Table of Index

|  |  |  |
| --- | --- | --- |
| S NO. | CONTENTS | PAGE NO. |
| 1. | INTRODUCTION | 2 |
| 2. | PROJECT DESCRIPTION | 3 |
| 3. | PROJECT REQUIREMENTS | 4 |
| 4. | PYTHON CONCEPTS | 5 |
| 5. | USER MANUAL | 8 |
| 6. | FLOW CHART | 9 |
| 7. | PROGRAM | 14 |
| 8. | CONCLUSION | 38 |
| 9. | BIBLIOGRAPHY | 39 |

**INTRODUCTION**

We need an application that will allow us to play or listen to digital audio files. MP3 player is the device to play MP3s and other digital audio files. The MP3 GUI program application attempts to emulate the physical MP3 Player. This program will allow you to play songs, music, and all MP3 files on your desktop or laptops. The main objective of this project is to allow users to play MP3 and digital audio files. To be engaging for users, the application has to have a simple but beautiful user interface.

This GUI project is developed using Python programming language. The GUI aspect of the application is built using the Tkinter library of Python. The interactive part of the application that handles the MP3 files uses the Pygame and Mutagen libraries.

The users will also expect the MP3 Player to have an interface that shows information on the file that is playing. Some of the information you can include are the name of the file, its length, the amount played, and the amount not played, in minutes and seconds.

Python has libraries that can play audio files, such as Pygame, which allows you to work with multimedia files in few lines of code. Similar libraries are Pymedia and Simpleaudio. These libraries can handle a lot of digital audio files. They can handle other file types, not just the MP3 files.

You can also implement a feature that allows users to create a playlist. To do this, you’ll need a database to store information on the created playlists. Python’s Mysql.connector module allows you to use the mysql database.The mysql database is a better option in this case, because it is file based and easier to set up than other mysql databases. While mysql.connector is file based, it is better for saving data than a regular file.

**Project description:**

1. To build an MP3 player using Python programming language to be able to play and listen to songs, MP3 files.

2. Determine the functionalities of the MP3 player.

3. The player should be have a simple and easy to use GUI with options for various functions, display screen to display the entire playlist and buttons to shut down the player.

4. The player should be able to play any song. It should be capable of playing MP3 files

5. The player should allow the user to browse through the contents of the computer drive to choose song/s to be played or queued.

6. It should provide the user with option to pause or resume the song.

7. The user should be able to play the previous or the next song in the playlist.

8. Lastly, the user should get basic details about the current playing song. The details can include the duration of the song, size of the file, etc.

**Project requirements**

***Software:***

Python 3.7

MySQL

Windows

***Hardware:***

Speakers

System

System typed 64-based PC

Processor intel Pentium cpu

Smbios 3.7

Ram 2.00 mb

Total physical memory 1.89 gb

Available physical memory 3.78 gb

Available virtual memory 2.83 gb

**Project concepts:**

* **String:**

A string in Python is a sequence of characters. It is a derived data type. Strings are immutable. This means that once defined, they cannot be changed.

* **Tuple:**

Tuples are used to store multiple items in a single variable. Tuple is one of 4 built-in data types in Python used to store collections of data, the other 3 are List, Set, and Dictionary, all with different qualities and usage.

* **Sequence:**

In Python, sequence is the generic term for an ordered set. Lists are the most versatile sequence type. ... The elements of a list can be any object, and lists are mutable - they can be changed. Elements can be reassigned or removed, and new elements can be inserted.

* **Module imported:**
  + - tkinter
    - mysql.connector
    - tkinter.messagebox
    - from tkinter filedialog
    - pygame
    - from pygame mixer
    - os
    - time
    - random
    - threading
    - from mutagen.mp3 MP3
    - ttk, ThemedTk
* **Tkinter:**

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 toolkit.

Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps –

* Import the Tkinter module.
* Create the GUI application main window.
* Add one or more of the above-mentioned widgets to the GUI application.
* Enter the main event loop to take action against each event triggered by the user.

Tkinter provides various controls, such as buttons, labels and text boxes used in a GUI application. These controls are commonly called widgets.

* **Mysql.connector:**

MySQL Connector/Python enables Python programs to access MySQL databases, using an API that is compliant with the Python Database API Specification v2. 0 (PEP 249). MySQL Connector/Python includes support for: Almost all features provided by MySQL Server up to and including MySQL Server version 8.0.

MySQL Connectors provide connectivity to the MySQL server for client programs. APIs provide low-level access to MySQL resources using either the classic MySQL protocol or X Protocol.

