkar College of Science and Commerce" for giving me the opportunity to accomplish this project work.

I am also thankful to all the teachers for their kind support and help.

I express my deepest gratitude towards my project guide **Ms Manali Patil** for her valuable and timely advice during the various phases in my project.

I extend my sincere thanks to our respected Head of Department **Mrs. Namrata Shinde** for allowing us to use the facilities available.

I would also like to thank all those unnamed but important people and Computer Lab Assistants and supportive staff who directly or indirectly helped me in the completion of this project and to my family and friends without whose support, motivation and encouragement this would not have been possible.

Thanking you,

Faiz Ansari

Emotion Based Music Player

INDEX

Sr. No.	Contents	Page No.
1	**Introduction**	7
	1.1 Abstract	
	1.2 Objective and Scope of the project	
2	**Requirement Specification**	11
	2.1 Functional Requirement	
	2.2 Non-Functional Requirement	
	2.3 Hardware Requirement	
	2.4 Software Requirement	
3	**System planning**	13
	3.1 Gantt Chart	
4	**System Design**	14
	4.1 Architecture (Spiral Model)	
	4.2.1 Use-Case Diagram	
	4.2.2 Activity Diagram	
	4.2.3 Sequence Diagram	
5	**System Implementation:**	21
	5.1 Code implementation	
	Results	
6	6.1 Test Cases	27
	6.2 Screen Shot	

6	**Conclusion and Future Scope**	29
	References	34

1. Introduction

1.1 Abstract

This project Emotion music player (an emotion based music player) is a novel approach that helps the user to automatically play songs based on the emotions of the user.

It recognizes the facial emotions of the user and plays the songs according to their emotion. The emotions are recognized using a machine learning method Support Vector Machine (SVM)algorithm.

SVM can be used for classification or regression problems. It finds an optimal boundary between the possible outputs. The training dataset which we used is Olivetti faces which contain 400 faces and its desired values or parameters.

The webcam captures the image of the user.It then extract the facial features of the user from the captured image.

The training process involves initializing some random values for say smiling and not smiling of our model, predict the output with those values, then compare it with the model's prediction and then adjust the values so that they match the predictions that were made previously.

Evaluation allows the testing of the model against data that has never been seen and used for training and is meant to be representative of how the model might perform when in the real world. According to the emotion ,the music will be played from the predefined directories.

Key Words : SVM, Olivetti faces, Emotions, Songs, Machine learning, Training, Testing

1.2 Objective of the project

Project Emotion player (an emotion based music player) is a novel approach that helps the user to automatically play songs based on the emotions of the user. It recognizes the facial emotions of the user and plays the songs according to their emotion. The emotions are recognized using a machine learning method Support Vector Machine(SVM)algorithm.

The human face is an important organ of an individual‘s body and it especially plays an important role in extraction of an individual‘s behaviours and emotional state.The webcam captures the image of the user.It then extract the facial features of the user from the captured image. Facial expression categorized into 2, smiling and not smiling. According to the emotion, the music will be played from the predefined directories.

1.2 The scope of the project

This Report describes all the requirements for the project. The purpose of this research is to provide a all information for the combination of both structured and unstructured information of our project "**Emotion Based Music Player**" Application.

The main concept of this project is to automatically play songs based on the emotions of the user.

It aims to provide user-preferred music with emotion awareness. In existing system user want to manually select the songs, randomly played songs may not match to the mood of the user, user has to classify the songs into various emotions and then for playing the songs user has to manually select a particular emotion. These difficulties can be avoided by using Emotion Music Player (Emotion based music player).

2. Requirement Specification

2.1 Functional Requirement

The functional requirements specify what the product must do. They relate to the actions that the product must carry out in order to satisfy the fundamental reasons for its existence. requirements are statement of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situation.

- The dataset train by support vector classifier,
- Machine learns support vector classification using support vector machine.
- Learn and identify image capture by web cam.

2.2 Non-Functional Requirement:

The non-functional requirement add functionally to the product's Existence, but are needed to make the product perform in the desired manner. In Emotion Based Music Player, these requirements are met with utmost care. The design is made so that even the new users can understand the navigation and walk through of the application. This program requires below average specification of hardware but most users can run it on their system with included options.

Non functional requirements are:

- Reliability
- Maintainability
- Portability
- Extensibility
- Reusability
- Simplicity.
- Utilization

2.3 HARDWARE REQUIREMENT:

- Processor type: Minimum: Intel i3 or more.
- Minimum Ram : 4gb .
- Processor speed: 1.5 GHz or higher.
- Webcam
- Speaker

