AI PROJECT

VOICE ASSISTANT

SUBMITTED BY:

Pragati Rai- 102017122

Radhika Aggarwal-102017139

Aim : Developing a voice assistant named ‘Jarvis’ by importing different libraries in Jupyter notebook.

Code:

import pyttsx3 #pip install pyttsx3

import speech\_recognition as sr #pip install speechRecognition

import datetime

import wikipedia #pip install wikipedia

import webbrowser

import os

import smtplib

engine = pyttsx3.init('sapi5')

voices = engine.getProperty('voices')

# print(voices[1].id)

engine.setProperty('voice', voices[0].id)

def speak(audio):

engine.say(audio)

engine.runAndWait()

def wishMe():

hour = int(datetime.datetime.now().hour)

if hour>=0 and hour<12:

speak("Good Morning!")

elif hour>=12 and hour<18:

speak("Good Afternoon!")

else:

speak("Good Evening!")

speak("I am Jarvis Sir. Please tell me how may I help you")

def takeCommand():

#It takes microphone input from the user and returns string output

r = sr.Recognizer()

with sr.Microphone() as source:

print("Listening...")

r.pause\_threshold = 1

audio = r.listen(source)

try:

print("Recognizing...")

query = r.recognize\_google(audio, language='en-in')

print(f"User said: {query}\n")

except Exception as e:

# print(e)

print("Say that again please...")

return "None"

return query

def sendEmail(to, content):

server = smtplib.SMTP('smtp.gmail.com', 587)

server.ehlo()

server.starttls()

server.login('youremail@gmail.com', 'your-password')

server.sendmail('youremail@gmail.com', to, content)

server.close()

if \_\_name\_\_ == "\_\_main\_\_":

wishMe()

while True:

# if 1:

query = takeCommand().lower()

# Logic for executing tasks based on query

if 'wikipedia' in query:

speak('Searching Wikipedia...')

query = query.replace("wikipedia", "")

results = wikipedia.summary(query, sentences=2)

speak("According to Wikipedia")

print(results)

speak(results)

elif 'open youtube' in query:

webbrowser.open("youtube.com")

elif 'open google' in query:

webbrowser.open("google.com")

elif 'open stackoverflow' in query:

webbrowser.open("stackoverflow.com")

elif 'play music' in query:

music\_dir = 'D:\\Non Critical\\songs\\Favorite Songs2'

songs = os.listdir(music\_dir)

print(songs)

os.startfile(os.path.join(music\_dir, songs[0]))

elif 'the time' in query:

strTime = datetime.datetime.now().strftime("%H:%M:%S")

speak(f"Sir, the time is {strTime}")

elif 'open code' in query:

codePath = "C:\\Users\\Haris\\AppData\\Local\\Programs\\Microsoft VS Code\\Code.exe"

os.startfile(codePath)

elif 'email to harry' in query:

try:

speak("What should I say?")

content = takeCommand()

to = "harryyourEmail@gmail.com"

sendEmail(to, content)

