A
Project Report
on

SMART VOICE ASSISTANT (JARVIS)

Submitted by
CHAMAN SINGH
(Roll No.: 20MA60R11)

in partial fulfillment for the award of the degree
of
M. TECH.
in
COMPUTER SCIENCE AND DATA PROCESSING

Department of Mathematics
IIT Kharagpur
Kharagpur – 721302, India

CERTIFICATE

Certified that this project report **"SMART VOICE ASSISTANT" (JARVIS)** is the bona-fide work of **"CHAMAN SINGH"** who carried out the project work under my supervision.

Guide

Dr. Pawan Kumar

Department of Mathematics
IIT Kharagpur
Kharagpur – 721302, India

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my project guide **Dr. Pawan Kumar** who gave me the golden opportunity to do this wonderful project on the topic **SMART VOICE ASSISTANT (JARVIS)**, which also helped me in doing a lot of research and I came to know about so many new things I am really thankful to them.

CHAMAN SINGH

(Roll No : 20MA60R11)

M. Tech.

Computer Science and Data Processing

Date : 11/04/20022

Place : IIT Kharagpur

Index

ABSTRACT

As we recognize Python is an rising language so it turns into clean to put in writing a script for Voice Assistant in Python. The commands for the assistant may be treated as according to the requirement of user. Speech reputation is the method of changing speech into text. This is typically utilized in voice assistants like Alexa, Siri, Cortana etc. In Python there may be an API referred to as SpeechRecognition which permits us to transform speech into text. It become an exciting undertaking to make my personal assistant. It have become less difficult to ship emails with out typing any word, Searching on Google with out commencing the browser, and acting many different day by day obligations like playing music, commencing your favorite IDE with the assist of a unmarried voice command. In the cutting-edge scenario, development in technology are such that they are able to carry out any undertaking with equal effectiveness or can say greater correctly than us. By making this task, I found out that the idea of AI in each area is reducing human attempt and saving time. Functionalities of this task include:

1. It can open calculator, MS word, excel etc.
2. It can play music online.
3. It can open WhatsApp in web browser.
4. It can open terminal, your favorite IDE like Visual Studio Code, Notepad etc.
5. It can play system music.
6. It can do Wikipedia searches for you and it can print and speak two paragraph according to Wikipedia
7. It can open websites like Google, YouTube, iit kgp etc., in a web browser.
8. It can have some basic conversation like hello, how are you etc.
9. It can send Emails.
10. It can set alarm.

Now the simple query arises in thoughts that how it's far an AI? The digital assistant that I actually have created is like if it isn't an AI, however it's far the output of a package of the statement. But fundamentally, the main motive of AI machines is that it could carry out human responsibilities with the equal performance or maybe extra efficaciously than humans. It is a truth that my digital assistant isn't a excellent instance of AI, however it's far an AI.

Chapter 1: Introduction

1. INTRODUCTION

Artificial Intelligence while used with machines, it suggests us the functionality of questioning like humans. In this, a pc device is designed in any such manner that generally calls for interplay from human. As we understand Python is an rising language so it will become clean to jot down a script for Voice Assistant in Python. The commands for the assistant may be dealt with as according to the requirement of user. Speech Recognition is the Alexa, Siri, Cortana etc. In Python there's an API known as SpeechRecognition which lets in us to transform speech into text. It changed into an thrilling assignment to make my very own assistant. It have become simpler to open MS word, Searching on Google without starting the browser, and acting many different each day responsibilities like playing music, starting your favored IDE with the assist of a unmarried voice command. In the modern scenario, development in technology are such that they could carry out any assignment with identical effectiveness or can say greater successfully than us. By making this project, I found out that the idea of AI in each area is reducing human attempt and saving time.

As the voice assistant is the use of Artificial Intelligence subsequently the end result that it's far imparting are fairly correct and green. The assistant can assist to lessen human attempt and consumes time whilst appearing any task, they eliminated the idea of typing absolutely and behave as some other man or woman to whom we're speaking and asking to carry out task. The assistant isn't anything any much less than a human assistant however we will say that that is extra powerful and green to carry out any task. The libraries and applications used to make this assistant specializes in the time complexities and decreases time.

The functionalities include, It can open calculator, word, excel etc. It can play music online. It can open WhatsApp on web. It can open terminal, your favorite IDE like Visual Studio Code, Sublime Text, Notepad etc. It can play system music. It can do Wikipedia searches for you and it can print and speak two paragraph according to Wikipedia. It can open websites like Google, YouTube, iit kgp etc., in a web browser. It can have some basic conversation like hello, how are you etc.

Tools and technology used are Visual Studio Code IDE for making this project, and I created all py documents in Visual Studio Code. Along with this I used following modules and libraries in my project. pyttsx3, SpeechRecognition, datetime, Wikipedia, pywhatkit, pyjokes, pyQt5, smtplib, etc. I actually have created a stay GUI for interacting with the JARVIS because it offers a layout and exciting appearance whilst having the conversation.

1.1 PRESENT SYSTEM

We are familiar with many existing voice assistants like Alexa, Siri, Google Assistant, Cortana which uses concept of language processing, and voice recognition. They listens the command given by the user as per their requirements and performs that specific function in a very efficient and effective manner.

As those voice assistants are the usage of Artificial Intelligence for this reason the end result that they may be offering are tremendously correct and green. These assistants can assist to lessen human attempt and consumes time whilst appearing any task, they eliminated the idea of typing absolutely and behave as any other person to whom we're speaking and asking to carry out task. These assistants aren't anything any much less than a human assistant however we will say that they may be greater powerful and green to carry out any task. The set of rules used to make those assistant makes a specialty of the time complexities and decreases time.

But for the usage of those assistants one need to have an account (like Google account for Google assistant, Microsoft account for Cortana) and may use it with net connection handiest due to the fact those assistants are going to paintings with net connectivity. They are incorporated with many gadgets like, phones, laptops, and audio system etc.

1.2 PROPOSED SYSTEM

It was an interesting task to make my own assistant. It became easier to open MS word, Calculator, MS Excel, Searching on Google without opening the browser, and performing many other daily tasks like playing music, opening your favorite IDE like Visual studio Code Sublime Text etc. with the help of a single voice command. **JARVIS** is different from other traditional voice assistants in terms that it is specific to desktop and user does not need to make account to use this, it also require internet connection while getting the instructions to perform any task.

The IDE used in this project is Visual studio Code. All the python files were created in Visual studio Code and all the necessary packages were easily installable in this IDE. For this project following modules and libraries were used i.e. pyttsx3, SpeechRecognition, datetime, Wikipedia, pywhatkit, pyjokes, PyQt5, os, smtplib, etc. I have created a live GUI for interacting with the **JARVIS** as it gives a design and interesting look while having the conversation.

With the advancement JARVIS can perform any task with same effectiveness or can say more effectively than us. By making this project, I realized that the concept of AI in every field is decreasing human effort and saving time. Functionalities of this project include, It can open calculator, MS word, excel etc. It can play music online. It can open WhatsApp on web. It can open terminal, your favorite IDE like Visual Studio Code, Sublime Text, Notepad etc. It can also play system music. It can do Wikipedia searches for you and it can print and speak two paragraph according to Wikipedia. It can open websites like Google, YouTube, iit kgp etc., in a web browser. It can have some basic conversation like hello, how are you etc.

Chapter 2: System Design

2.1 DATA FLOW

The data flow for JARVIS is as follow:

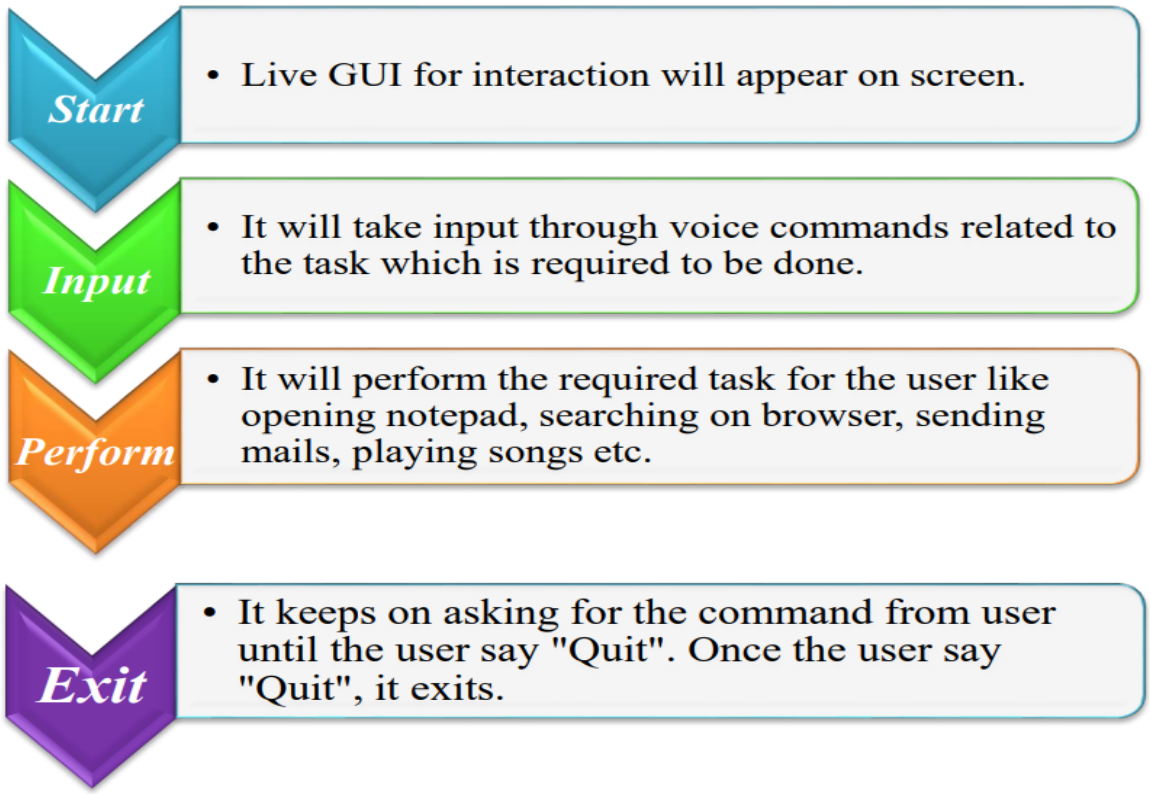

Figure 2.1 Data flow for JARVIS

The gadget is designed the use of the idea of Artificial Intelligence and with the assist of essential applications of Python. Python affords many libraries and applications to carry out the obligations, as an example pywhatkit may be used to look end result on web. The info of those applications are noted in Chapter three of this report.

The statistics on this venture is not anything however person enter, regardless of the person says, the assistant plays the undertaking accordingly. The person enter is not anything particular however the listing of obligations which a person desires to get done in human language i.e. English.

Chapter 3: Software Details

The IDE used in this project is Visual Studio Code . All the python files were created in Visual Studio Code and all the necessary packages were easily installable in this IDE. For this project following modules and libraries were used i.e. pyttsx3, SpeechRecognition, datetime, Wikipedia, pywhatkit, pyjokes, PyQt5 etc. I have created a live GUI for interacting with the JARVIS as it gives a design and interesting look while having the conversation.

3.1. Visual Studio Code

It is an IDE i.e. Integrated Development Environment which has many features like it supports scientific tools (like matplotlib, numpy, scipy etc.) web frameworks (example Django, web2py and Flask) refactoring in Python, integrated python debugger, code completion, code and project navigation etc.

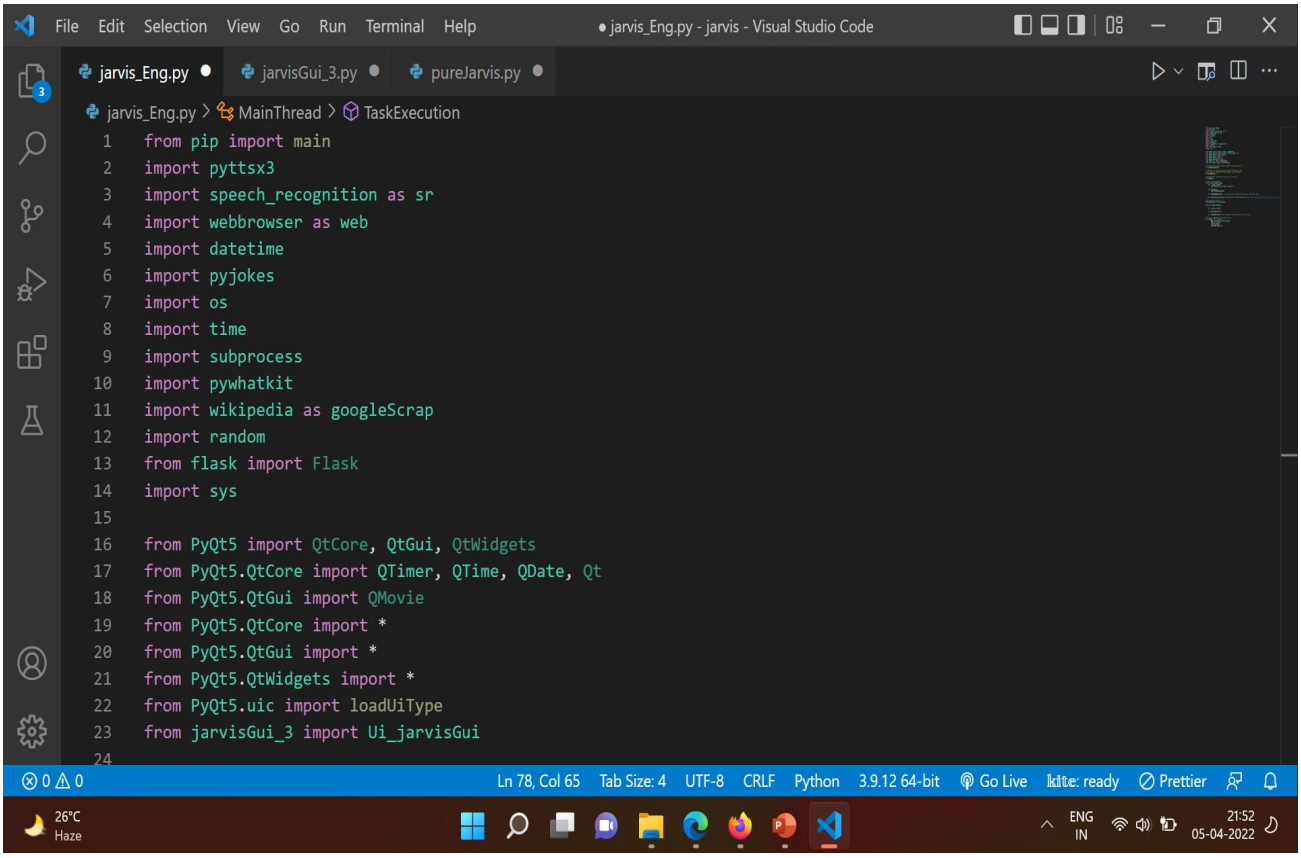

Figure 3.1 Visual Studio Code IDE

3.2. QT DESIGNER FOR LIVE GUI

PyQt5 is the most important python binding. It contains set of GUI widgets. PyQt5 has some important python modules like QtWidgets, QtCore, QtGui, and QtDesigner etc.

