**OSTL PROJECT** C-13

NAME: HARSH HARWANI-51

JENIL JAIN-55

VEDANT JASANI-59

TITLE: VOICE ASSISTANT/CHAT BOT

DESCRIPTION:

PACKAGES USED:

Pyttsx3 lets you synthesize text in to audio you can hear. This package works in Windows, Mac, and Linux. It uses native speech drivers when available and works completely offline.

There are some other cool features that are not covered here, like the event system. You can hook in to the engine on certain events. You can use this to count how many words are said and cut it off if it has received input that is too long. You can inspect each word and cut it off if there are inappropriate words. The event hooks are not covered here but are worth a mention.

The pyttsx3 module supports native Windows and Mac speech APIs but also supports espeak, making it the best available text-to-speech package in my opinion.

To perform the code you need to install the following packages in python:-

pyttsx3:

An OFFLINE Python Text to Speech library (TTS) which works for both python3 and python2.This library very usefull especially if you don't want any delay in the speech produced and don't want to depend only on the internet for TTS conversion. It also supports multiple TTS engines like Sapi5 , nsss , espeak .

Command to install:-

pip install pyttsx3

SpeechRecognition:

Library for performing speech recognition, with support for several engines and APIs, online and offline.

Command to install:-

pip install SpeechRecognition

PyAudio:

PyAudio is required if and only if you want to use microphone input (Microphone). PyAudio version 0.2.11+ is required, as earlier versions have known memory management bugs when recording from microphones in certain situations.

If not installed, everything in the library will still work, except attempting to instantiate a Microphone object will raise an AttributeError.

Command to install:-

pip install PyAudio

wikipedia:

Command to install:-

pip install wikipedia

**CODE:**

import pyttsx3

import speech\_recognition as sr

import wikipedia

import webbrowser

recording=sr.Recognizer()

mic=sr.Microphone()

def Alina():

#pyttsx3 initializations

engine=pyttsx3.init()

voices=engine.getProperty('voices')

engine.setProperty('voice',voices[7].id)

engine.setProperty('rate',150)

x=""

while x=="":

with mic as source:

recording.adjust\_for\_ambient\_noise(source)

audio=recording.listen(source)

x=recording.recognize\_google(audio)

if x.lower()=="let's go alina":

engine.say("Alina here, how can I help you ?")

print("alina here, how can I help you ?")

engine.runAndWait()

engine.stop()

return Requests()

def Requests():

#pyttsx3 initializations

engine=pyttsx3.init()

voices=engine.getProperty('voices')

engine.setProperty('voice',voices[7].id)

engine.setProperty('rate',150)

with mic as source:

audio=recording.listen(source)

x=recording.recognize\_google(audio)

print(x)

engine.say("I can search for "+x)

print("I can search for "+x)

engine.say("Say yes if you want me to search for "+x)

print("Say yes if you want me to search for "+x)

engine.runAndWait()

engine.stop()

return Search(x)

def Search(y):

engine=pyttsx3.init()

voices=engine.getProperty('voices')

engine.setProperty('voice',voices[7].id)

engine.setProperty('rate',150)

with mic as source:

audio=recording.listen(source)

order=recording.recognize\_google(audio)

if(order=="yes"):

engine.say("Which engine would you like me to use?")

engine.say("Is it google, yahoo or bing ?")

print("Is it GOOGLE, YAHOO or BING ?")

engine.runAndWait()

with mic as source:

audio=recording.listen(source)

searchengine=recording.recognize\_google(audio)

if("google" in searchengine.lower()):

engine.say("Here you go")

tabURL="http://google.com/?#q=";

webbrowser.open(tabURL+y);

engine.say("These are some results I got from google")

engine.runAndWait()

engine.stop()

elif("yahoo" in searchengine.lower()):

engine.say("Here you go")

tabURL="http://yahoo.com/?#q=";

webbrowser.open(tabURL+y);

engine.say("These are some results I got from yahoo")

engine.runAndWait()

engine.stop()

elif("bing" in searchengine.lower()):

engine.say("Here you go")

tabURL="http://bing.com/?#q=";

webbrowser.open(tabURL+y);

engine.say("These are some results I got from bing")

engine.runAndWait()

engine.stop()

else:

engine.say("I can't recognize the audio")

print("I can't recognize the audio")

OUTPUT: