GUI for Text To Speech

PROJECT REPORT

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By:

S.no.	Name	Roll No.	Registration no.
1.	Prince Kumar	04	11901772

Courses Code: INT213

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab (India)

Objective

The primary objective of this project is to implement what we've learnt throughout our course of Python programming and use that to develop a Graphical User Interface (GUI) for "Text To Speech" with all the required functionalities. This project also aims at providing a user friendly interface to the users to let them. The project deals with learning about the use some APIs and engines that offer it. There are a lot of APIs out there that offers this service, one of the commonly used services is Google Text to Speech. In this project, we will be using Google Text to Speech it.

Introduction

This software project is windows based application that reads a text file to the user. The bot reads a text file and associated pronunciations in its temporary database. The bot then reads an entire word to the user. The pronunciations of articles and basic words have been fed to the bot, the rest of the words and complex ones are calculated and read accordingly. The bot can be effectively used to help read the text document for the user so that the user does not constantly need to look at the screen and read the entire document. Test to speech converter is a recent software project that allows even the visually challenged to read and understand various documents.

Features:

Text input: Users are provided with a text box where they can enter the required text in the software.

gTTS : gTTS is a very easy to use tool which converts the text entered, into audio which can be saved as a mp3 file.

Play/Reset Options: Users can play, reset too or play from start the speech whenever required.

Software Requirements:

- Windows Xp, Windows 7(ultimate, enterprise)
- mysql
- gTTS

Hardware Components:

- Processor – i3
- Hard Disk – 5 GB
- Memory – 1GB RAM

Advantages:

- The system is helpful for persons having learning disabilities or visually challenged.
- Prevents eye from strain, and user can sit and listen comfortably.
- Saves time especially while driving, exercising.
- Easy to use.
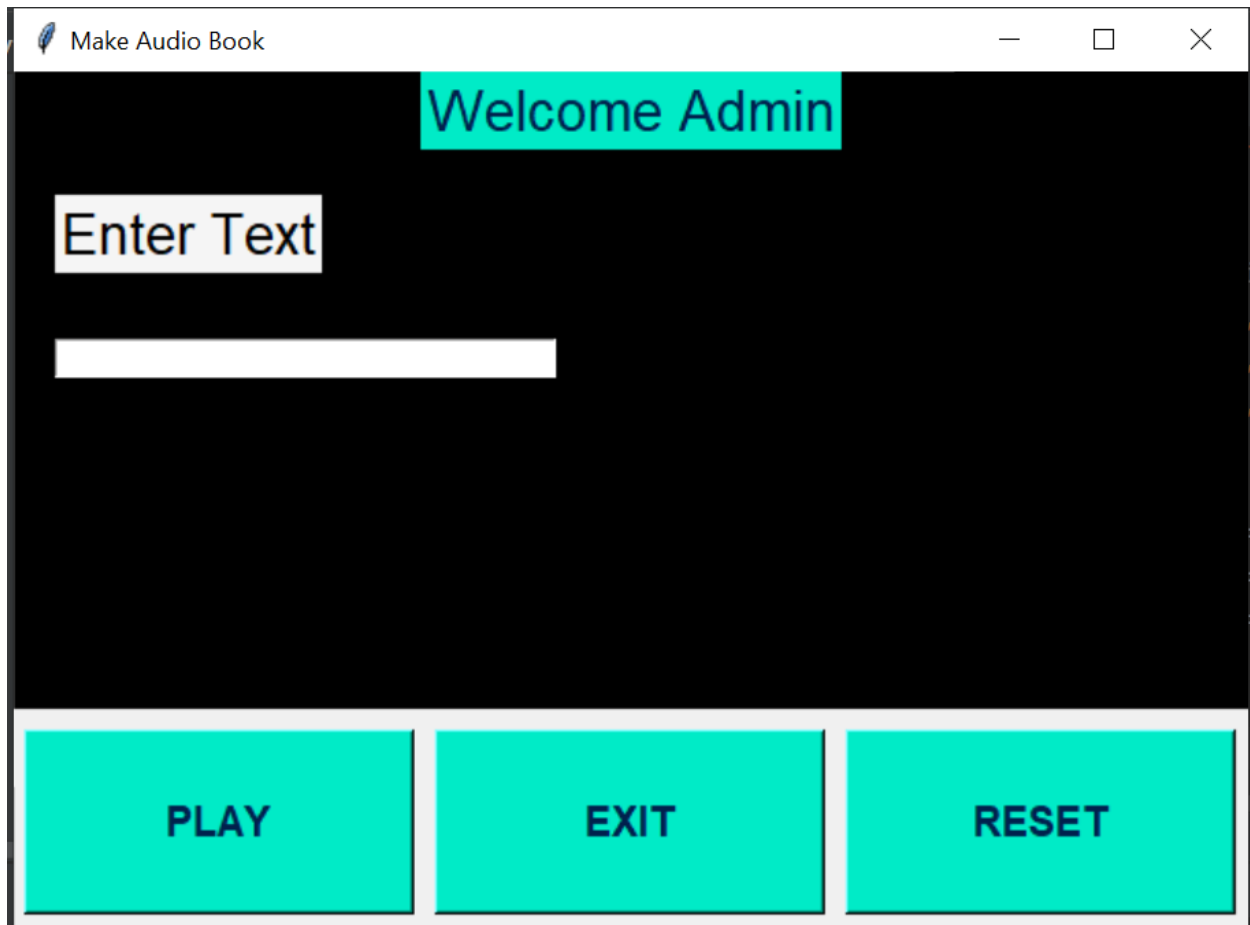- Help improving spelling, reading, writing skills.