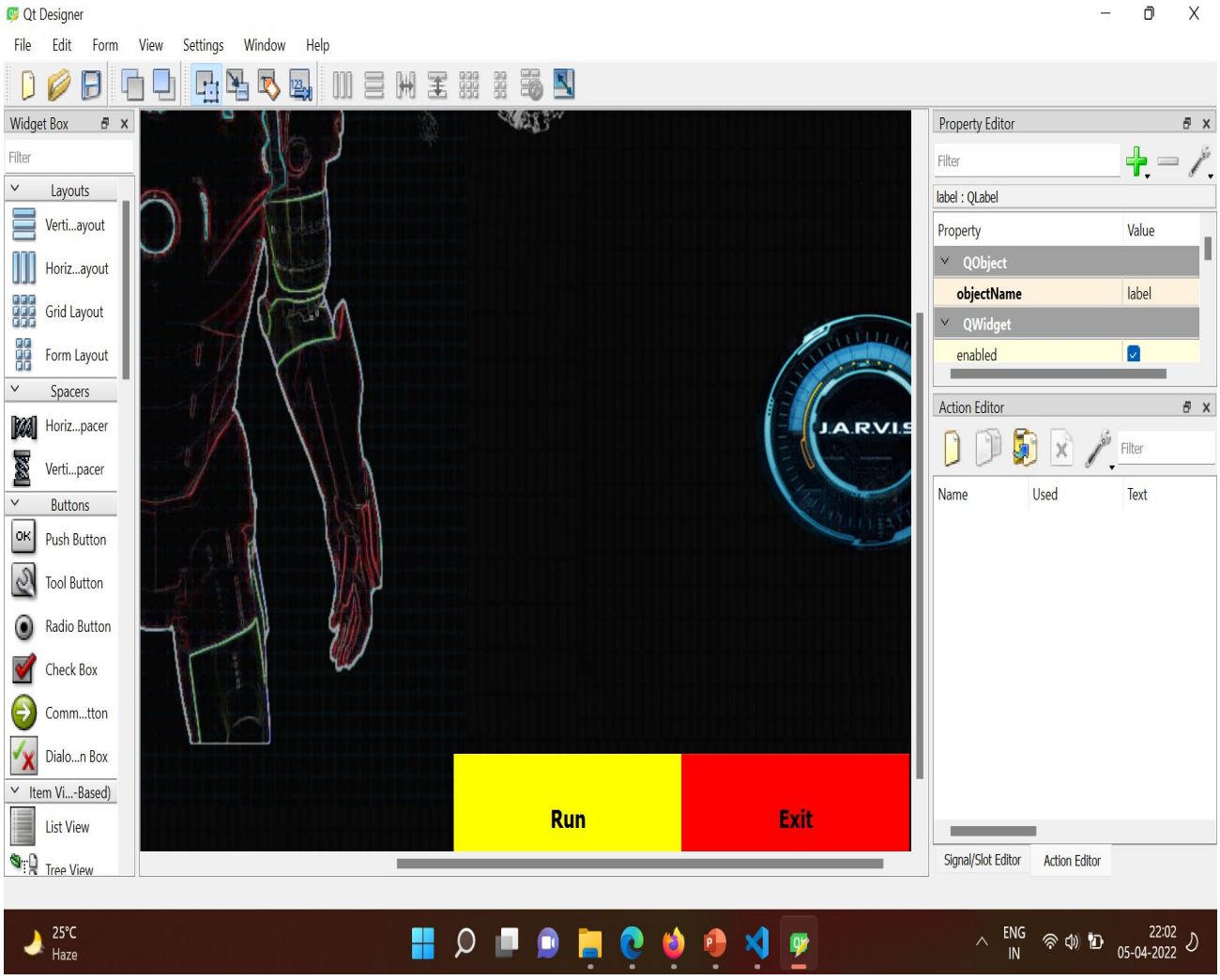

Figure 3.2 Qt Designer

3.3. PYTHON LIBRARIES

In JARVIS following python libraries were used:

3.3.1. pyttsx3: It is a python library which converts text to speech.

3.3.2. SpeechRecognition: It is a python module which converts speech to text.

3.3.3. pywhatkit: It is python library to send WhatsApp message at a particular time with some additional features like searching on web.

3.3.4. datetime: This library provides us the actual date and time.

3.3.5. Wikipedia: It is a python module for searching anything on Wikipedia.

3.3.6. Pyjokes: It is a python libararies which contains lots of interesting jokes in it.

3.3.7. Webbrowser: It provides interface for displaying web-based documents to users.

3.3.8. os: It represents Operating System related functionality.