***Advantage of mysql:***

* Data Security
* On-Demand Scalability
* High Performance
* Round-the-clock Uptime , ETC .
* **Tkinter.Messagebox:**

The messagebox module is used to display the message boxes in the python applications. There are the various functions which are used to display the relevant messages depending upon the application requirements.

* **tkinter import filedialog:**

Python Tkinter (and TK) offer a set of dialogs that you can use when working with files. By using these you don't have to design standard dialogs your self.

* **Pygame:**

pygame is a Python wrapper for the SDL library, which stands for Simple DirectMedia Layer. SDL provides cross-platform access to your system's underlying multimedia hardware components, such as sound, video, mouse, keyboard, and joystick.

* **Os:**

The OS module in Python provides functions for interacting with the operating system. OS comes under Python's standard utility modules.

* **Time:**

time() The time() function returns the number of seconds passed since epoch.

* **Random:**

random() is an inbuilt function of the random module in Python3. The random module gives access to various useful functions and one of them being able to generate random floating numbers, which is random (). ... Returns this method returns a random floating number between 0 and 1.

* **Threading:**

Threadinginpython is used to run multiple threads (tasks, function calls) at the same time. Pythonthreads are used in cases where the execution of a task involves some waiting.

* **Mutagen.mp3 import MP3:**

Mutagen is a Python module to handle audio metadata. It supports ASF, FLAC, MP4, Monkey's Audio, MP3, Musepack, Ogg Opus, Ogg FLAC, Ogg Speex, Ogg Theora, Ogg Vorbis, True Audio, WavPack, OptimFROG, and AIFF audio files. All versions of ID3v2 are supported, and all standard ID3v2.

* **Ttk and themedtk:**

Tk themed widgets. ttk module provides access to the Tk themed widget set, introduced in Tk 8.5 .If Python has not been compiled against Tk 8.5, this module can still be accessed if Tile has been installed.

**User manual:**

* + Choice of songs user wants to hear.
  + Clicking play, pause, stop buttons.

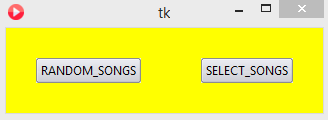
***Flow chat:***

**OUTPUTS**

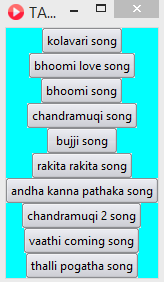
****

*CHOICE OF LANGUAGE \MUSIC FROM SYSTEM*

*****PLAYING SONG FROM SYSTEM*

****

*CHOICE OF METHOD*

**** *CHOICE OF SONGS*



*PLAY\PAUSE\STOP\CONTROL VOLUME*

**FUNCTIONS**

* ***play ():***

Audio is played and if music is paused , it will break the pause function and play the audio.

* ***stop():***

Audio is terminated.

* ***pause():***

Audio is paused

* ***Total\_length():***

It displays the total length of audio file.

* ***Current\_time():***

It displays the running time in -1 format.

* ***Sound\_control():***

It controls the volume. Initially the volume will be set to 70.

* ***delete():***

It will delete audio file from label box.

* ***fetch():***

It will fetch only one mp3 file from mysql database

* ***start\_count():***

It will start the timing of the song when it is playing.

* ***add\_to\_playlist():***

it will choose mp3 file from file explorer box into the label box.

* ***Music\_page():***

Contain play, pause, stop, music file length, etc buttons.

**PROGRAM:**

from tkinter import \*

import mysql.connector as sql

import tkinter.messagebox

from tkinter import filedialog

import pygame

from pygame import mixer

import os

import time

import random

import threading

from mutagen.mp3 import MP3

from tkinter import ttk

from ttkthemes import ThemedTk

mixer.init()

mycon = sql.connect(host="localhost",user="root",password="ricemms",database="songs")

cursor = mycon.cursor()

def fetch():

global choose\_music

choose\_music = cursor.fetchone()

def hin\_1():

cursor.execute("select song\_name from songs where sno=1")

fetch()

def hin\_2():

cursor.execute("select song\_name from songs where sno=2")

fetch()

def hin\_3():

cursor.execute("select song\_name from songs where sno=3")

fetch()

def hin\_4():

cursor.execute("select song\_name from songs where sno=4")

fetch()

def hin\_5():

cursor.execute("select song\_name from songs where sno=5")

fetch()

def hin\_6():

cursor.execute("select song\_name from songs where sno=6")

fetch()

def hin\_7():

cursor.execute("select song\_name from songs where sno=7")

fetch()

def hin\_8():

cursor.execute("select song\_name from songs where sno=8")

fetch()

def hin\_9():

cursor.execute("select song\_name from songs where sno=9")