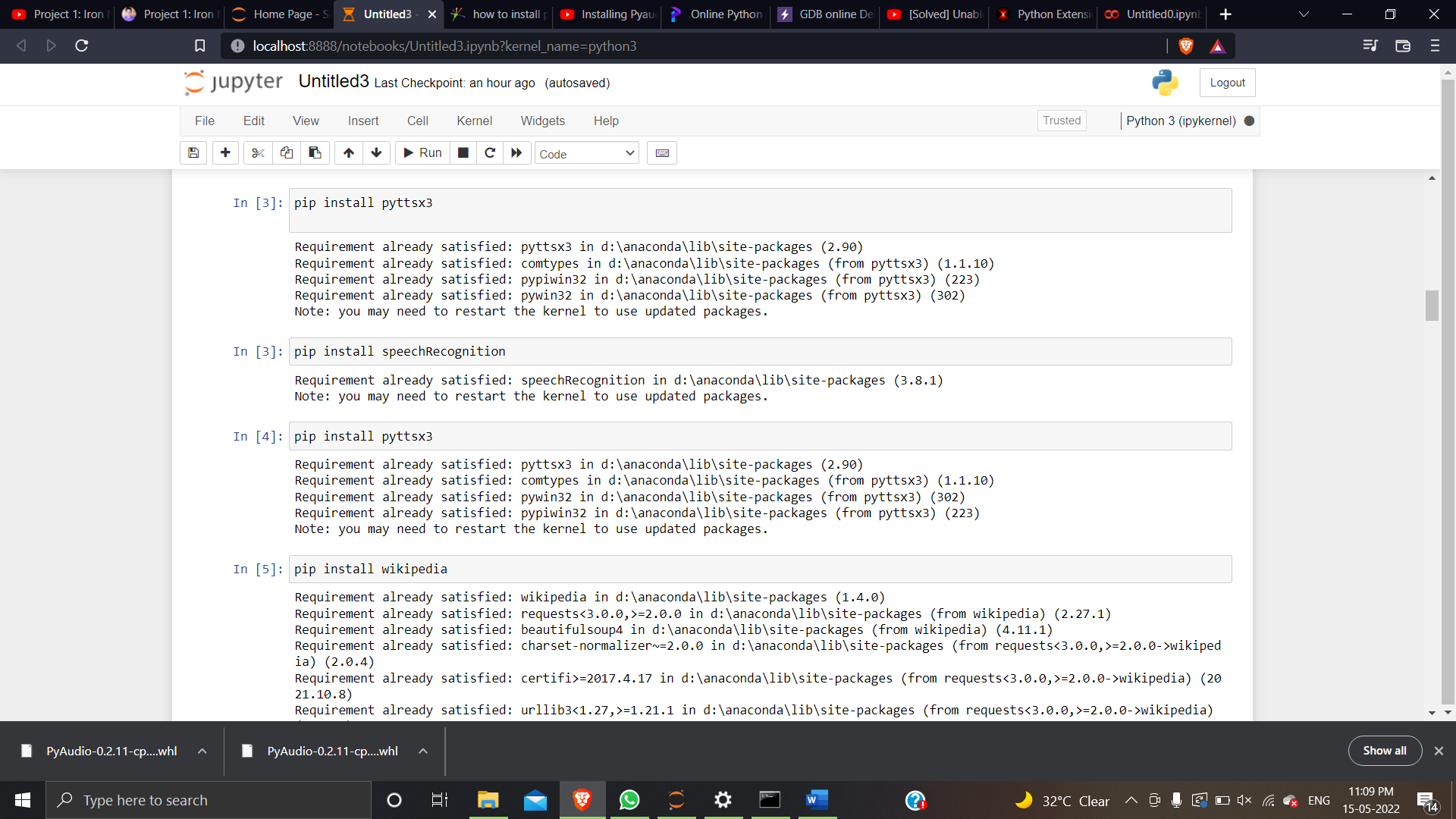
speak("Email has been sent!")

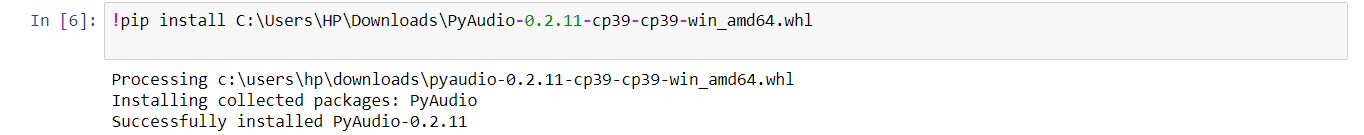
except Exception as e:

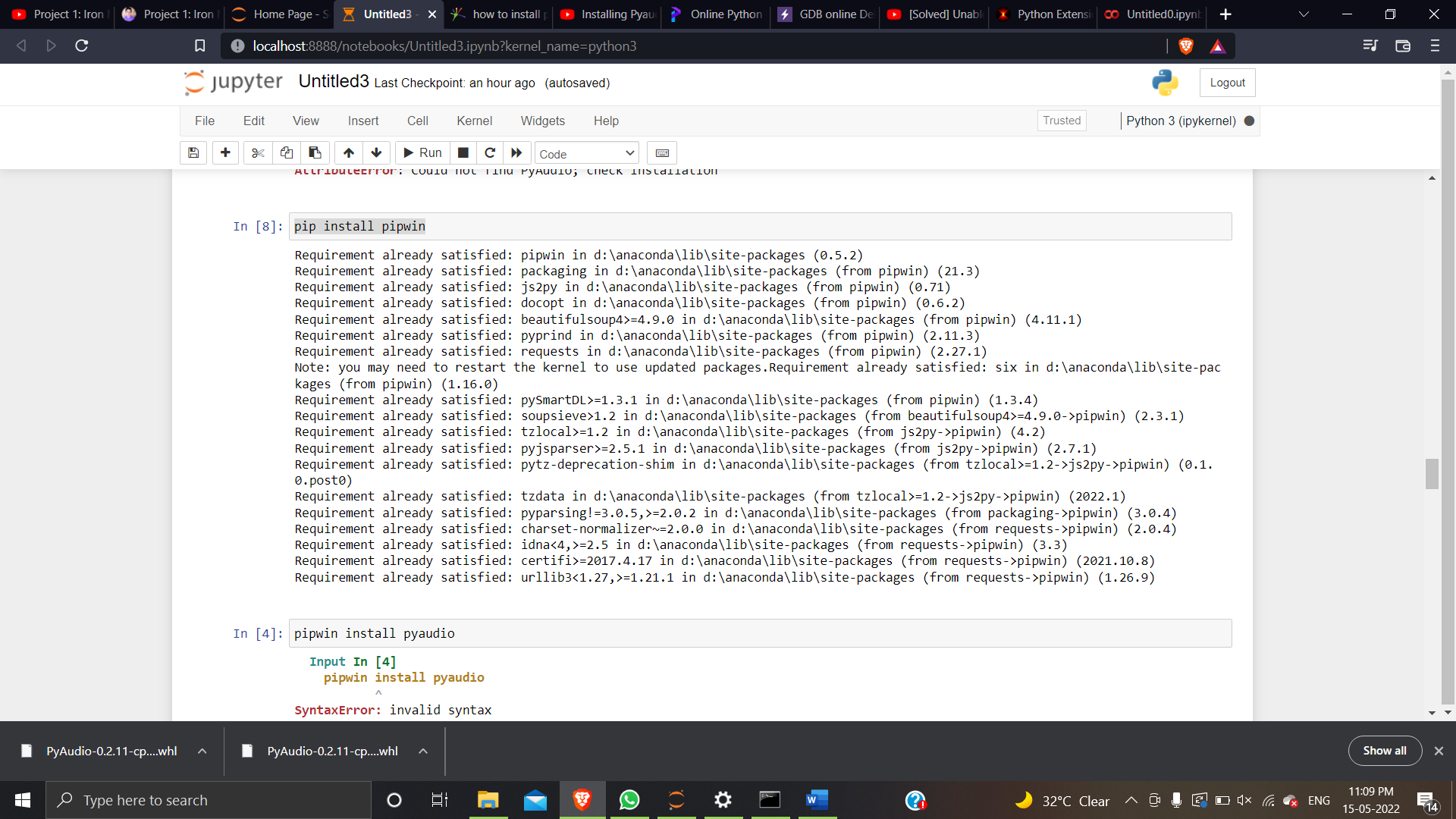
print(e)

speak("Sorry my friend harry bhai. I am not able to send this email")

Following libraries are installed:







Coding logic for Voice assistant:

we will develop logic for different commands such as Wikipedia searches, playing music, etc. To do Wikipedia searches, we need to install and import the Wikipedia module into our program. Type the below command to install the Wikipedia module :

pip install wikipedia

To open any website, we need to import a module calledwebbrowser. It is an in-built module, and we do not need to install it with a pip statement; we can directly import it into our program by writing an import statement.

Code:

elif 'open youtube' in query:

webbrowser.open("youtube.com")

we are using an elif loop to check whether Youtube is in the user's query. Let' suppose the user gives a command as "J.A.R.V.I.S., open youtube." So, open youtube will be in the user's query, and the elif condition will be true.

elif 'open google' in query:

webbrowser.open("google.com")

To play music, we need to import a module called os. Import this module directly with an import statement.

elif 'play music' in query:

music\_dir = 'D:\\Non Critical\\songs\\Favorite Songs2'

songs = os.listdir(music\_dir)

print(songs)

os.startfile(os.path.join(music\_dir, songs[0]))

In this code we are using the datetime() function and storing the current or live system time into a variable called strTime. After storing the time in strTime, we are passing this variable as an argument in speak function. Now, the time string will be converted into speech.

elif 'the time' in query:

strTime = datetime.datetime.now().strftime("%H:%M:%S")

speak(f"Sir, the time is {strTime}")

To open VS code program

elif 'open code' in query:

codePath = "C:\\Users\\Haris\\AppData\\Local\\Programs\\Microsoft VS Code\\Code.exe"

os.startfile(codePath)

**To send email**- we will create a sendEmail() function, which will help us send emails to one or more than one recipient.

def sendEmail(to, content):

server = smtplib.SMTP('smtp.gmail.com', 587)

server.ehlo()

server.starttls()

server.login('youremail@gmail.com', 'your-password')

server.sendmail('youremail@gmail.com', to, content)

server.close()