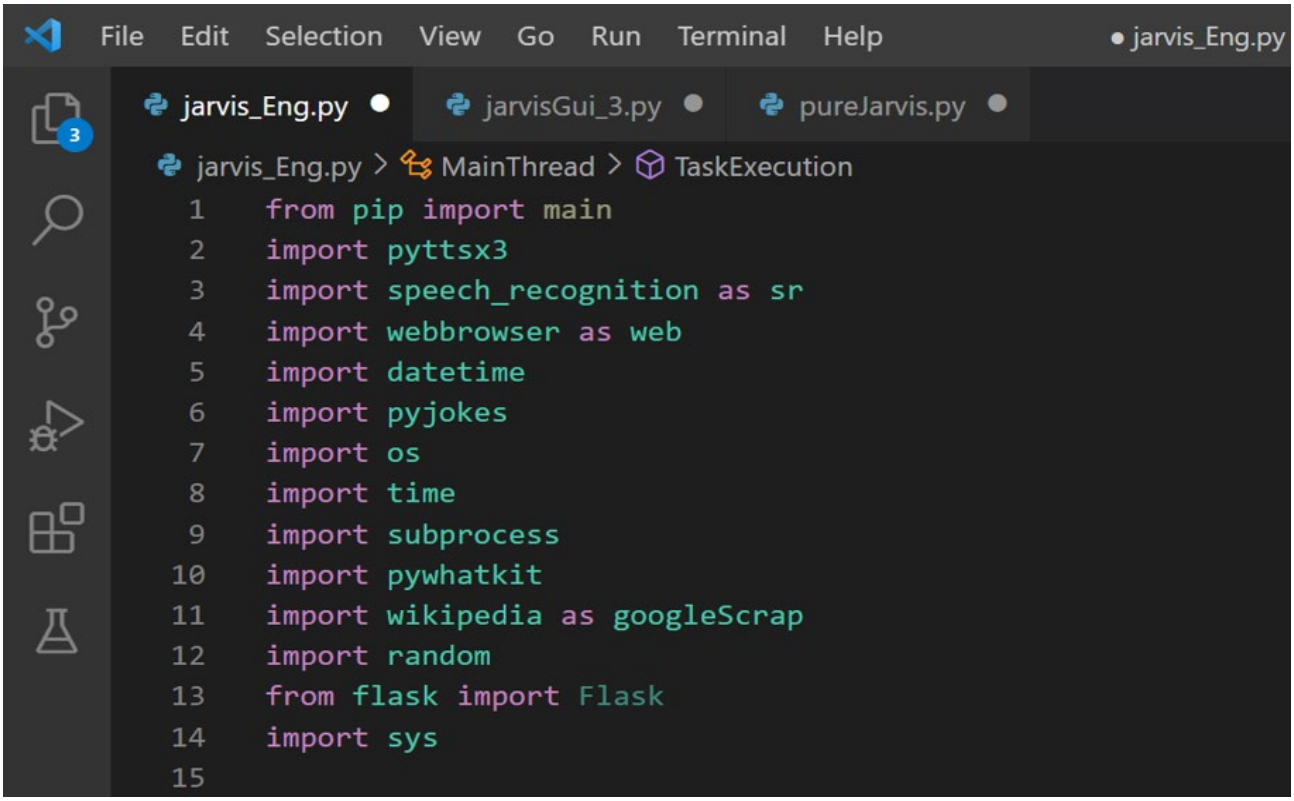

Figure 3.3 Imported Modules

Chapter 4: Implementation Work Details

JARVIS, a desktop assistant is a voice assistant that can perform many daily tasks of desktop like playing music, opening your favorite IDE with the help of a single voice command. Jarvis is different from other traditional voice assistants in terms that it is specific to desktop and user does not need to make account to use this, it also require internet connection while getting the instructions to perform any task.

4.1. REAL LIFE APPLICATION

4.1.1. Saves time: JARVIS is a laptop voice assistant which matches at the voice command supplied to it, it may do voice searching, voice-activated tool manipulate and can allow us to whole a fixed of tasks.

4.1.2. Conversational interplay: It makes it simpler to finish any project as it mechanically do it via way of means of the usage of the crucial module or libraries of Python, in a conversational interplay way. Hence any person while teach any project to it, they experience like giving project to a human assistant due to the conversational interplay for giving enter and getting the preferred output withinside the shape of project done.

4.1.3. Reactive nature: The laptop assistant is reactive this means that it know human language thoroughly and apprehend the context this is furnished via way of means of the person and offers reaction withinside the equal way, i.e. human comprehensible language, English. So person reveals its response in an knowledgeable and clever way.

4.1.4. Multitasking: The principal software of it may be its multitasking ability. It can ask for non-stop training one after different till the person “QUIT” it.

4.1.5. No Trigger section: It asks for the training and pay attention the reaction this is given via way of means of person with no need any cause section after which handiest executes the project.

4.2. DATA IMPLEMENTATION AND PROGRAM EXECUTION

As the first step, install all the necessary packages and libraries. The command used to install the libraries is “pip install” and then import it. The necessary packages included are as follows:

4.2.1. LIBRARIES AND PACKAGES

4.2.1.1. pyttsx3: It is a python library which converts text in to speech.

4.2.1.2. SpeechRecognition: It is a python module which converts speech in to the text.

4.2.1.3. pywhatkit: It is python library to send WhatsApp message at a particular time with some additional features like searching on web.

4.2.1.4. datetime: This library provides us the actual date and time in string format.

4.2.1.5. wikipedia: It is a python module for searching anything on Wikipedia.

4.2.1.6. pyjokes: It is a python libraries which contains lots of interesting jokes in it.

4.2.1.7. webbrowser: It provides interface for displaying web-based documents to users.

4.2.1.8. os: It represents Operating System related functionality.

4.2.1.9. sys: It allows operating on the interpreter as it provides access to the variables and functions that usually interact strongly with the interpreter.

4.2.1.10. smtplib: Simple mail transfer protocol that allows us to send mails and to route mails between mail servers.

4.2.2. FUNCTIONS

4.2.2.1. TakeCommand(): The function is used to take the command as input through microphone of user and returns the output as string.

4.2.2.2. For wishing: If you say “wish me” then it is wishing according to time like that Good Morning(<10 AM), Good noon(10 to 11:59 AM), Good Afternoon (12 to 4:59 PM) and Good Evening(5 to 11:59 PM).

4.2.2.3. TaskExecution(): This is the function which contains all the necessary task execution definition like open MS Excel, MS word, open whatsapp, open calculator, set alarm, and many conditions in if condition like “open google”, “open notepad”, “search on Wikipedia”, “play system music”, “play music online” and “open terminal” and open anything on web etc.

4.2.2.4. playAlarmSound(): This function is used to play specific alarm ringtone when alarm is set.

4.2.2.5. playMusic(): This function used for playing random song of System Music and printing all the songs those are present in the folder.
4.2.2.6. sendEmail(): This is used to sending Emails via Gmail.
4.2.2.7 speak(): This function used for speak anything.

Chapter 5: Source Code and Commands

5.1. Source Code:

You can see the full Python code (Two .py file Name: jarvis_Eng.py and jarvis_Gui3.py) for **"SMART VOICE ASSISTANT" (JARVIS)** on the given Google Drive link below…

Drive Link : https://drive.google.com/drive/folders/1LmBRPiNJzuj8bBetDcx5nKtKBC80x9ct?usp=sharing

5.2. Commands:

I'm giving some commands and it's responses here for using My "SMART VOICE ASSISTANT" (JARVIS).

Commands	Responses
open calculator	→ it will open calculator.
open ms word	→ it will open MS word.
open powerpoint	→ it will open PowerPoint.
open notepad	→ it will open Notepad.
open vs code	→ it will open VS code editor.
open firefox	→ it will open Firefox.
open paint	→ it will open Paint app.
play system music	→ it will play random song of system music and print all songs.
play music online	→ it will play online music on https://www.jiosaavn.com/
send email	→ it will ask what is message then it will sent email via Gmail.
set alarm	→ it will asking enter the time then alarm will set automatically
open terminal	→ it will open terminal.
tell me date	→ it will tell you the today date.
tell me time	→ it will tell you the time.
wish me	→ it will wishing according to time
weather report	→ firstly it will ask the city name then it will tell you the weather report.
<Anything> wikipedia	→ it will tell you two paragraph about < **Anything** > according to Wikipedia and also search on Google.
open google	→ it will asking what do you want to search on Google then it will search.

NOTE : If it doesn't understand your voice then it will exit automatically and understand something and those are not in our specific command then it will search on Google.

Partial Code of "SMART VOICE ASSISTANT" (JARVIS) without GUI

```
import pyttsx3
import speech_recognition as sr
import webbrowser as web
import datetime
import pyjokes
import os
import time
import subprocess
import pywhatkit
import wikipedia as googleScrap
import random

# Function for converting the voice into text form...
def TakeCommand():
    recognizer = sr.Recognizer()
    with sr.Microphone() as source:
        print("\nSay something... *****I'm listening here*****")
        recognizer.pause_threshold = 1
        recognizer.adjust_for_ambient_noise(source)
        audio = recognizer.listen(source)
        try:
            print("\nRecognizing...")
            data = recognizer.recognize_google(audio, language="en-in")
            return data
        except sr.UnknownValueError:
            print("\nCould not understand your voice.\nPlease speak carefully...\nThanks for using...\n")
            speak("Could not understand your voice.")
            speak("please speak carefully")
            speak("thanks for using")
            quit()

# Function for converting any text in to voice...
def speak(x):
    engine = pyttsx3.init("sapi5")
    voices = engine.getProperty("voices")
    engine.setProperty("voice", voices[0].id)
    rate = engine.getProperty("rate")
    engine.setProperty("rate", 150)
    print("  ")
    engine.say(x)
    engine.runAndWait()

def TaskExecution():

        # Declaring time and date here...
        Nowtime = datetime.datetime.now().strftime("%I : %M : %p")
        hr = int(Nowtime[0: 2])
        amPm = Nowtime[10: 12]
        date = datetime.datetime.now().strftime("Today date is : %B %d, %Y")
```

```
# For wishing according to time...
    if hr <= 9 and amPm == "AM":
        print("\nGood morning sir... its - " +Nowtime )
        speak("good morning sir")
        speak("its" + Nowtime)
    elif 10 <= hr <= 11 and amPm == "AM":
        print("\nGood noon sir... its - " + Nowtime)
        speak("good noon sir")
        speak("its" + Nowtime)
    elif (hr == 12 or 1 <= hr <= 4) and amPm == "PM":
        print("\nGood afternoon sir... its - " + Nowtime)
        speak("good afternoon sir")
        speak("its" + Nowtime)
    elif (5 <= hr <= 11) and amPm == "PM":
        print("\nGood evening sir... its - " + Nowtime)
        speak("good evening  sir")
        speak("its" + Nowtime)
    print("\n*************** INSTRUCTIONS ***************")
    print("1. The name of voice assistant : JARVIS")
    print("2. Please wait THREE seconds after every task")
    print("\nJARVIS here...\nhow may i help you?")
    speak("jarvis here")
    speak("how may i help you")

    while True:
        Nowtime = datetime.datetime.now().strftime("%I : %M : %p")
        hr = int(Nowtime[0: 2])
        amPm = Nowtime[10: 12]
        date = datetime.datetime.now().strftime("Today date is : %B %d, %Y")

        query = TakeCommand().lower()
        if "your name" in query:            # For speaking voice assistant name...
            print("You said : " + query)
            print("My answer : My name is jarvis")
            print("how may i help you?")
            speak("my name is jarvis")
            speak("how may i help you")
        elif "youtube" in query:        # For opening youtube...
            print("You said : " + query)
            web.open("https://www.youtube.com")
            print("\nMy answer : Ok sir Youtube has been opened please enjoy.")
            speak("ok sir")
            speak("youtube has been opend please enjoy.")

        elif "firefox" in query:         # For opening Firefox...
            print("You said : " + query)
            web.open("firefox.exe")
            print("\nMy answer : Ok sir Firefox has been opened please enjoy.")
            speak("ok sir")
            speak("firefox has been opend.")

        elif "gmail" in query:          # For opening gmail...
            print("You said : " + query)
            web.open("https://mail.google.com/mail/u/0/#inbox")
            print("\nMy answer : Ok sir Gmail has been opened")
            speak("ok sir")
            speak("gmail has been opened")
```

```
        elif "whatsapp" in query:       # For opening whatsapp...
                print("You said : " + query)
                web.open("https://web.whatsapp.com/")
                print("\nMy answer : Ok sir Whatsapp has been opened")
                speak("ok sir")
                speak("whatsapp has been opened")

        elif "music online" in query:    # For opening online music on jio saavn...
                print("You said : " + query)
                web.open("https://www.jiosaavn.com/")
                print("\nMy answer : Ok sir I have to play music on jio saavan for you")
                speak("ok sir")
                speak("i have to play music on jio saavan for you")
        elif "calculator" in query:     # For opening calculator...
                print("You said : " + query)
                subprocess.Popen("calc.exe")
                print("\nMy answer : Ok sir Calculator has been opened")
                speak("ok sir")
                speak("calculator has been opened")
        elif "time" in query:           # For opening and speaking time...
                print("You said : " + query)
                print("\nMy answer : Ok sir now the time = " + Nowtime)
                speak("now the time is" + Nowtime)
        elif "wikipedia" in query or "wiki" in query:       # For searching anything on wikipedia...
                query = query.replace("wikipedia", "")
                result = googleScrap.summary(query, 2)
                print("You said : " + query)
                print("\nAccording to wikipedia : " + result)
                try:
                   pywhatkit.search(query + "wikipedia")
                   speak("according to wikipedia" + result)
                except:
                   speak("this may not searchable data")
         elif “open google" in query or "google" in query: # For searching anything on google...
                print("You said : " + query)
                print("What do you want to search on google?")
                speak("What do you want to search on google")
                query = TakeCommand().lower()
                print("You said : " +query)
                pywhatkit.search(query)
                speak("your input has been opened on google")
        elif “exit” in query or "bye" in query or “quit” in query: # For Exit...
                print("You said : " + query)
                print("\nMy answer : Now Exit...\nOk bye...\nThanks for using\nHave a good day\n")
                speak("ok byee thanks for using")
                speak("have a good day")
                break

        else:                   # For searching anything on google...
                print("You said : " + query)
                pywhatkit.search(query)
                speak("these are results on web")

            time.sleep(3)


# Calling TaskExecution function here...
TaskExecution()
```