fetch()

def hin\_10():

cursor.execute("select song\_name from songs where sno=10")

fetch()

def hin\_11():

cursor.execute("select song\_name from songs where sno=11")

fetch()

def hin\_12():

cursor.execute("select song\_name from songs where sno=12")

fetch()

def tam\_1():

cursor.execute("select song\_name from songs where sno=101")

fetch()

def tam\_2():

cursor.execute("select song\_name from songs where sno=102")

fetch()

def tam\_3():

cursor.execute("select song\_name from songs where sno=103")

fetch()

def tam\_4():

cursor.execute("select song\_name from songs where sno=104")

fetch()

def tam\_5():

cursor.execute("select song\_name from songs where sno=105")

fetch()

def tam\_6():

cursor.execute("select song\_name from songs where sno=106")

fetch()

def tam\_7():

cursor.execute("select song\_name from songs where sno=107")

fetch()

def tam\_8():

cursor.execute("select song\_name from songs where sno=108")

fetch()

def tam\_9():

cursor.execute("select song\_name from songs where sno=109")

fetch()

def tam\_10():

cursor.execute("select song\_name from songs where sno=110")

fetch()

def eng\_1():

cursor.execute("select song\_name from songs where sno=201")

fetch()

def eng\_2():

cursor.execute("select song\_name from songs where sno=202")

fetch()

def eng\_3():

cursor.execute("select song\_name from songs where sno=203")

fetch()

def eng\_4():

cursor.execute("select song\_name from songs where sno=204")

fetch()

def eng\_5():

cursor.execute("select song\_name from songs where sno=205")

fetch()

def eng\_6():

cursor.execute("select song\_name from songs where sno=206")

fetch()

def eng\_7():

cursor.execute("select song\_name from songs where sno=207")

fetch()

def eng\_8():

cursor.execute("select song\_name from songs where sno=208")

fetch()

def eng\_9():

cursor.execute("select song\_name from songs where sno=209")

fetch()

def eng\_10():

cursor.execute("select song\_name from songs where sno=210")

fetch()

def hin\_random\_music():

global \_songs

a = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Tumse Milke Dilka Jo Haal [Full Song]Main Hoon NaShahrukh Khan.mp3')

b = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\ARIJIT SINGH VERSION Bekhayali (LYRICAL)Kabir SinghShahid K,Kiara ASandeep Reddy V Irshad.mp3')

c = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Agar Tum Saath Ho - Full Song - ALKA YAGNIK and ARIJIT SINGH.mp3')

d = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Channa Mereya - Lyric VideoAe Dil Hai MushkilKaran JoharRanbirAnushkaPritamArijit.mp3')

e = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Full Song KHAIRIYAT (BONUS TRACK) CHHICHHORE Sushant, Shraddha Pritam, Amitabh BArijit S.mp3')

f = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Jo Bheji Thi Dua Arijit Singh Full Song (Lyrics) Nandini Srikar.mp3')

g = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\LUDO Aabaad Barbaad (Full VIdeo) Abhishek B, Aditya K, Rajkummar R, Sanya, Fatima Arijit, Pri.mp3')

h = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Lyrical Tum Hi Aana Marjaavaan Riteish D, Sidharth M, Tara S Jubin Nautiyal,Payal Dev,Kuna.mp3')

i = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Matkar Maya Ko Ahankar.mp3')

j = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Mere Rashke Qamar Song With LyricsBaadshahoAjay Devgn, Ileana, Nusrat & Rahat Fateh Ali Khan.mp3')

k = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Rakta Charitra- Mila To Marega(Warrior Version With English Subtitles).mp3')

l = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi songs\Titliaan Harrdy Sandhu Sargun Mehta Afsana Khan Jaani Avvy Sra Arvindr Khaira.mp3')

\_songs = [a,b,c,d,e,f,g,h,i,j,k,l]

def tam\_random\_music():

global \_songs

a = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\3 - Why This Kolaveri Di Official Video Dhanush, Anirudh.mp3')

b = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Bhoomi - Kadai Kannaaley Video Jayam Ravi, Nidhhi Agerwal D. Imman.mp3')

c = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Bhoomi - Tamizhan Endru Sollada Video Jayam Ravi, Nidhhi Agerwal D. Imman.mp3')

d = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Chandramukhi Tamil Movie SongsDevuda Devuda SongRajinikanthJyothikaNayantaraPrabhu.mp3')

e = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Jagame Thandhiram - Bujji Video Dhanush Santhosh Narayanan Karthik Subbaraj Anirudh.mp3')

f = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Master - Andha Kanna Paathaakaa Lyric Thalapathy Vijay Anirudh Ravichander Lokesh Kanagara.mp3')

g = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Master - Vaathi Coming Video Thalapathy Vijay Anirudh Ravichander Lokesh Kanagaraj.mp3')

h = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Rajinikanth Tamil Hits 2017 Chandramukhi Songs Athinthom Video Song Rajinikanth Nayantha.mp3')

i = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Thalli Pogathey - Official Single Achcham Yenbadhu Madamaiyada A R Rahman Lyric Video.mp3')

j = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil songs\Jagame Thandhiram - Rakita Rakita Rakita LyricDhanushSanthosh NarayananKarthik Subbaraj.mp3')