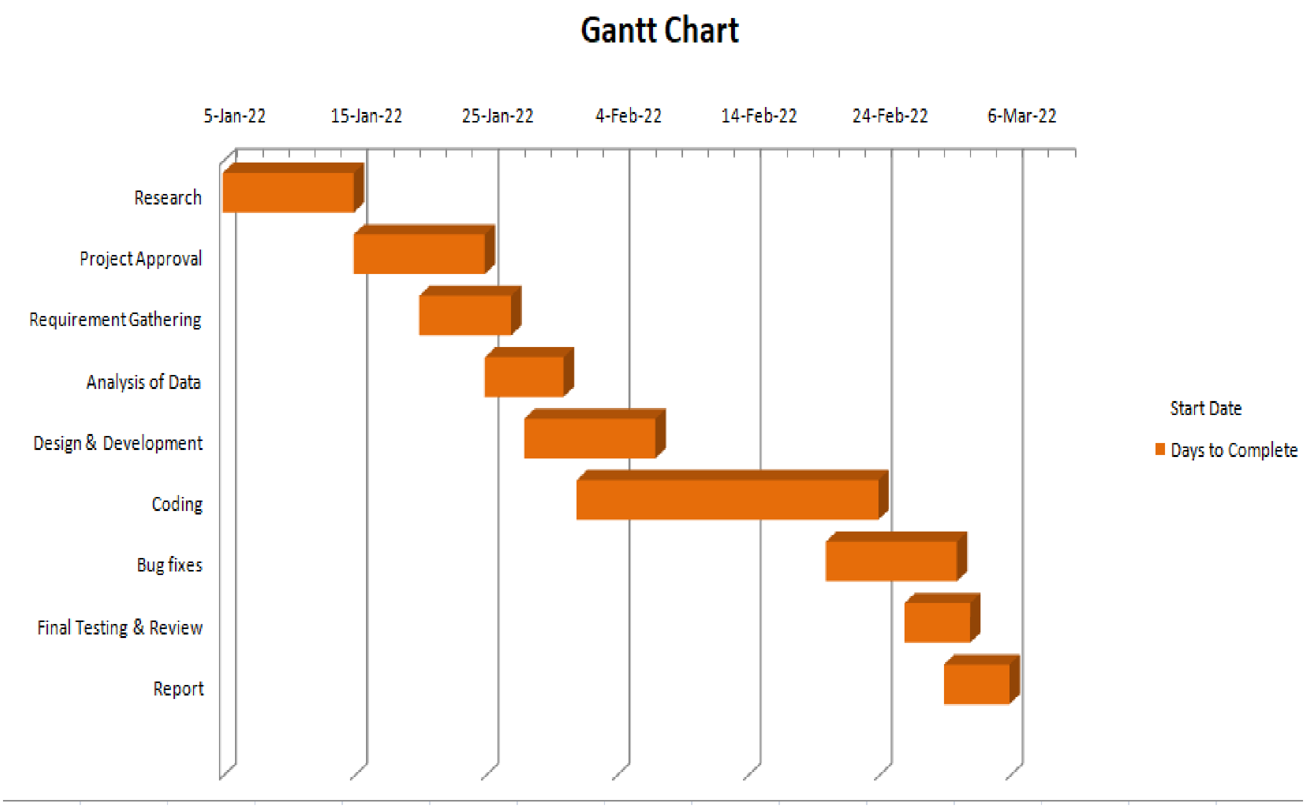
2.4 Software Requirements:

- Operating system :
  Windows 7 or above
- Language :
  Python
- Software Application :
  Visual Studio Code and any .Py editor

3.0 System planning

3.1 Gantt Chart

Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar, the position and length of reflects the start date, duration and end date of the activity.



4. System Design

4.1 Methodology Adopted

Python: - Python is an interpreted, high-level and general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is often described as a "batteries included" language due to its comprehensive standard library.

Visual Studio Code: Working with Python in Visual Studio Code, using the Microsoft Python extension, is simple, fun, and productive. The extension makes VS Code an excellent Python editor, and works on any operating system with a variety of Python interpreters. It leverages all of VS Code's power to provide auto complete and IntelliSense, linting, debugging, and unit testing, along with the ability to easily switch between Python environments, including virtual and conda environments.

Python Libraries: -

OpenCV-Python: OpenCV (Open-Source Computer Vision Library) is an open-source computer vision software library. which is built to provide a common infrastructure for computer vision. It has thousands of optimized algorithms which can be used different purposes like detecting and recognizing faces, identifying objects and many more. We need it to take pictures using our webcam and some manipulation needed to be done in the image.

NumPy: NumPy is a Python library used for working with arrays. NumPy is used to convert our images into some form of an array so that we can store the model that has been trained.

OS: The OS module in python provides functions for interacting with the operating system. OS, comes under Python's standard utility modules. This module provides a portable way of using operating system dependent functionality.

Haar Cascade: HaarCascade is basically a classifier which is used to detect the objects for which it has been trained for, from the source. We need haarcascade frontal face recognizer to detect the face from image.

MTCNN: MTCNN or Multi-Task Cascaded Convolutional Neural Networks is a neural network which detects faces and facial landmarks on images. It was published in 2016 by Zhang et al. MTCNN output example. MTCNN is one of the most popular and most accurate face detection tools today.

Matplotlib: matplotlib. pyplot is a collection of functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

Tensor Flow: TensorFlow is an end-to-end open-source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the stateof-the-art in ML and developers easily build and deploy ML powered applications.
What is Pathlib library in Python?

Pathlib: module in Python provides various classes representing file system paths with semantics appropriate for different operating systems. This module comes **under** Python's standard utility modules. Path classes in Pathlib module are divided into pure paths and concrete paths.