Chapter 6: Input/Output Screenshots

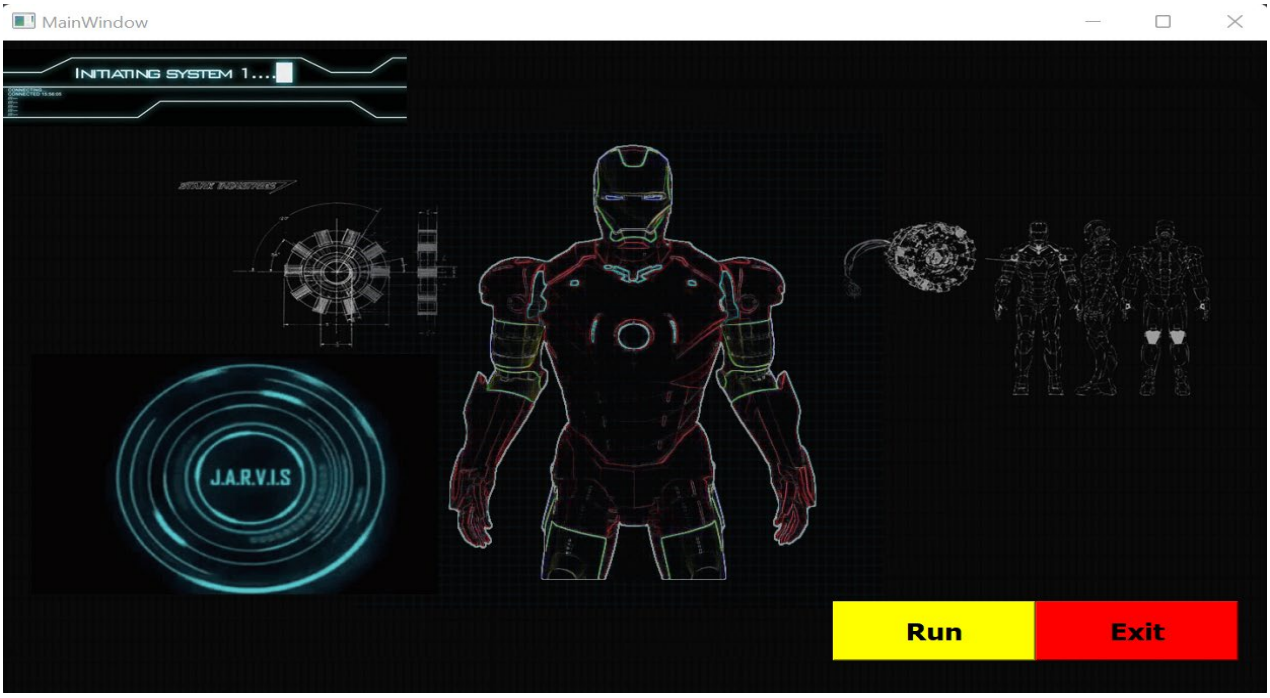

Figure 6.1 Live GUI of JARVIS

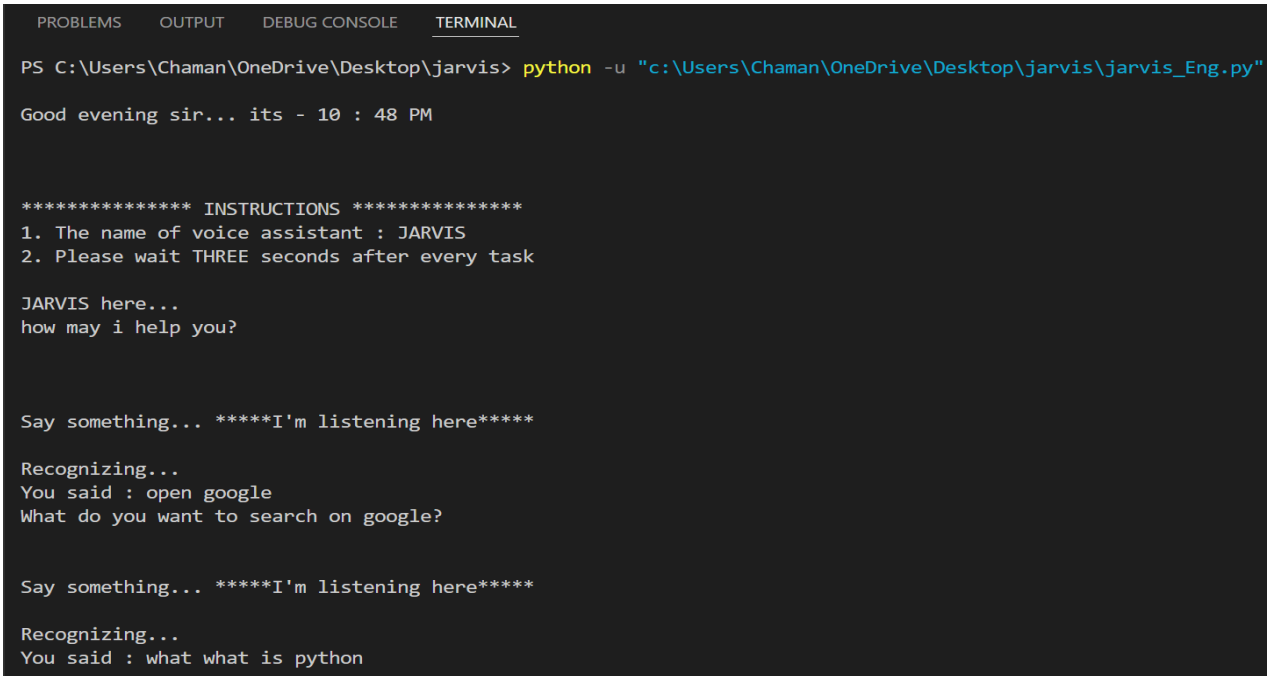

Figure 6.2 Input for Google Search

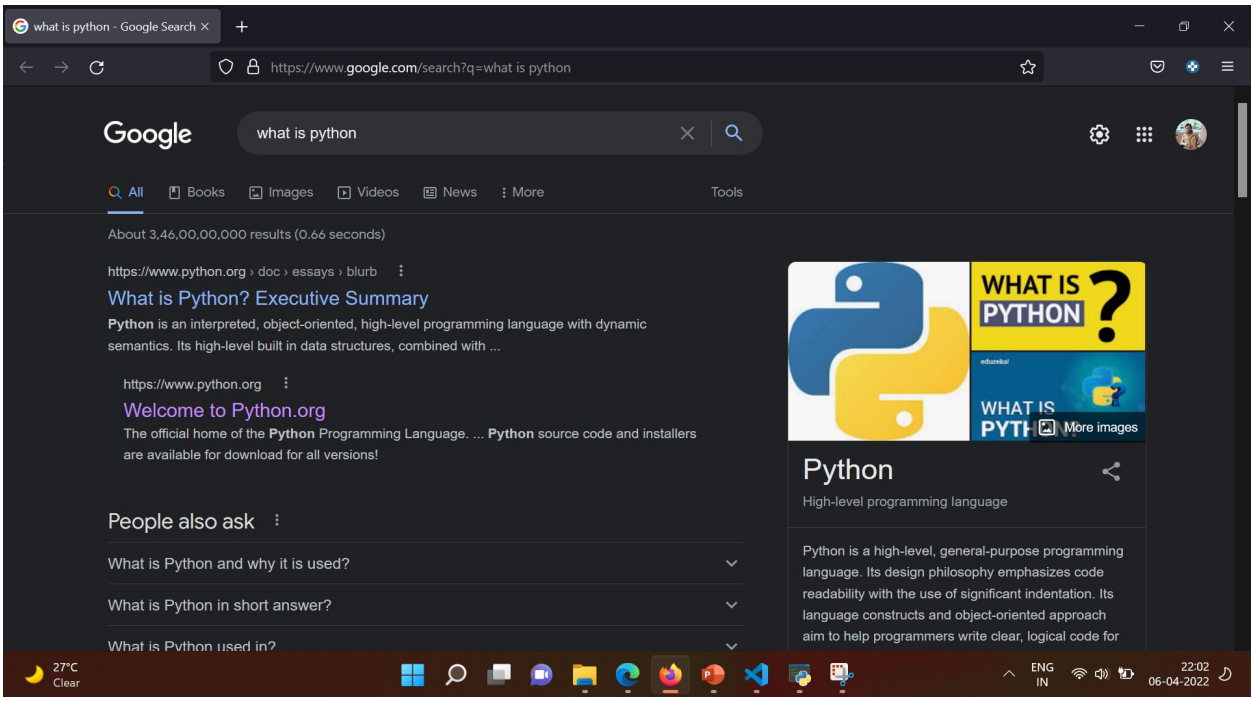

Figure 6.3 Output for Google Search

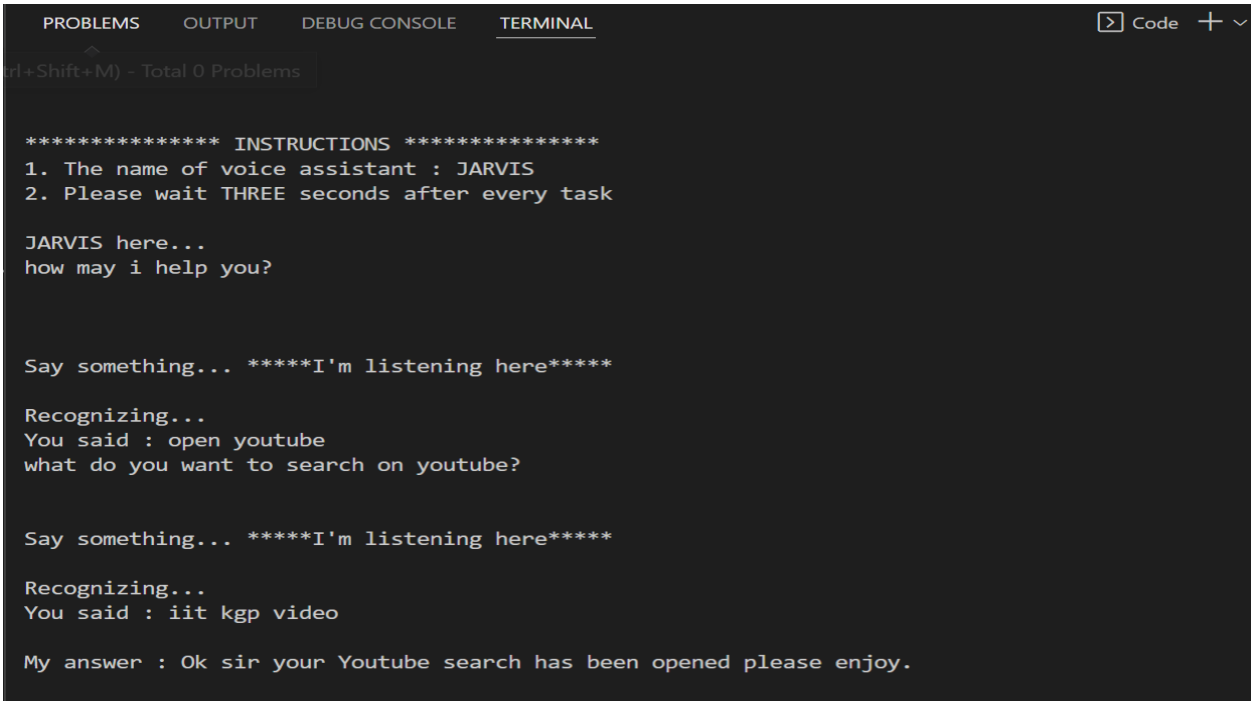

Figure 6.4 Input for YouTube Search

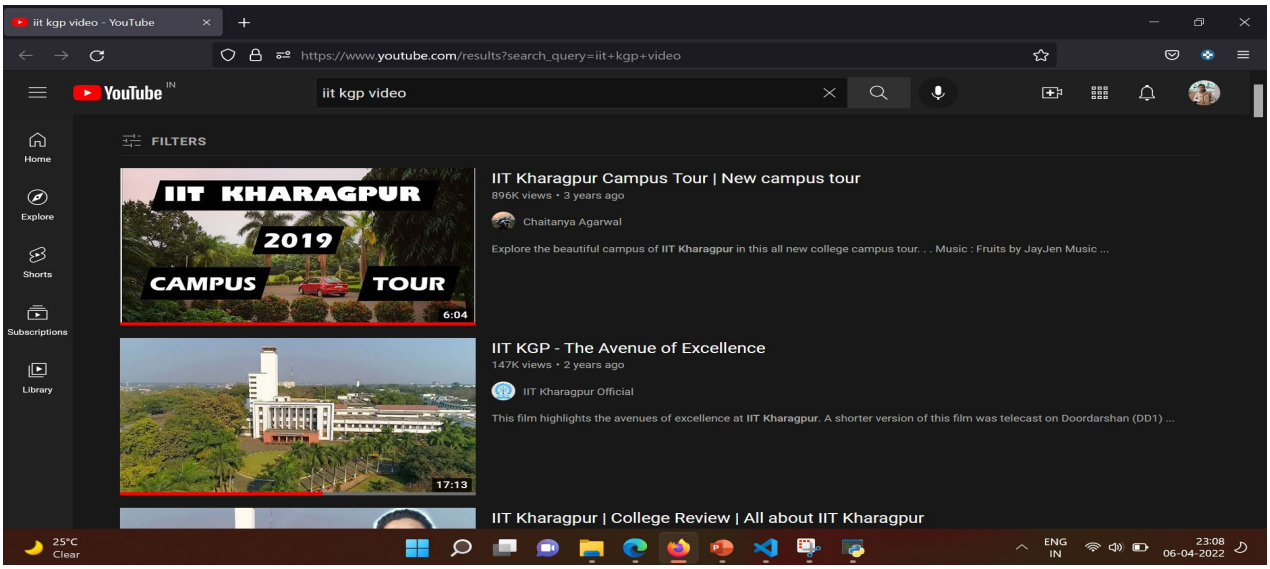

Figure 6.5 Output for YouTube search

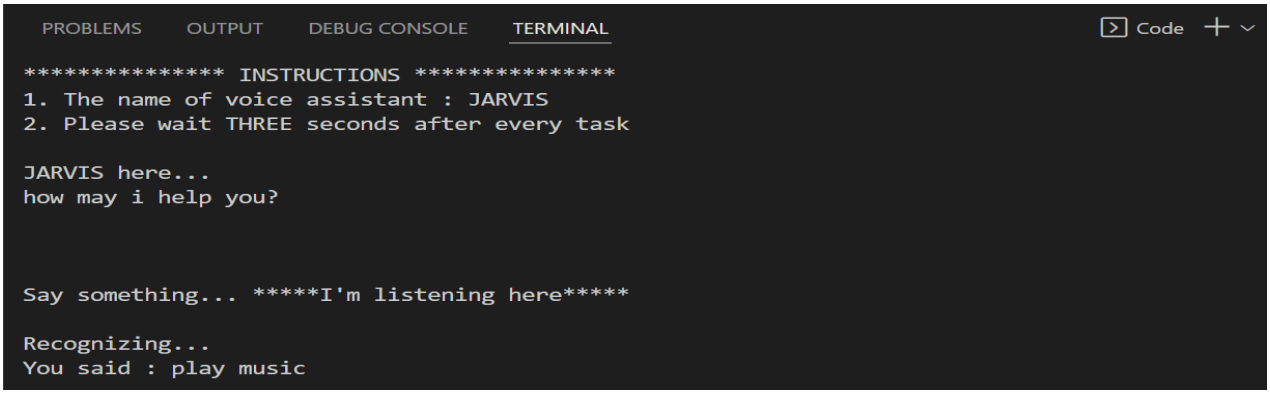

Figure 6.6 Input to play music

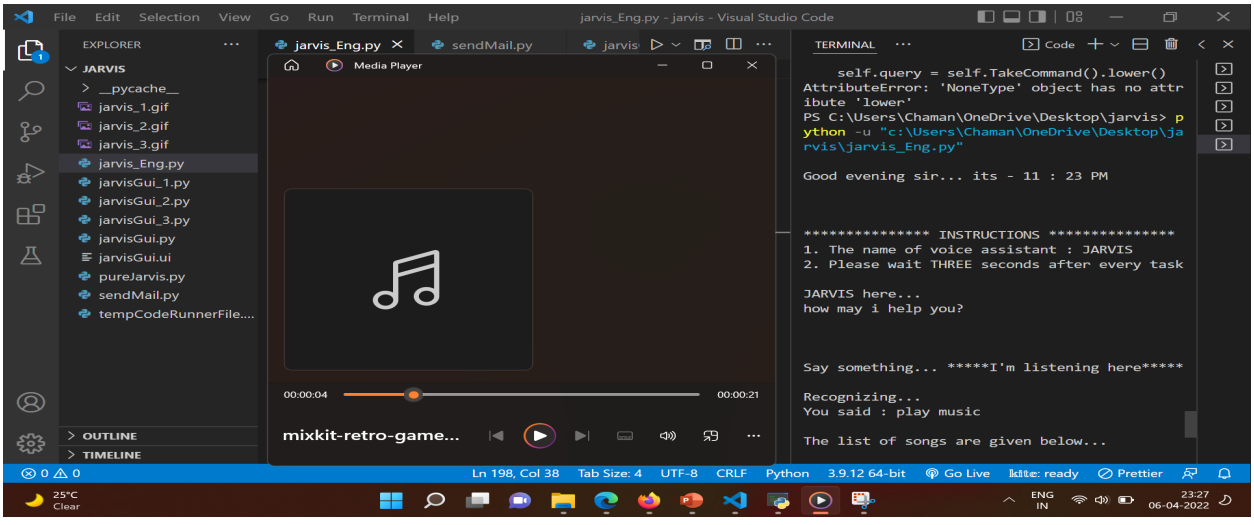

Figure 6.7 Output to play music

Chapter 7: System testing

The system testing is done on fully integrated system to check whether the requirements are matching or not. The system testing for **JARVIS** desktop assistant focuses on the following four parameters:

7.1. FUNCTIONALITY

In this we take a look at the capability of the device whether or not the device plays the challenge which it became meant to do. To take a look at the capability every feature became checked and run, if it could execute the desired challenge successfully then the device passes in that specific capability test. For example to check whether **JARVIS** can search on Google or not, as we can see in the **figure 6.2**, user said “Open Google”, then Jarvis asked, ”What do you want to search on Google?” then user said, “What is Python”, Jarvis open Google and searched for the required input (see output in the **figure 6.3)**.

7.2. USABILITY

Usability of a machine is checked via way of means of measuring the easiness of the software program and the way person pleasant it's far for the person to use, the way it responses to every question this is being requested via way of means of the person.

It makes it simpler to finish any assignment because it mechanically do it via way of means of the usage of the critical module or libraries of Python, in a conversational interplay way. Hence any person while coach any assignment to it, they sense like giving assignment to a human assistant due to the conversational interplay for giving enter and getting the favored output withinside the shape of assignment done.

The computing device assistant is reactive because of this that it realize human language thoroughly and recognize the context this is supplied via way of means of the person and offers reaction withinside the identical way, i.e. human comprehensible language, English. So person unearths its response in an knowledgeable and clever way.