Disadvantages:

- Not natural sounding.
- Cannot read symbols.

Applications:

- Can be used for visually and learning disabled people.
- Can be used in learning pronunciations.
- Can be used as reminders in offices, houses.

GUI Screenshots:

Login

Please enter Login details

Username

Password

Login   Back

Welcome to GTTS

Login

Register

SOURCE CODE

tts.py

```python
from tkinter import *

from gtts import gTTS

import tkinter

import platform


root = Tk()

root.geometry("450x600")

root.configure(bg='black')

root.title("Make Audio Book")

# Frame ---------------------------------------------------

header = tkinter.Frame(root,bg='black')

header.pack(side=TOP, fill=X)

Label(header, text = "Welcome Admin", font=("TitilliumWeb-Regular", 20), bg ='#00ebc7',
fg='#00214d').pack()




Msg = StringVar()
```

```python
Label(root,text ="Enter Text", font=("TitilliumWeb-Regular", 20), bg ='white
smoke').place(x=20,y=60)

entry_field = Entry(root, textvariable = Msg ,width ='40')

entry_field.place(x=20,y=130)

def Text_to_speech():

    Message = entry_field.get()

    speech = gTTS(text = Message)

    speech.save('tts-save.mp3')



def Exit():

    root.destroy()



def Reset():

    Msg.set("")



# Group1 Frame ----------------------------------------------------

body = tkinter.Frame(root)

body.pack(side=BOTTOM, fill=X)



p=Button(body, text = "PLAY", font = 'arial 15 bold' , command = Text_to_speech
,width=15,height=3,bg ='#00ebc7', fg='#00214d')

x=Button(body, font = 'arial 15 bold',text = 'EXIT', width=15,height=3, command = Exit,  bg
='#00ebc7', fg='#00214d')

r=Button(body, font = 'arial 15 bold',text = 'RESET',width=15,height=3, command = Reset, bg
='#00ebc7', fg='#00214d')
```

```python
if platform.system() == "Darwin":  # if its a Mac

    p.configure(highlightbackground="#00ebc7", fg="#00214d")

    x.configure(highlightbackground="#00ebc7", fg="#00214d")

    r.configure(highlightbackground="#00ebc7", fg="#00214d")


else:  # if its Windows or Linux

    p.configure(bg="#00ebc7", fg="#00214d")

    x.configure(bg="#00ebc7", fg="#00214d")

    r.configure(bg="#00ebc7", fg="#00214d")


p.grid(row=0, column=1, padx=5, pady=10)

x.grid(row=0, column=2, padx=5, pady=10)

r.grid(row=0, column=3, padx=5, pady=10)


root.mainloop()
```