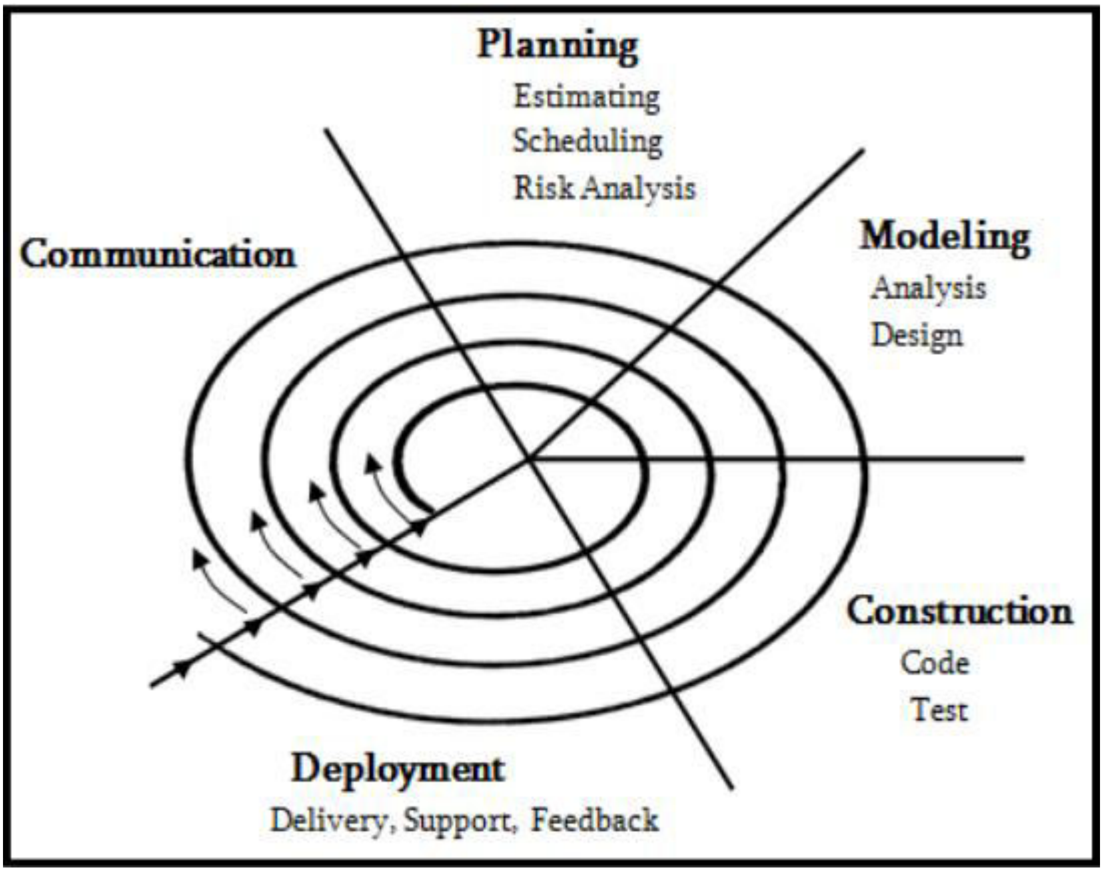
VLC: The VLC media player is a free and open-source portable cross-platform media player software and streaming media server developed by the VideoLAN project. ... Note: In order to use the vlc module in python, the user system should have vlc media player already installed on the machine.

Tkinter: It is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter is included with standard GNU/Linux, Microsoft Windows and macOS installs of Python.

4.2 Architecture

Spiral model:

Is a combination of sequential and prototype model. This model is best used for large projects which involves continuous enhancements. There are specific activities which are done in one iteration (spiral) where the output is a small prototype of the large software. The same activities are then repeated for all the spirals till the entire software is build.



Functions of Spiral Model:

Requirements : Requirements are gathered from the customers and the objectives are identified, elaborated and analyzed at the start of every phase. Then alternative solutions possible for the phase are proposed in this quadrant.

Planning : All the possible solutions are evaluated to select the best possible solution. Then the risks associated with that solution is identified and the risks are resolved using the best possible strategy. At the end of this quadrant, Prototype is built for the best possible solution.

Modeling : In this Section we will Analysis and requirements, according to needs. We will create a design of our projects and analysis of project. Internal testing and Deployments are done in Modeling.

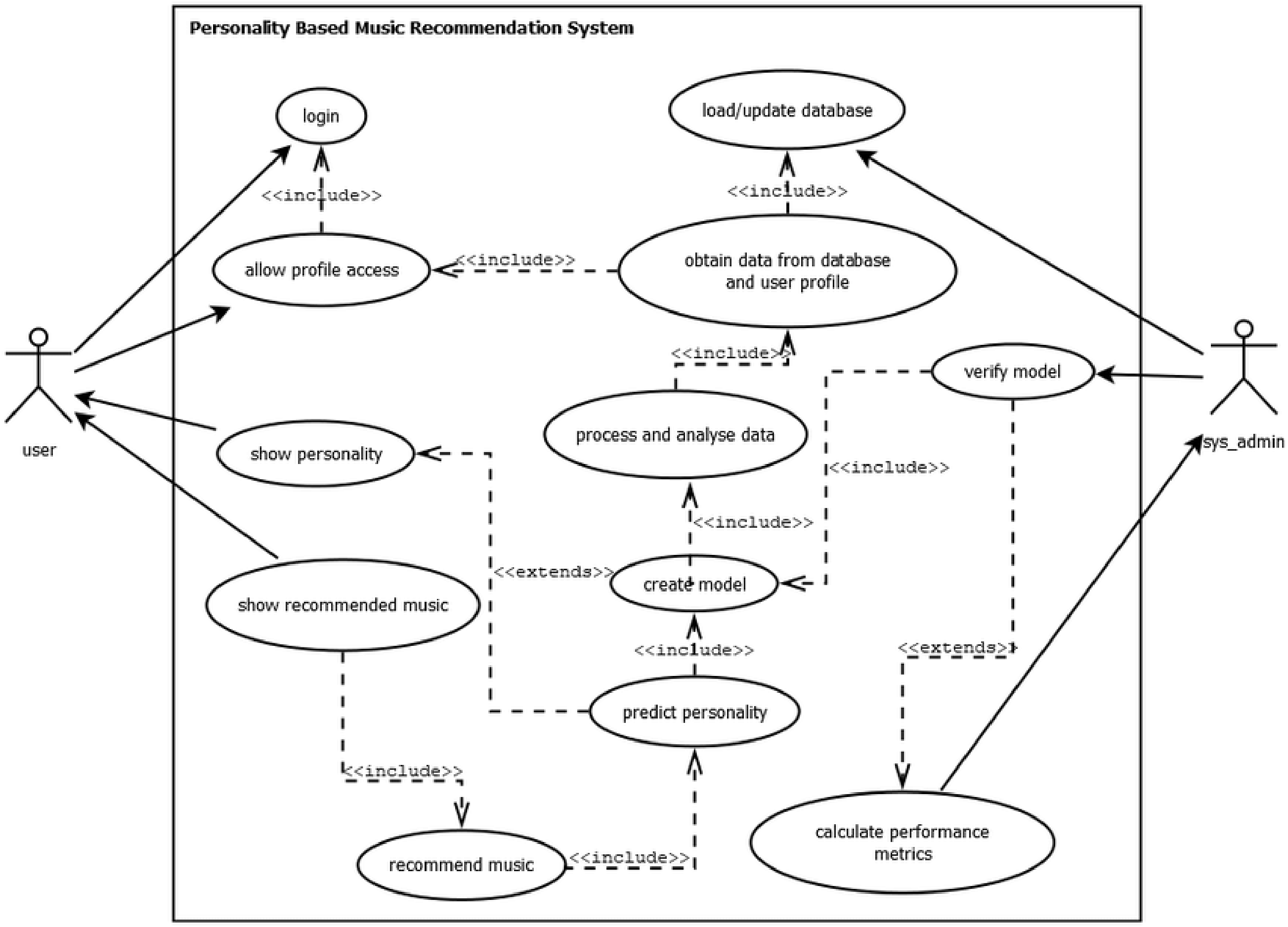
Construction : The team delivered high-quality working software in priority order, which was created in accordance with the changing needs of our potential users. What"s more important, the team could deploy this solution into a pre-production testing/QA sandbox for system integration testing.