The essential utility of it may be its multitasking ability. It can ask for non-stop coaching one after different till the person “QUIT” or “EXIT” it. It asks for the coaching and pay attention the reaction this is given via way of means of person without having any cause section after which most effective executes the assignment.

7.3. SECURITY

The protection trying out particularly makes a specialty of vulnerabilities and risks. As JARVIS is a neighborhood computing device application, as a result there's no chance of statistics breaching thru far off access. The software program is devoted to a selected device so whilst the consumer logs in, it'll be activated.

7.4. STABILITY

Stability of a system depends upon the output of the system, if the output is bounded and specific to the bounded input then the system is said to be stable. If the system works on all the poles of functionality then it is stable.

Chapter 8: Individual Contribution

The project titled **"SMART VOICE ASSISTANT" (JARVIS)** was designed by me individually. From installing of all the packages, importing, creating all the necessary functions, designing GUI in PyQt5 and connecting that live GUI with the backend, was all done by me individually.

I, myself have done all the research before making this project, designed the requirement documents for the requirements and functionalities, wrote synopsis and all the documentation, code and made the project in such a way that it is deliverable at each stage. I have created the front-end (.ui file) of the project using PyQt5 Designer, the front-end comprises of a live GUI and is connected with the **.py** file which contains all the classes and packages of the **.ui** file. The live GUI consists of moving GIFs which makes the front-end attractive and user friendly.

I have written the complete code in Python language and in Visual Studio Code IDE from where it was very easy to install the packages and libraries, I have created the functions like TakeCommand(), sendEmail() and TaskExecution() which has the following functionalities, like TakeCommand() which is used to take the command as input through microphone of user and returns the output as string, sendEmail() which is used to send Email automatically via Gmail and TaskExecution() which contains all the necessary task execution definition like sendEmail(), playMusic(), playAlarmSound() and many conditions in if condition like "open Google", "open notepad", "search on google", "open youtube" ,"play music" and "open terminal" etc.

