Music Player

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology/Master of Technology

In

Computer Science and Engineering

School of Engineering and Sciences

Submitted by

JOYCE PRINCE K---AP21110010358

Y ABHIRAM---AP21110010373

KODURI C S N V SATYANARAYANA---AP21110011199

T NIKHIL---AP21110011200

Under the Guidance of

SUNKARA SURESH BABU

SRM University–AP

Neerukonda, Mangalagiri, Guntur

Andhra Pradesh – 522 240

[DECEMBER, 2022]

Certificate

Date: 15-Dec-22

This is to certify that the work present in this Project entitled "**MUSIC PLAYER**" has been carried out by **[Joyce Prince K, Y Abhiram, Koduri C S N V Satyanarayana, T Nikhil]** under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology/Master of Technology in **School of Engineering and Sciences**.

Supervisor

(Signature)

SUNKARA SURESH BABU

Assistant Professor, CSE Department

SRM UNIVERSITY, AP.

Co-supervisor

(Signature)

SRM UNIVERSITY, AP.

Table of Contents

Abstract

This project is about the mp3 music player. The purpose of this project is to state the goal and objective of the project. The project Presented here is Music Player. The continuous growing of people's music library requires more advanced ways of computing playlists. Several approaches have been made to enhance the user's listening experience, I can't say this is very Unique but yes this project will be a lot easier. The resulting application is a fully functional but feature music player that can also serve as a framework for future playlist generating algorithms. Media usage is changing rapidly these days. This process has been ignited by several technological advances, in particular, the availability of this player is an easy to use based interface where users can listen the music Music which are downloaded in their os.

1. Introduction

Our project title is “MUSIC PLAYER” in this project the user can listen the music which downloaded in their os and can select the songs as pause, forward, backward and he can able to see label(song which is playing). This project is done by Graphical User Interface(GUI).

Music has always been a means of entertaining people even from the earliest ages of the civilization. Historically it was produced by musicians and only available during life concerts. The technological evolution made it possible to save the music on electromagnetic charged stripes, CDs until the technology brought us to saving tracks digitally. When dealing with a huge collection of tracks, people encounter management problems they did not have before. So they have to develop new ways of using the music collection for their entertainment.

Another way, which is given more and more attention that can listen the music which downloaded in the os. The music player allows a user to play MP3 media file formats. It can be used to play audios.

2. Methodology

2.1 Design:

From tkinter import –

Python provides various options for developing Graphical User Interfaces. Tkinter is the standard GUI library for python, python when combined with tkinter provides a fast and easy way to create GUI applications.

Tkinter provides a powerful Object-Oriented-Interface to the Tk GUI tool kit.

Tkinter LIBRARY:

Tkinter is the standard GUI library for Python.

Widgets in Tkinter

Button:

The Button widget is a standard Tkinter widget, which is used for various kinds of buttons. A button is a widget that is designed for the user to interact with, i.e, if the button is pressed by mouse click some action might be started.

Label:

Tkinter Label is a widget that is used to implement display boxes where you can place text or images. The text displayed by this widget can be changed by the developer at any time you want. It is important to note that a label can use only one font at a time to display text.

Attributes of widgets in Tkinter:

➔ Some of the attributes of Tkinter used in our project are as follows:

Bg: The normal background colour displayed behind the label and indicator.

Fg: This option specifies the colour of the text.

font: If you are displaying text in the label, the font option is used to specify in what font that text in the label will be displayed.

height: This option is used to set the vertical dimension of the new frame.

width: Width of the label in characters. If this option is not set, the label will be sized to fit its contents.

anchor: This options is used to control the positioning of the text if the widget has more space than required for the text. The default is anchor=CENTER, which centers the text in the available space.

Import pygame:

Pygame is a Python module that works with computer graphics and sound libraries and is designed with the power of playing with different multimedia formats like audio, video, etc. While creating our Music Player application, we used Pygame's mixer.music module that is usually related to the manipulation of the song tracks.

Import OS:

This module provides different functions for interaction with the Operating System. we used the OS module for fetching the playlist of songs from the specified directory and make it available to the music player.

init Constructor

With the help of this constructor, we will set the title and geometry for the window. We will initiate pygame and pygame mixer and then declare the track variable and status variable.

- We can Create the label, play, pause, unpause, forward and backward buttons into it as well as listbox to display the songs list.

def select():

To select the song in the given list box. After selection the user want to press play button to play song.

def pause():

To pause or stop the song the user want to press pause button.

def unpause():

The song will continue the playing where the song was paused, previously. For that the user want to press unpause button

def play_prev():

For this function if user press prev button the song will shift to the previous song.

def forward_next():

If user press forward_next button the song will shift to the next song in the list.