\_songs = [a,b,c,d,e,f,g,h,i,j]

def eng\_random\_music():

global \_songs

a = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\Bon Appétit.mp3')

b = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\GANGNAM STYLE.mp3')

c = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\GENTLEMAN .mp3')

d = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\Imagine Dragons - Believer.mp3')

e = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\Master - Master the Blaster Lyric Thalapathy Vijay AnirudhRavichander LokeshKanagaraj.mp3')

f = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\PSY - DADDY(feat. CL of 2NE1) MV.mp3')

g = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\PSY - New Face MV.mp3')

h = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\Shape of You.mp3')

i = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\Thanos Sings A Song (Marvel Avengers Infinity War Parody).mp3')

j = (r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english songs\The Chainsmokers - Closer (Lyric) ft. Halsey.mp3')

\_songs = [a,b,c,d,e,f,g,h,i,j]

def \_music\_():

global paused

\_currently\_playing\_song = None

next\_song = random.choice(\_songs)

element = \_songs.index(next\_song)

deleted\_song = \_songs.pop(element)

while (next\_song == \_currently\_playing\_song):

next\_song = random.choice(\_songs)

\_currently\_playing\_song = next\_song

if (paused):

mixer.music.unpause()

paused = False

else:

pygame.mixer.music.load(next\_song)

pygame.mixer.music.play()

show\_details\_random(next\_song)

def show\_details\_random(play\_song):

file\_data = os.path.splitext(play\_song)

audio = MP3(play\_song)

total\_length = audio.info.length

mins, secs = divmod(total\_length, 60)

mins = round(mins)

secs = round(secs)

timeformat = '{:02d}:{:02d}'.format(mins, secs)

ran\_lengthlabel['text'] = 'TOTAL LENGTH -' + timeformat

threading.Thread(target=start\_count, args=(total\_length,)).start()

def show\_details\_other(play\_song):

file\_data = os.path.splitext(play\_song)

audio = MP3(play\_song)

total\_length = audio.info.length

mins, secs = divmod(total\_length, 60)

mins = round(mins)

secs = round(secs)

timeformat = '{:02d}:{:02d}'.format(mins, secs)

lengthlabel['text'] = 'TOTAL LENGTH -' + timeformat

threading.Thread(target=start\_count, args=(total\_length,)).start()

def show\_details(play\_song):

global total\_length

audio = MP3(play\_song)

total\_length = audio.info.length

mins, secs = divmod(total\_length, 60)

mins = round(mins)

secs = round(secs)

timeformat = '{:02d}:{:02d}'.format(mins, secs)

lengthlabel['text'] = 'TOTAL LENGTH -' + timeformat

threading.Thread(target=start\_count, args=(total\_length,)).start()

def start\_count(t):

global paused

while t and mixer.music.get\_busy():

if (paused):

continue

else:

mins, secs = divmod(t, 60)

mins = round(mins)

secs = round(secs)

timeformat = '{:02d}:{:02d}'.format(mins, secs)

currenttimelabel['text'] = 'CURRENT TIME -' + timeformat

time.sleep(1)

t = t - 1

def play\_music():

global paused

if (paused):

mixer.music.unpause()

paused = False

else:

music = choose\_music[0]

mixer.music.load(music)

mixer.music.play()

show\_details(music)

musicpage.mainloop()

def play\_other\_music():

global paused

if (paused):

mixer.music.unpause()

paused = False

else:

stop\_music()

time.sleep(1)

selected\_song = playlistbox.curselection()

selected\_song = int(selected\_song[0])

play\_it = playlist[selected\_song]

mixer.music.load(play\_it)

mixer.music.play()

show\_details\_other(play\_it)

def stop\_music():

global stop

stop = True

mixer.music.stop()

paused = False

def pause\_music():

global paused

paused = True

mixer.music.pause()

def sound\_control(val):

soundcotrol = True

volume = float(val) / 100

mixer.music.set\_volume(volume)

def project():

tkinter.messagebox.showinfo('-----DESCRIPTION-----', ''' ---IT'S A MUSIC ALBUM WITH MUSIC PLAYER---