Deployment : Once the software testing phase is over and no bugs or errors left in the system then the final deployment process starts. Based on the feedback given by the project manager, the final software is released and checked for deployment issues if any.

4.3 UML Diagrams

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

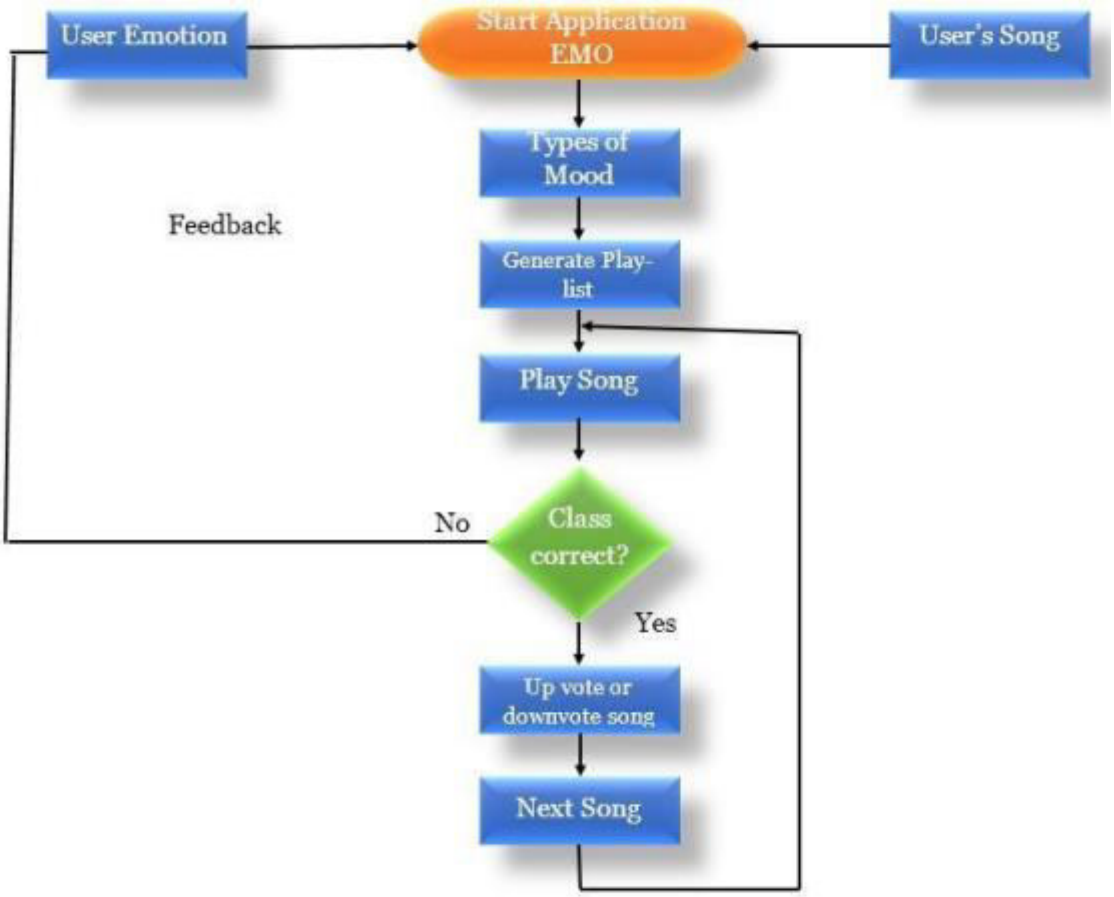
4.3.1 Use Case Diagram



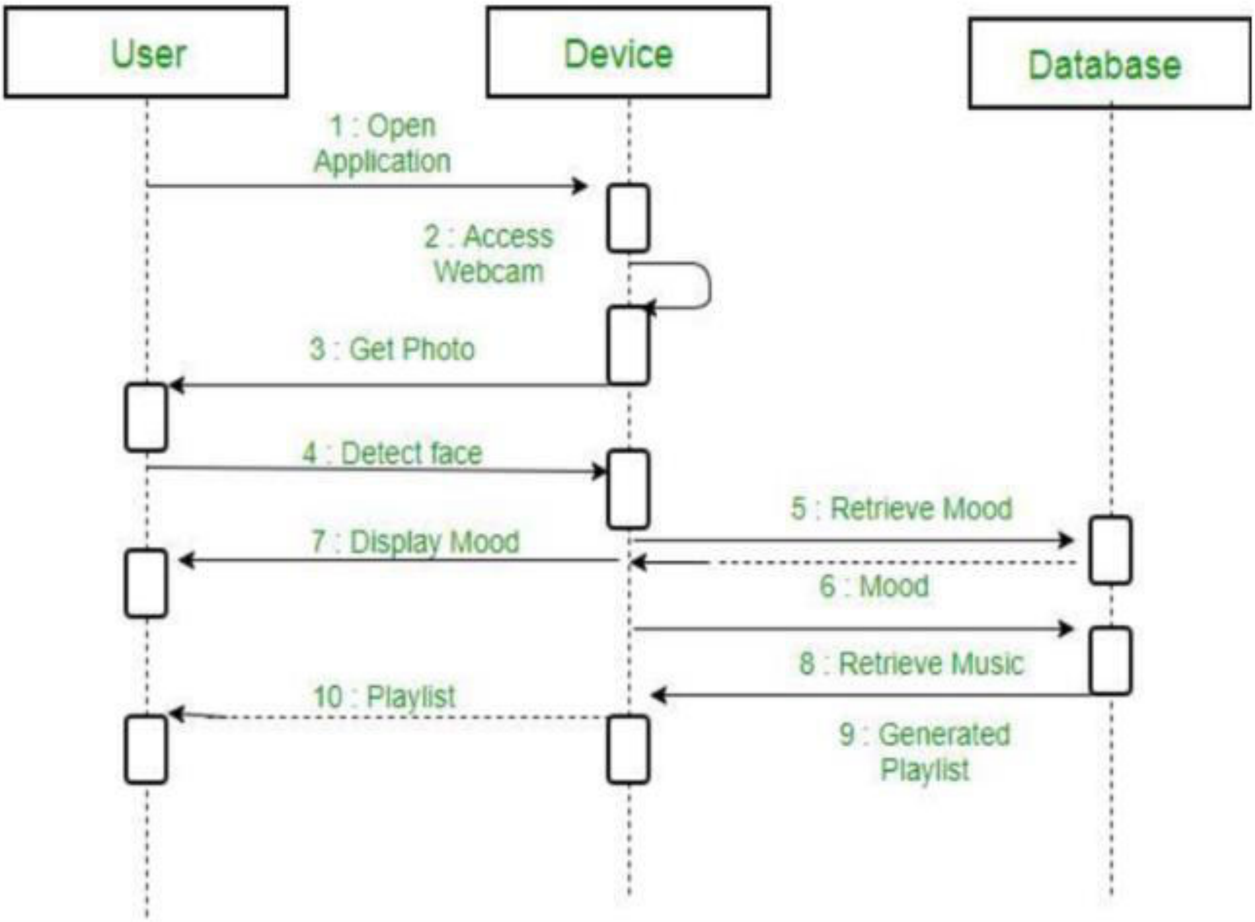
4.3.3 Activity Diagram:

An Activity Diagram visually presents a series of actions of control is system similar to a flowchart or a dataflow .

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.



4.3.4 Sequence Diagram



5. System Implementation:

5.1 Code implementation

Sample code :

Emotion.py

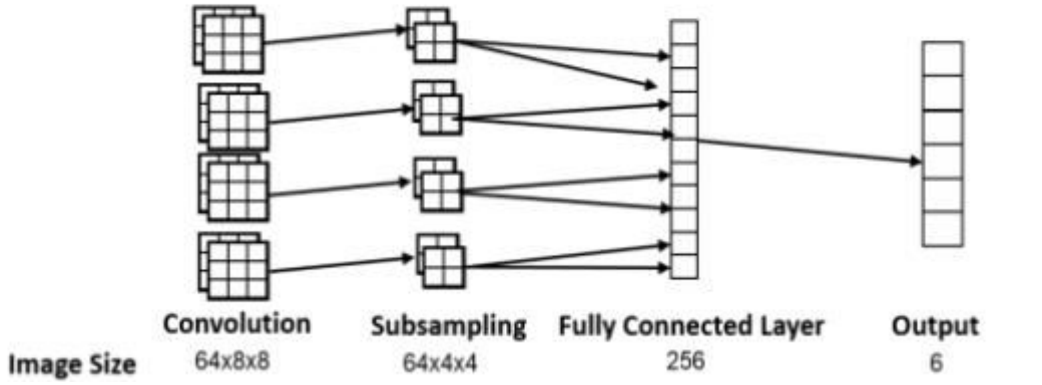
```
import numpy as np

import cv2

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Dropout, Flatten

from tensorflow.keras.layers import Conv2D

from tensorflow.keras.layers import MaxPooling2D

import os

import vlc

import time

from pathlib import Path

from random import randint

from subprocess import call

from tkinter import *

os.environ['TF_CPP_MIN_LOG_LEVEL'] = '4'


#music player function

def music_player(emotion_str):

    from musicplayer import MusicPlayer

    root = Tk()

    print('\nPlaying ' + emotion_str + ' songs')
    MusicPlayer(root,emotion_str)

    root.mainloop()



# Create the model

model = Sequential()


model.add(Conv2D(32, kernel_size=(3, 3), activation='relu', input_shape=(48,48,1)))

model.add(Conv2D(64, kernel_size=(3, 3), activation='relu'))

model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Dropout(0.25))


model.add(Conv2D(128, kernel_size=(3, 3), activation='relu'))

model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(128, kernel_size=(3, 3), activation='relu'))

model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Dropout(0.25))


model.add(Flatten())

model.add(Dense(1024, activation='relu'))

model.add(Dropout(0.5))