While making this project I realized that with the advancement **JARVIS** can perform any task with same effectiveness or can say more effectively than us. By making this project, I realized that the concept of AI in every field is decreasing human effort and saving time. Functionalities of this project include, It can send emails, It can open Excel, It can open WhatsApp, It can open terminal, It can open your favorite IDE like Visual Studio code, Sublime Text, Notepad etc., It can play music, It can do Wikipedia searches for you, It can open websites like Google, YouTube, etc., in a web browser, It can have some basic conversation.

At last, I have updated my report and completed it by attaching all the necessary screen captures of inputs and outputs, mentioning the limitations and scope in future of this project.

Chapter 9: Conclusion

JARVIS is a very useful voice assistant without any doubt as it saves time of the user by conversational interactions, its effectiveness and efficiency. But while working on this project, there were some limitations encountered and also realized some scope of enhancement in the future which are mentioned below:

9.1. LIMITATIONS

9.1.1. Security is somewhere an issue, there is no voice command encryption in this project.

9.1.2. Background voice can interfere.

9.1.3. Misinterpretation because of accents and may cause inaccurate results.

9.1.4. JARVIS can't be referred to as externally every time like different conventional assistants like Google Assistant may be referred to as simply via way of means of saying, “Ok Google!”

9.2. SCOPE FOR FUTURE WORK

9.2.1. Make JARVIS to learn more on its own and develop a new skill in it.

9.2.2. JARVIS android app can also be developed.

9.2.3. Make more Jarvis voice terminals.

9.2.4. Voice commands can be encrypted to maintain security.

References

- **VOICE ACTIVATED DESKTOP ASSISTANT USING PYTHON (Dhiraj Pratap Singh, Deepika Sherawat, Sonia) School of Computer Science & Engineering, Galgotias University Greater Noida-203201**
- **https://www.youtube.com/c/wscubetechjodhpur**
- **https://www.youtube.com/c/AviUpadhyay**
- **https://www.geeksforgeeks.org/**
- **https://docs.python.org/3/library/index.html**
- **https://www.stackoverflow.com**

Thank You