2.2 Implimentation:

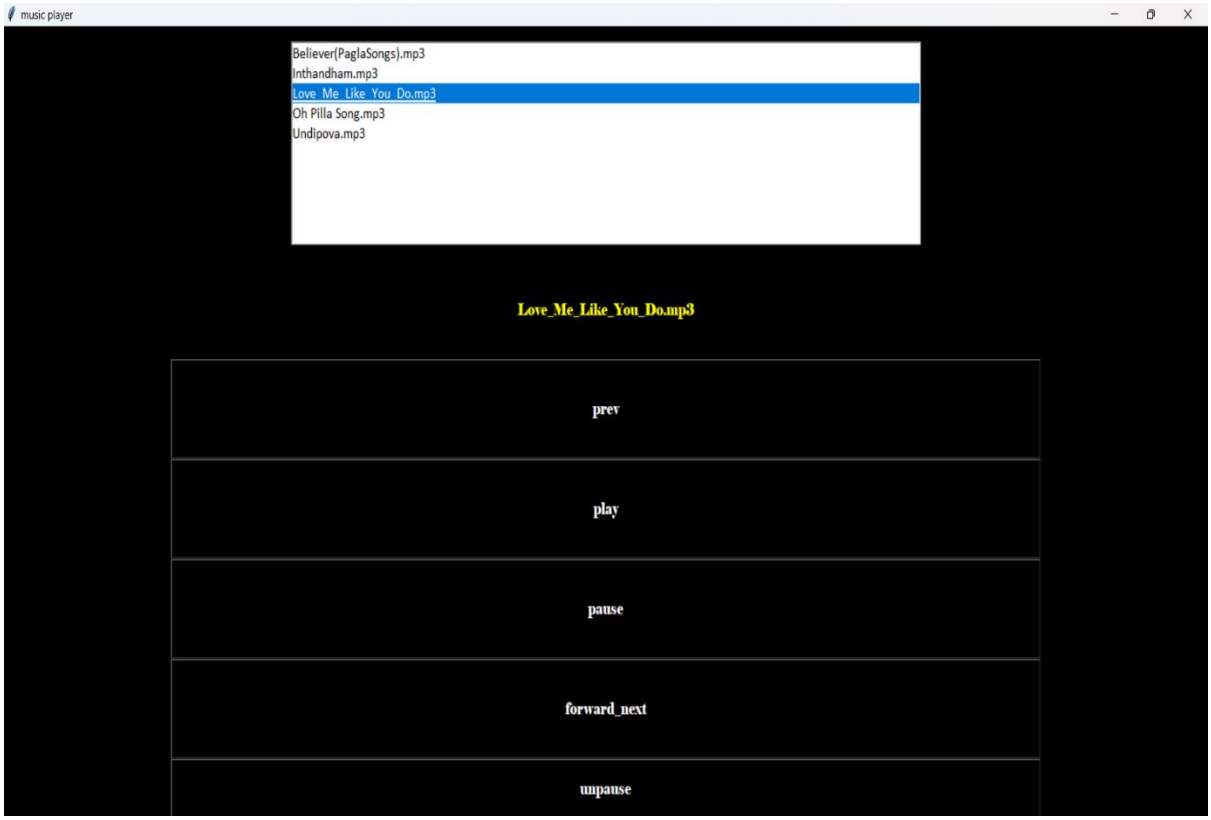
Code:

```
import tkinter as tkr
import fnmatch
import os
from pygame import mixer

music=tkr.Tk()
music.title("music player")
music.geometry("600x900")
music.config(bg= 'black')

roothpath = "D:\\CHAIT\mpyth"
pattern="*.mp3"

mixer.init()

def select():
    label.config(text = l_Box.get("anchor"))
    mixer.music.load(roothpath + "\\" + l_Box.get("anchor"))
    mixer.music.play()

def pause():
    mixer.music.pause()

def unpause():
    mixer.music.unpause()

def play_prev():
    next_song = l_Box.curselection()
    next_song = next_song[0] - 1
    next_song_name = l_Box.get(next_song)
    label.config(text = next_song_name)

    mixer.music.load(roothpath + "\\" + next_song_name)
    mixer.music.play()

    l_Box.select_clear(0, 'end')
    l_Box.activate(next_song)
    l_Box.select_set(next_song)
```

```

def forward_next():
    next_song = l_Box.curselection()
    next_song = next_song[0] + 1
    next_song_name = l_Box.get(next_song)
    label.config(text = next_song_name)

    mixer.music.load(roothpath + "\\" + next_song_name)
    mixer.music.play()

    l_Box.select_clear(0, 'end')
    l_Box.activate(next_song)
    l_Box.select_set(next_song)

l_Box=tkr.Listbox(music,fg="black",bg="white",width=100,font="Calibri")
l_Box.pack(padx = 15, pady = 15)

label=tkr.Label(music,text='',fg="yellow",bg="black",width=100,height=3,font="Elephant")
label.pack(pady = 15)

b1=tkr.Button(music,width=100,height=4,font="Elephant",text="prev",command=play_prev,bg="black",fg="white").pack()
b2=tkr.Button(music,width=100,height=4,font="Elephant",text="play",bg="black",command=select,fg="white").pack()
b3=tkr.Button(music,width=100,height=4,font="Elephant",text="pause",bg="black",command=pause,fg="white").pack()
b4=tkr.Button(music,width=100,height=4,font="Elephant",text="forward_next",bg="black",command=forward_next,fg="white").pack()
b5=tkr.Button(music,width=100,height=4,font="Elephant",text="unpause",bg="black",command=unpause,fg="white").pack()

for root, dirs, files in os.walk(roothpath):
    for filename in fnmatch.filter(files,pattern):
        l_Box.insert('end', filename)

music.mainloop()
```

3. Discussion

Outputs:



4. Conclusion

- Here we completed with our GUI project using Tkinter and pygame.
- With these steps we have successfully created a music player project using python.
- We learned how to create listbox, label and buttons.
- In this way we successfully created our music player python project.

References

- ➔ https://docs.python.org/3/library/tkinter.html
- ➔ https://www.tutorialspoint.com/python3/python_gui_programming.htm