model.add(Dense(7, activation='softmax'))



model.load_weights('model.h5')


print('\n Welcome to Music Player based on Facial Emotion Recognition \n')

print('\n Press \'q\' to exit the music player \n')
```

```
# prevents openCL usage and unnecessary logging messages
cv2.ocl.setUseOpenCL(False)

# dictionary which assigns each label an emotion (alphabetical order)
emotion_dict = {0: "Angry", 1: "Disgusted", 2: "Fearful", 3: "Happy", 4: "Neutral", 5: "Sad", 6: "Surprised"}

#File to append the emotions
with open(str(Path.cwd())+"\emotion.txt","w") as emotion_file:

        # start the webcam feed
    cap = cv2.VideoCapture(0)
    now = time.time()  ###For calculate seconds of video
    future = now + 10
    while True:
        # Find haar cascade to draw bounding box around face
        ret, frame = cap.read()
        if not ret:
            break
        facecasc = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        faces = facecasc.detectMultiScale(gray,scaleFactor=1.3, minNeighbors=5)

        for (x, y, w, h) in faces:
            cv2.rectangle(frame, (x, y-50), (x+w, y+h+10), (255, 0, 0), 2)
            roi_gray = gray[y:y + h, x:x + w]
            cropped_img = np.expand_dims(np.expand_dims(cv2.resize(roi_gray, (48, 48)), -1), 0)
            prediction = model.predict(cropped_img)
            maxindex = int(np.argmax(prediction))
            cv2.putText(frame, emotion_dict[maxindex], (x+20, y-60), cv2.FONT_HERSHEY_SIMPLEX, 1, (255, 255, 255), 2, cv2.LINE_AA)

            text = emotion_dict[maxindex]

            emotion_file.write(emotion_dict[maxindex]+"\n")

            emotion_file.flush()


        cv2.imshow('Video', cv2.resize(frame,(300,300),interpolation = cv2.INTER_CUBIC))

        if cv2.waitKey(1) & 0xFF == ord('q'):

            break



        if time.time() > future:  ##after 10 second music will play

            cv2.destroyAllWindows()

            music_player(text)

            future = time.time() + 10


    cap.release()
```