-------ONLY MP3 FILES-------

------CONTAIN 10 HINDI,TAMIL,ENGLISH SONGS--------

-------YOUR OWN FILES CAN ALSO BE PLAYED-------''')

def about\_us():

tkinter.messagebox.showinfo('-----MUSIC ALBUM-----', """ ----------PythonTkinter--WITH--MySql----------

by ----------SACHIN-PRAVIN-BALA----------""")

playlist = []

def choose\_song():

global filename

filename = filedialog.askopenfilename()

add\_to\_playlist(filename)

def del\_song():

selected\_song = playlistbox.curselection()

selected\_song = int(selected\_song[0])

playlistbox.delete(selected\_song)

playlist.pop(selected\_song)

def add\_to\_playlist(f):

f = os.path.basename(f)

index = 0

playlistbox.insert(index, f)

playlist.insert(index, filename)

index += 1

def on\_closing():

stop\_music()

root.destroy()

def while\_closing():

stop\_music()

frame.destroy()

def random\_page():

global ran\_lengthlabel

global currenttimelabel

global play\_photo,play\_button

ran\_page = ThemedTk(theme="plastik")

ran\_page['bg']='magenta2'

topframe = Frame(ran\_page)

topframe['bg']='magenta2'

topframe.pack(padx=15, side=TOP, pady=20)

bottomframe = Frame(ran\_page)

bottomframe['bg']='magenta2'

bottomframe.pack(padx=15, side=BOTTOM, pady=20)

rightframe = Frame(bottomframe)

rightframe['bg']='magenta2'

rightframe.pack(padx=15, side=LEFT, pady=30)

leftframe = Frame(bottomframe)

leftframe['bg']='magenta2'

leftframe.pack(padx=15, side=RIGHT, pady=30)

play\_button = ttk.Button(topframe, text='PLAY', command=\_music\_)

play\_button.grid(row=0, column=0, padx=30)

stop\_button = ttk.Button(topframe, text='STOP', command=stop\_music)

stop\_button.grid(row=0, column=1, padx=30)

pause\_button = ttk.Button(topframe, text='PAUSE', command=pause\_music)

pause\_button.grid(row=0, column=2, padx=30)

currenttimelabel = ttk.Label(leftframe,text='CURRENT TIME - --:--',relief=GROOVE,font='Times 10 italic')

currenttimelabel.grid(pady=10)

ran\_lengthlabel = ttk.Label(leftframe,text='TOTAL LENGTH - --:--',relief=GROOVE,font='Times 10 italic')

ran\_lengthlabel.grid(pady=5)

volume\_control = ttk.Scale(rightframe, from\_=0, to\_=100, orient=HORIZONTAL, command=sound\_control)

volume\_control.set(70)

mixer.music.set\_volume(0.7)

volume\_control.grid(row=4, column=2, padx=20)

ran\_page.mainloop()

def music\_page():

global musicpage

global lengthlabel

global currenttimelabel

musicpage = ThemedTk(theme="plastik")

musicpage['bg']='lime'

musicpage.title('WELCOME TO MUSIC PLAYER')

musicpage.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

topframe = Frame(musicpage)

topframe['bg']='lime'

topframe.pack(padx=15, side=TOP, pady=20)

bottomframe = Frame(musicpage)

bottomframe['bg']='lime'

bottomframe.pack(padx=15, side=BOTTOM, pady=20)

rightframe = Frame(bottomframe)

rightframe['bg']='lime'

rightframe.pack(padx=15, side=LEFT, pady=30)

leftframe = Frame(bottomframe)

leftframe['bg']='lime'

leftframe.pack(padx=15, side=RIGHT, pady=30)

play\_button = ttk.Button(topframe, text='PLAY', command=play\_music)

play\_button.grid(row=0, column=0, padx=30)

stop\_button = ttk.Button(topframe, text='STOP', command=stop\_music)

stop\_button.grid(row=0, column=1, padx=30)

pause\_button = ttk.Button(topframe, text='PAUSE', command=pause\_music)

pause\_button.grid(row=0, column=2, padx=30)

lengthlabel = ttk.Label(rightframe, text='TOTAL LENGTH - --:--', relief=GROOVE, font='Times 10 italic')

lengthlabel.grid(pady=5)

currenttimelabel = ttk.Label(rightframe, text='CURRENT TIME - --:--', relief=GROOVE, font='Times 10 italic')

currenttimelabel.grid(pady=10)

volume\_control = ttk.Scale(leftframe, from\_=0, to\_=100, orient=HORIZONTAL, command=sound\_control)