Signup Page

```python
from tkinter import*
import mysql.connector
import tkinter
```

```python
mydb=mysql.connector.connect(host="localhost",user="root",passwd="1234@567",database="signup"
)
mycursor=mydb.cursor()
# mycursor.execute("create database signup")

# mycursor.execute("create table detail(Name varchar(50),Email varchar(50),Gender varchar(50),Age
int(3),Passwd varchar(30))")



mycursor.execute("select * from detail")
for tb in mycursor:
    print(tb)


signup = Tk()
signup.geometry('500x500')
signup.title("Registration Form")
signup.configure(bg='black')

label_0 = Label(signup, text="Registration
form",width=20,bg="#11afed",fg="black",font=("TitilliumWeb-Regular", 20))
label_0.place(x=150,y=0)

label_1 = Label(signup, text="FullName",width=7,bg="#11afed",fg="black",font=("TitilliumWeb-
Regular", 20))
label_1.place(x=0,y=40,)

entry_1 = Entry(signup,textvariable=label_1, bg="orange", fg="black", font="lucida 15")
entry_1.place(x=120,y=40)

label_2 = Label(signup, text="Email",width=7,bg="#11afed",fg="black",font=("TitilliumWeb-
Regular", 20))
label_2.place(x=0,y=80)

entry_2 = Entry(signup, textvariable=label_2, bg="orange", fg="black", font="lucida 15")
entry_2.place(x=120,y=80)

label_3 = Label(signup, text="Gender",width=7,bg="#11afed",fg="black",font=("TitilliumWeb-
Regular", 20))
label_3.place(x=0,y=120)
```

```python
var = IntVar()
Radiobutton(signup, text="Male",padx = 5, variable=var, value=1).place(x=120,y=120)
Radiobutton(signup, text="Female",padx = 20, variable=var, value=2).place(x=200,y=120)

label_4 = Label(signup, text="Age:",width=7,bg="#11afed",fg="black",font=("TitilliumWeb-Regular",
20))
label_4.place(x=0,y=160)

entry_3 = Entry(signup, textvariable=label_3, bg="orange", fg="black", font="lucida 15")
entry_3.place(x=120,y=160)

label_5 = Label(signup, text="Password:",width=7,bg="#11afed",fg="black",font=("TitilliumWeb-
Regular", 20))
label_5.place(x=0,y=200)

entry_5 = Entry(signup, textvariable=label_5, bg="orange", fg="black", font="lucida 15")
entry_5.place(x=120,y=200)

entry_1  = StringVar()
entry_2 = StringVar()
entry_3 = IntVar()
entry_4 = IntVar()
entry_5 = StringVar()

def signUP(signup):
    # message = tkinter.Frame(signup, bg='yellow')
    # message.place(x=100,y=280)
    Label(signup, text='welcome`{ Name }`User',font=("TitilliumWeb-Regular",
20),fg='black').place(x=100,y=280)

def data(label_1,label_2,label_3,label_4,label_5):
    cur = mydb.cursor()
    Name = entry_1 .get( )
    Email = entry_2.get( )
    Gender = entry_3.get( )
    Age = entry_4.get( )
    Passwd=entry_5.get()
    sql = 'insert into detail(Name,Email,Gender,Age,Passwd) values(%s,%s,%s,%s,%s)'
    values = (Name, Email, Gender, Age,Passwd)
    cur.execute(sql, values)

    cur.execute('commit')
```

```
        cur.close( )
        mydb.close( )

sign=Button(signup,width=7, height=1,text="SignUP",bg="black",fg="orange",font="lucida 15
bold",command=lambda:[signUP(signup),data(entry_1 ,entry_2,entry_3,entry_4,entry_5)])
sign.place(x=0,y=240)



def goback():
        signup.destroy( )
        import main


Button(signup, command=goback, text="Back", width=7, height=1,font=("TitilliumWeb-Regular",
20)).place(x=120,y=240)
signup.mainloop()
```

<u>Login Page</u>

```
from tkinter import *
import tkinter
import mysql.connector

# mydb = mysql.connector.connect(host="localhost", user="root", passwd="1234@567",
database="signup")




# mycursor.execute("show tables")
# for tb in mycursor:
#     print(tb)

def LoginPage():
    login_screen=Tk()
    login_screen.title("Login")
    login_screen.geometry("500x500")
    login_screen.configure(bg='black')

    Label(login_screen, text="Please enter Login details",bg="#11afed",fg="black").place(x=150,y=0)
```

```python
Label(login_screen, text="Username").place(x=10,y=40,)

username_entry = Entry(login_screen, textvariable="username")
username_entry.place(x=120,y=40,)

Label(login_screen, text="Password").place(x=10,y=80,)

password_entry = Entry(login_screen, textvariable="password", show= '*')
password_entry.place(x=120,y=80,)

def login():
    if username_entry.get()=='admin' and password_entry.get() =='123':
        print("Login Sucessful")
        login_screen.destroy( )
        import tts
    else:
        message = tkinter.Frame(login_screen, bg='white')
        message.pack(side=TOP, fill=X)
        Label(message,text="Unknown User").pack()

Label(login_screen, text="").pack()
Button(login_screen, text="Login",command=login,width=10, height=1).place(x=10,y=120,)

def back():
    login_screen.destroy( )
    import main
Button(login_screen, command=back,text="Back", width=10, height=1).place(x=120,y=120,)
login_screen.mainloop()

    LoginPage()
```

Main.py

```python
from tkinter import *

import platform

root = Tk()

root.geometry("450x600")

root.configure(bg='black')
```

```python
root.title("Make Audio Book")

Label(root, text = "Welcome to GTTS", font=("TitilliumWeb-Regular", 20), bg ='#00ebc7',
fg='#00214d').pack()

def login():

    root.destroy()

    import login

def signup():

    root.destroy( )

    import signup


log=Button(root, text = "Login", command = login,width=15,height=3,bg ='#00ebc7',
fg='#00214d',)

reg=Button(root, text = "Register", command = signup ,width=15,height=3,bg ='#00ebc7',
fg='#00214d')


if platform.system() == "Darwin":

    reg.configure(highlightbackground="#00ebc7", fg="#00214d")

    log.configure(highlightbackground="#00ebc7", fg="#00214d")

else:

    reg.configure(bg="#00ebc7", fg="#00214d")

    log.configure(bg="#00ebc7", fg="#00214d")


log.pack(side=LEFT)

reg.pack(side=RIGHT)
```

root.mainloop()

Conclusion

- The end product is obtained that includes all the mentioned modules discussed earlier.

- Learnt to make a GUI using Tkinter in Python.

- Learnt to use Google Text To Speech.

- Learnt to implement database connectivity using mysql.

- The project is capable of convert text into speech.

References

- https://www.geeksforgeeks.org/python/

- https://basics.org/text-to-speech/python/

- https://www.thepythoncode.com/article/convert-text-to-speech-in-python