6. Results

Trained Dataset

Convolution Neural Network



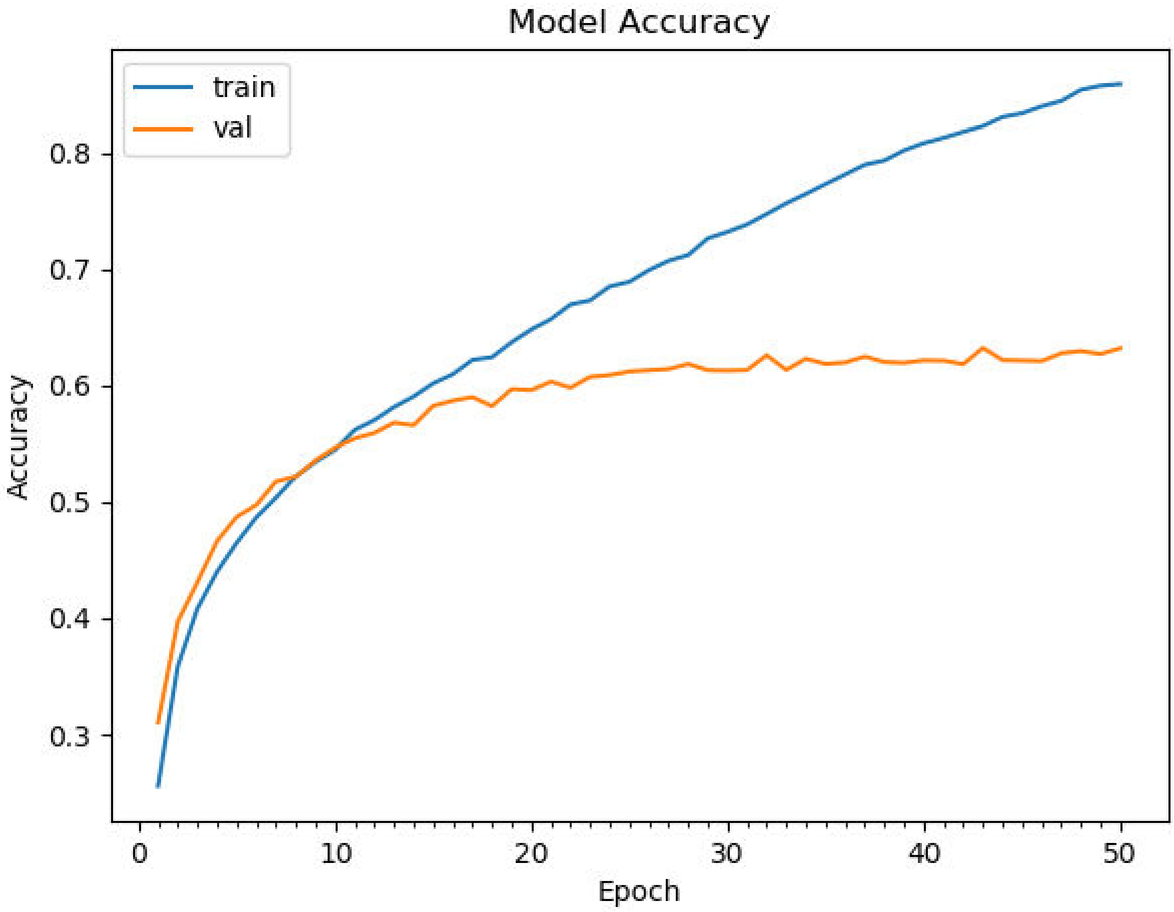
6.1 Test Cases

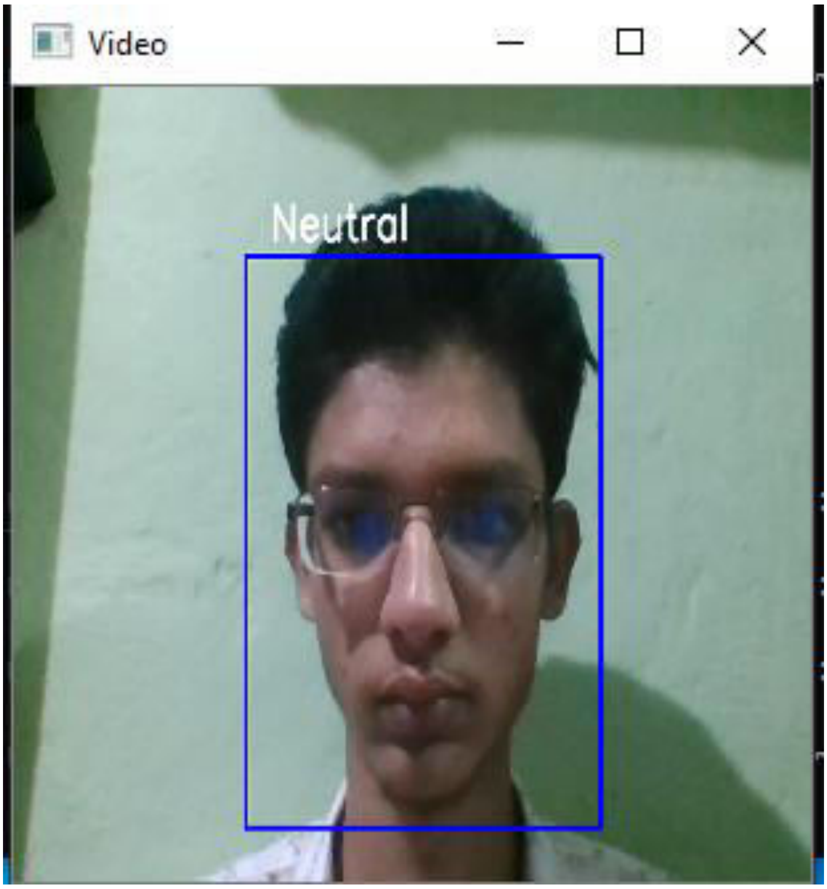
Module	Time Taken(sec)
Face Detection	0.8126
Facial Feature Extraction	0.9216
Classification using SVM	0.1956
Emotions	0.9994

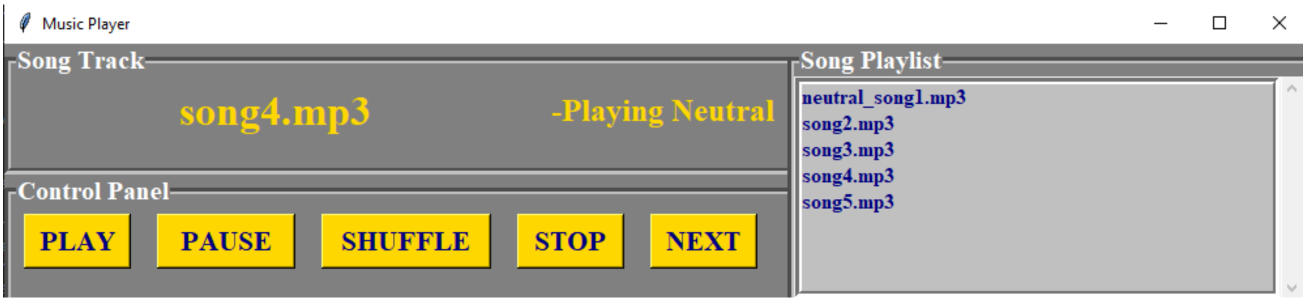
SR. No	**Test Cases**	**Facial Expression**	**Status**	**Output**	**Accuracy**
1	Happy	Happy	Performed	Yes	100%
2	Sad	Sad	Performed	Yes	91%
3	Angry	Angry	Performed	Yes	88%
4	Neutral	Neutral	Performed	Yes	100%
5	Fearful	Fearful	Performed	Yes	92%
6	Surprised	Surprised	Performed	Yes	90%