volume\_control.set(70)

mixer.music.set\_volume(0.7)

volume\_control.grid(row=4, column=3, padx=20)

def hinmusic():

hinpage = ThemedTk(theme="plastik")

hinpage['bg']='cyan'

hinpage.title('HINDI MUSICS')

hinpage.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

h\_1 = ttk.Button(hinpage, text='palbhar sambhal jau',command=lambda: [music\_page(), hin\_1()])

h\_1.pack()

h\_2 = ttk.Button(hinpage, text='bekhayali', command=lambda: [music\_page(), hin\_2()])

h\_2.pack()

h\_3 = ttk.Button(hinpage, text='acha chalta hum', command=lambda: [music\_page(), hin\_3()])

h\_3.pack()

h\_4 = ttk.Button(hinpage, text='khariyat ', command=lambda: [music\_page(), hin\_4()])

h\_4.pack()

h\_5 = ttk.Button(hinpage, text='jo bheje thi dua', command=lambda: [music\_page(), hin\_5()])

h\_5.pack()

h\_6 = ttk.Button(hinpage, text='abaad barbaad', command=lambda: [music\_page(), hin\_6()])

h\_6.pack()

h\_7 = ttk.Button(hinpage, text='Tum Hi Aana', command=lambda: [music\_page(), hin\_7()])

h\_7.pack()

h\_8 = ttk.Button(hinpage, text='mat kar maya ko', command=lambda: [music\_page(), hin\_8()])

h\_8.pack()

h\_9 = ttk.Button(hinpage, text='tare laska kamal', command=lambda: [music\_page(), hin\_9()])

h\_9.pack()

h\_10 = ttk.Button(hinpage, text='rakht charitra', command=lambda: [music\_page(), hin\_10()])

h\_10.pack()

h\_11 = ttk.Button(hinpage, text='titliyaan varga', command=lambda: [music\_page(), hin\_11()])

h\_11.pack()

h\_12 = ttk.Button(hinpage, text='tum se mil ke', command=lambda: [music\_page(), hin\_12()])

h\_12.pack()

def tammusic():

tampage = ThemedTk(theme="plastik")

tampage['bg']='cyan'

tampage.title('TAMIL MUSICS')

tampage.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

t\_1 = ttk.Button(tampage, text='kolavari song', command=lambda: [music\_page(), tam\_1()])

t\_1.pack()

t\_2 = ttk.Button(tampage, text='bhoomi love song', command=lambda: [music\_page(), tam\_2()])

t\_2.pack()

t\_3 = ttk.Button(tampage, text='bhoomi song', command=lambda: [music\_page(), tam\_3()])

t\_3.pack()

t\_4 = ttk.Button(tampage, text='chandramuqi song', command=lambda: [music\_page(), tam\_4()])

t\_4.pack()

t\_5 = ttk.Button(tampage, text='bujji song', command=lambda: [music\_page(), tam\_5()])

t\_5.pack()

t\_6 = ttk.Button(tampage, text='rakita rakita song', command=lambda: [music\_page(), tam\_6()])

t\_6.pack()

t\_7 = ttk.Button(tampage, text='andha kanna pathaka song', command=lambda: [music\_page(), tam\_7()])

t\_7.pack()

t\_8 = ttk.Button(tampage, text='chandramuqi 2 song', command=lambda: [music\_page(), tam\_8()])

t\_8.pack()

t\_9 = ttk.Button(tampage, text='vaathi coming song', command=lambda: [music\_page(), tam\_9()])

t\_9.pack()

t\_10 = ttk.Button(tampage, text='thalli pogatha song', command=lambda: [music\_page(), tam\_10()])

t\_10.pack()

def engmusic():

engpage = ThemedTk(theme="plastik")

engpage['bg']='cyan'

engpage.title('ENGLISH MUSICS')

engpage.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

e\_1 = ttk.Button(engpage, text='1st\_song', command=lambda: [music\_page(), eng\_1()])

e\_1.pack()

e\_2 = ttk.Button(engpage, text='2nd\_song', command=lambda: [music\_page(), eng\_2()])

e\_2.pack()

e\_3 = ttk.Button(engpage, text='3rd\_song', command=lambda: [music\_page(), eng\_3()])

e\_3.pack()

e\_4 = ttk.Button(engpage, text='1st\_song', command=lambda: [music\_page(), eng\_4()])

e\_4.pack()

e\_5 = ttk.Button(engpage, text='1st\_song', command=lambda: [music\_page(), eng\_5()])

e\_5.pack()

