Mood-Based Music Player

Project Report

Submitted By:

Shrenika Rajpoot

Reg No: 25BAI11384

B.Tech CSE (AI & ML) – 1st Year

VIT Bhopal University

Submitted To:

Shristika Raikwar

Course: Introduction to Programming & Problem Solving

Abstract

This project presents a Python-based Mood-Based Music Player capable of selecting and playing music according to the user’s emotional state. The program uses simple text-based input, a curated playlist, ASCII animation, and smart random selection to

provide an engaging user experience. The project demonstrates Python fundamentals such as functions, loops, modules, dictionaries, randomization, and user interaction.

Introduction

This project is a simple Python-based music player that lets the user choose their mood and automatically plays songs that match it. It makes listening quick, easy, and mood-friendly.

Problem definition

Music has the power to lift our mood, calm us down, or match exactly how we feel. But when we are stressed, tired, or overwhelmed, searching for the “right song” becomes an extra task. Most music players don’t understand how we feel—they just play whatever we choose. And choosing songs everytime becomes tiring, especially when someone wants quick comfort or a matching vibe. A simple tool is needed that lets users pick their mood directly and instantly get songs that fit that emotion without searching. The project also includes engaging animations and a music visualizer for a creative touch.

Solution

The Mood-based music player provides an easy solution by allowing users to select their mood manually. Once the user chooses a mood button (like happy, sad, energetic or calm),the program automatically plays a preset playlist made for that emotion. Using simple python and basic libraries, the project gives a smooth, personalised music experience.

Objectives

- Create an interactive Python program.
- Implement mood-based music selection.
- Use animations for better user experience.
- Demonstrate understanding of Python modules and functions.
- Build a simple but unique project suitable for first-year coursework.

Requirements

- Python 3.8 or above
- Web browser module (built-in)
- Random module (built-in)
- Time module (built-in)
- Sys module (built-in)
- Iter tools module (built-in)
- Internet connection to play YouTube links

Methodology

1.User selects a mood from predefined categories.

2. The program validates the mood.

3. A loading animation runs to simulate processing.

4. A random song is selected from a playlist mapped to the mood.

5. ASCII visualizer runs to create engagement.

6. The selected YouTube link opens automatically in the browser.

7. User can continue or exit the program.

Program Flow (Algorithm)

1. Display title screen

2. Show list of moods

3. Take mood input from user

4. Validate mood

5. Display loading animation

6. Randomly select a song from the mood’s playlist

7. Show ASCII visualizer

8. Launch song in web browser

9. Ask user if they want to continue

10. Exit message when quitting

Key Features

- Smart mood recognition
- Playlist-based song selection
- Rotating loading animation
- ASCII music visualizer
- YouTube integration
- Simple and clean UI
- Beginner-friendly Python implementation

RESULTS

The Mood-Based Music Player successfully:

- Takes user input for mood selection
- Shows a realistic loading animation
- Displays a dynamic ASCII music visualize
- Opens a randomly selected YouTube song based on mood
- Allows repeated use through a program loop This demonstrates use of Python decision-making, loops, functions, animations, and external linking.

The project executed successfully and met all requirements.

CONCLUSION

This project successfully demonstrates the implementation of a mood-based music system using Python. It showcases logical thinking, modular programming, data handling, animations, and user interaction – that is unique, creative, and academically valuable.

Reference

VITYARTHI course material