6.2 Screen Shot

Accuracy**:**





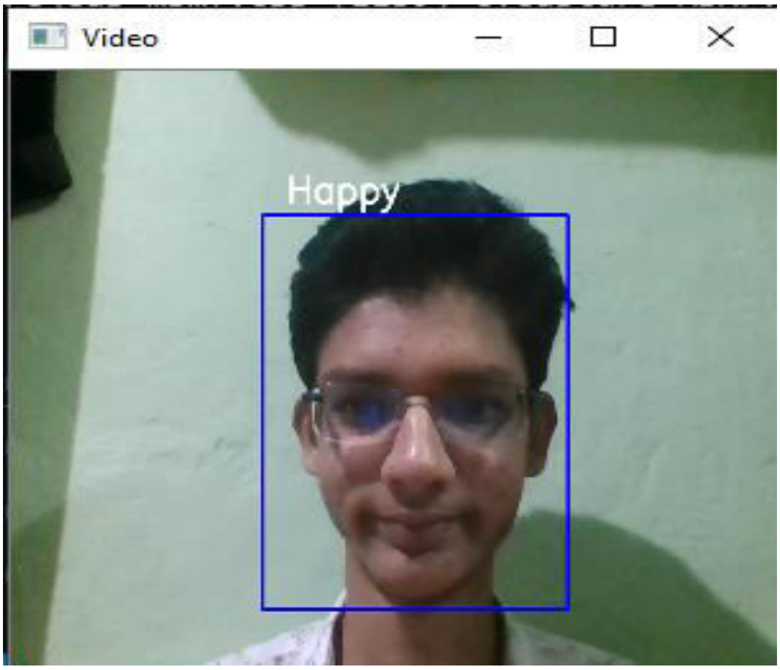


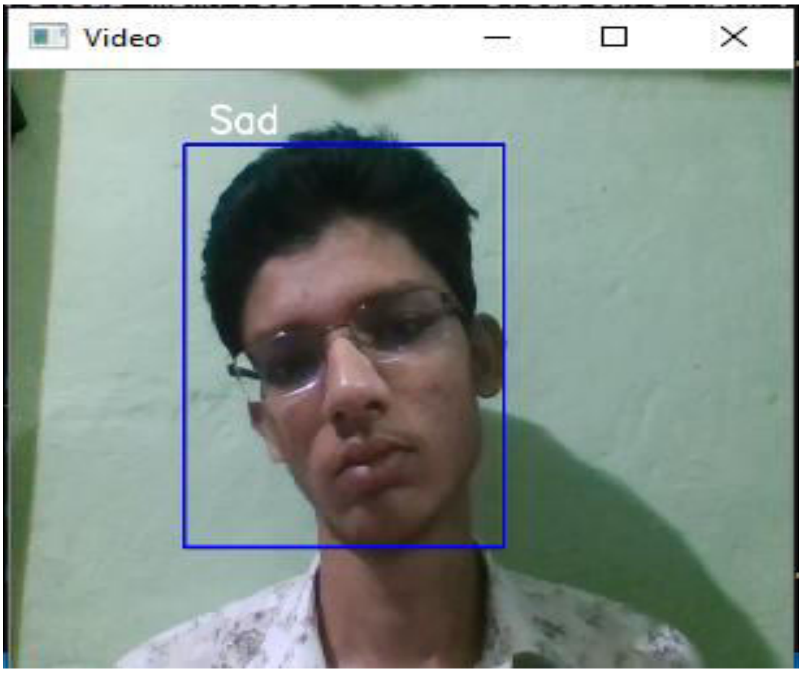
Song Track

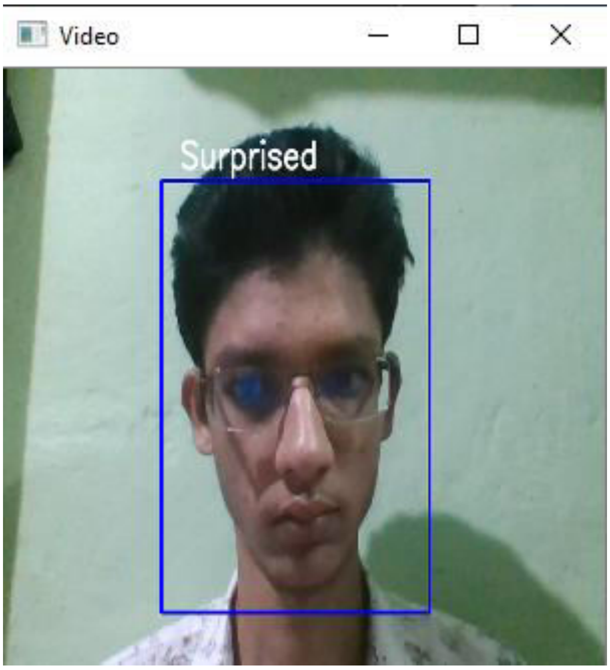
song4.mp3 -Playing Neutral

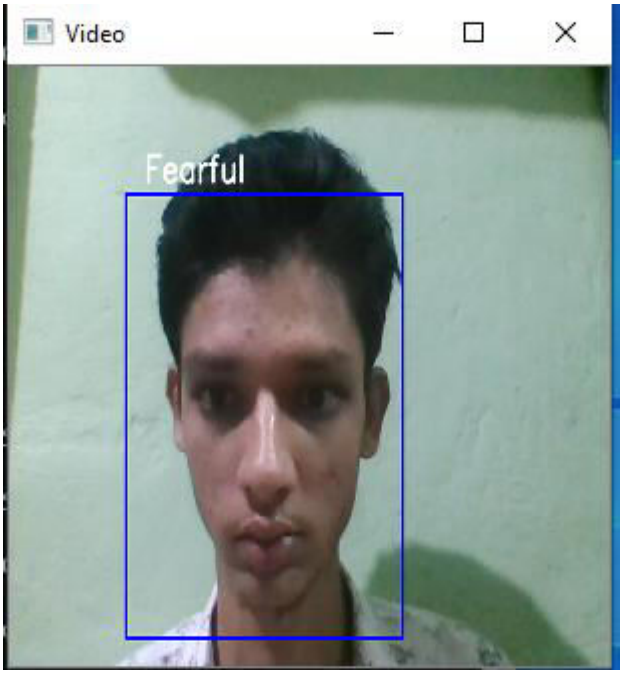
Song Playlist

neutral_song1.mp3
song2.mp3
song3.mp3
song4.mp3
song5.mp3


Video
Happy


Video
Sad


Video
Surprised


Video
Fearful

7. Conclusion and Future Scope

The Emotion Based Music Player is used to automate and give a best music player experience for end user. Application solves all the basic needs of music listeners without troubling them as existing applications do. It uses technology to increase the interaction of the system with the user in numerous ways. It eases the work of user by capturing the image using phone's camera, detecting their emotion and suggesting a customized playlist with advanced features. The user's negative or bad thoughts are slowly converted to positive thoughts by changing the song from low tone to excited tone.

7.1 Future Scope

In future Music Player can be enhanced with Google play music, so songs which are not present in local storage can also be played and to access the whole application in speech based.

Also the future scope in the system would to design a mechanism that would be helpful in music therapy treatment and provide the music therapist the help needed to treat the patients suffering from disorders like mental stress, anxiety, acute depression and trauma. The proposed system also tends to avoid in future the unpredictable results produced in extreme bad light conditions and very poor camera resolution.

8. References

- https://www.udemy.com/course/music-player-using-python-advance-project-for-all-levels/
- https://www.youtube.com/watch?v=5dgNBy2XTV4
- Renuka R. Londhe, Dr. Vrushshen P. Pawar, ―Analysis of Facial Expression and Recognition Based On Statistical Approach‖, International Journal of Soft Computing and Engineering (IJSCE) Volume-2, May 2012.
- https://stackoverflow.com/questions/43370953
- Chang, C. Hu, R. Feris, and M. Turk, ―Manifold based analysis of facial expression,‖ Image Vision Comput ,IEEE Trans. Pattern Anal. Mach. Intell. vol. 24, pp. 05-614, June 2006