e\_6 = ttk.Button(engpage, text='1st\_song', command=lambda: [music\_page(), eng\_6()])

e\_6.pack()

e\_7 = ttk.Button(engpage, text='1st\_song', command=lambda: [music\_page(), eng\_7()])

e\_7.pack()

e\_8 = ttk.Button(engpage, text='1st\_song', command=lambda: [music\_page(), eng\_8()])

e\_8.pack()

e\_9 = ttk.Button(engpage, text='1st\_song', command=lambda: [music\_page(), eng\_9()])

e\_9.pack()

e\_10 = ttk.Button(engpage, text='1st\_song', command=lambda: [music\_page(), eng\_10()])

e\_10.pack()

def hindi\_page():

hin\_page = ThemedTk(theme="plastik")

hin\_page['bg']='yellow'

hin\_page.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

random\_button = ttk.Button(hin\_page, text='RANDOM\_SONGS', command=lambda:[hin\_random\_music(),random\_page()])

random\_button.grid(row=0, column=0, padx=30, pady=30)

music\_button = ttk.Button(hin\_page, text='SELECT\_SONGS', command=hinmusic)

music\_button.grid(row=0, column=1, padx=30, pady=30)

hin\_page.mainloop()

def tamil\_page():

tam\_page = ThemedTk(theme="plastik")

tam\_page['bg']='yellow'

tam\_page.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

random\_button = ttk.Button(tam\_page, text='RANDOM\_SONGS', command=lambda:[tam\_random\_music(),random\_page()])

random\_button.grid(row=0, column=0, padx=30, pady=30)

music\_button = ttk.Button(tam\_page, text='SELECT\_SONGS', command=tammusic)

music\_button.grid(row=0, column=1, padx=30, pady=30)

tam\_page.mainloop()

def english\_page():

eng\_page = ThemedTk(theme="plastik")

eng\_page['bg']='yellow'

eng\_page.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

random\_button = ttk.Button(eng\_page, text='RANDOM\_SONGS', command=lambda:[eng\_random\_music(),random\_page()])

random\_button.grid(row=0, column=0, padx=30, pady=30)

music\_button = ttk.Button(eng\_page, text='SELECT\_SONGS', command=engmusic)

music\_button.grid(row=0, column=1, padx=30, pady=30)

eng\_page.mainloop()

def othermusic():

global frame

global playlistbox

global lengthlabel

global currenttimelabel

frame = ThemedTk(theme="plastik")

frame.title('----MUSIC--PLAYER----')

frame['bg']='purple2'

frame.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

rightframe = Frame(frame)

rightframe['bg']='purple2'

rightframe.pack(padx=15, side=RIGHT, pady=30)

topframe = Frame(rightframe)

topframe['bg']='purple2'

topframe.pack()

bottomframe = Frame(rightframe)

bottomframe['bg']='purple2'

bottomframe.pack(padx=20, pady=20)

bottomleftframe = Frame(bottomframe)

bottomleftframe['bg']='purple2'

bottomleftframe.pack(side=LEFT, padx=30)

leftframe = Frame(frame)

leftframe['bg']='purple2'

leftframe.pack(side=LEFT, padx=30, pady=30)

playlistbox = Listbox(leftframe)

playlistbox.pack()

add\_btn = ttk.Button(leftframe, text='ADD', command=choose\_song)

add\_btn.pack(side=LEFT)

del\_btn = ttk.Button(leftframe, text='DELETE', command=del\_song)

del\_btn.pack(side=LEFT)

play\_button = ttk.Button(topframe, command=play\_other\_music, text='PLAY')

play\_button.grid(row=0, column=0, padx=20)

stop\_button = ttk.Button(topframe, command=stop\_music, text='STOP')

stop\_button.grid(row=0, column=1, padx=20)

pause\_button = ttk.Button(topframe, command=pause\_music, text='PAUSE')

pause\_button.grid(row=0, column=2, padx=20)

lengthlabel = ttk.Label(bottomframe, text='TOTAL LENGTH - --:--', relief=GROOVE, font='Times 10 italic')

lengthlabel.pack(pady=10)

currenttimelabel = ttk.Label(bottomframe, text='CURRENT TIME - --:--', relief=GROOVE, font='Times 10 italic')

currenttimelabel.pack(pady=10)

volume\_control = ttk.Scale(bottomleftframe, from\_=0, to\_=100, orient=HORIZONTAL, command=sound\_control)

volume\_control.set(70)

mixer.music.set\_volume(0.7)

volume\_control.pack(pady=10)

menu\_bar = Menu(frame)

frame.config(menu=menu\_bar)

sub\_menu = Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label='FILE', menu=sub\_menu)

sub\_menu.add\_command(label="PROJECT DESCRIPTION", command=project)

sub\_menu.add\_command(label="ABOUT US", command=about\_us)

frame.protocol('WM\_DELETE\_WINDOW', while\_closing)

frame.mainloop()

root = ThemedTk(theme="plastik")

root.title('MUSIC FILE')

root.geometry('1200x600')

root['bg']='dark orange'

root.iconbitmap(r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\play\_fv0\_icon.ico')

label = ttk.Label(root,text='M-U-S-I-C P-L-A-Y-E-R',relief=RIDGE,anchor=CENTER,font="Arial 30")

label.place(x=2, y=2)

label.pack(side=TOP)

topframe = Frame(root)

topframe['bg']='dark orange'

topframe.pack()

bottomframe = Frame(root)

bottomframe['bg']='dark orange'

bottomframe.pack()

hindi = PhotoImage(file=r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\hindi image.png')

hinlabel2 = ttk.Label(topframe, text='')

hinbtn = ttk.Button(topframe, image=hindi,command=hindi\_page)

hinlabel1 = ttk.Label(topframe, text='')

hinlabel2.grid(row=0, column=0, pady=10)

hinbtn.grid(row=1, column=0, padx=30)

hinlabel1.grid(row=3, column=0, pady=10)

tamil = PhotoImage(file=r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\tamil image.png')

tamlabel2 = ttk.Label(topframe, text='')

tambtn = ttk.Button(topframe, image=tamil, command=tamil\_page)

tamlabel1 = ttk.Label(topframe, text='')

tamlabel2.grid(row=0, column=1, pady=10)

tambtn.grid(row=1, column=1, padx=30)

tamlabel1.grid(row=3, column=1, pady=10)

english = PhotoImage(file=r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\english image.png')

englabel2 = ttk.Label(topframe, text='')

engbtn = ttk.Button(topframe, image=english, command=english\_page)

englabel1 = ttk.Label(topframe, text='')

englabel2.grid(row=0, column=2, pady=10)

engbtn.grid(row=1, column=2, padx=30)

englabel1.grid(row=3, column=2, pady=10)

other = PhotoImage(file=r'C:\Users\Ricemms2\Documents\mp3 player(sachin)\other.png')

othlabel2 = ttk.Label(bottomframe, text='')

othbtn = ttk.Button(bottomframe, image=other, command=othermusic)

othlabel1 = ttk.Label(bottomframe, text='')

othlabel2.grid(row=0, column=3, pady=10)

othbtn.grid(row=1, column=3, padx=30)

othlabel1.grid(row=3, column=3, pady=10)

menu\_bar = Menu(root)

root.config(menu=menu\_bar)

sub\_menu = Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label='FILE', menu=sub\_menu)

sub\_menu.add\_command(label="ABOUT US", command=about\_us)

sub\_menu.add\_command(label="PROJECT DESCRIPTION", command=project)

sub\_menu.add\_command(label='EXIT', command=root.destroy)

root.protocol('WM\_DELETE\_WINDOW', on\_closing)

root.mainloop()

**CONCLUSION**

MP3 player is a device built to play and listen to digital audio files. These is MP3 files or. The player was built using Python language. A GUI implementation of the application was developed that is simple and easy to use.

The application provides the user with five options — to add song to a playlist, to play the song, to pause or resume the song, to stop the song.

The player also has the capability to add multiple songs to the playlist one by one. It has a large display area where the playlist is visible.

Once a song is selected and played, we can hear it and can also see details about the song. This information includes details about the song duration of the song, size of the file.

The Tkinter library of Python was used to create the GUI of the project. It was used to create the option buttons, the label and the display area.

The Pygame and Mutagen library was used to add songs, play the songs, provide pause and resume options.

In conclusion, a successful project was built in which songs will play.

This project “MP3 PLAYER” was done for using it in our school for helping to know more about programming language. In the process of completing this project I was able to learn many new concepts in python programming language and also I gained knowledge and experience on how a program works practically and what type of errors arises and how to clear the errors. Learning python and doing program first time was not a simple thing for me. But I sincerely thank MR.NAGARAJ.C who helped me through the project and clarified the doubt asked.

**THANK YOU.**

**BIBLIOGRAPHY**

**YOUTUBE CHANNEL**

Build\_with\_python

**BOOKS**

COMPUTER SCIENCE WITH PYTHON CLASS 11– SUMITA ARORA

COMPUTER SCIENCE WITH PYTHON CLASS 12– SUMITA ARORA

**WEBSITES REFERRED**

stackoverflow.com

[www.python.org](http://www.python.org)

realpython.com