A.L.A.I.N.A. - A Marvel Movie Inspired Voice Assistant for the Open- Source Community

Author: Omkar Kadam[1], Rohan Kelshikar[2], Asst. Prof. Odilia Gonsalves[3]

[1,2]Student, Department of Information and Technology, Atharva College of Engineering, Mumbai, Maharashtra, India.
[3]Assistant Professor, Department of Information and Technology, Atharva College of Engineering, Mumbai, Maharashtra, India.

---***---

Abstract- *We all have voice assistants in our phones like Siri/Alexa/ Google Assistant, but to use any of these voice assistants we need some or the other device and basically, we have to buy some device to use them. ALAINA would be a completely open source and available to all kinds of people, be it a developer who wants to integrate their IOT devices with an AI system without buying an ALEXA enabled device or a user with no device who wants to integrate their inventions with an AI voice assistant system. ALAINA supports most languages spoken and has space to add various types of custom commands for the user, as it will be available to anyone who has the knowledge of python and the source code is available for anyone to modify and customize according to their own will, ensuring complete customization and freedom to play with it.*

ALAINA uses keyword based neural network architecture, so basically put, it will analyze the keywords in the sentences to provide the user with the feedback and actions based on the user's commands. To keep the coding redundancy to a minimum, this approach was used along with bag of words algorithm.

Index Terms- **Neural Network, Bag of words, Voice assistant, Artificial Intelligence, Speech-to-text, Speech-to-action.**

I. INTRODUCTION

As we know, the last few years, from 2015-till date, we have had a lot of ways to communicate with a computer/smartphone/IOT.

Privacy related issues:

The user can use ALAINA as their own server controller and enable access through all devices that the user owns and does not have to buy ALEXA enabled device for it, which no one knows how much data mining and information leak of each user is done by amazon, google, Facebook. It is an open secret that Facebook steals data and sells it to corporates so they can target their advertisement to specific users to influence, half of which is done by our voice assistants and MIC's embedded in the devices that we buy. Privacy is breached at the cost of convenience. This is what ALAINA intends to avoid and keep the data of the user with devices with ease by using voice assistants and Speech recognition software's. But for any of those software's to run, we need expensive software's that cost around 6000-8000 Rs. Minimal, while on the other hand devices like Arduino, Raspberry PI can range from 1000-2000 Rs, using which a person can integrate his own system with some other device for free, reducing their cost of prototyping and experimentation. But none of the modules have AI based voice assistant, they need to be built, so ALAINA aims to be the base of operations for such PROTOTYPE applications, experiments and also be used with various ANDROID and WINDOWS/MAC if they user wants to implement it.

Problem Statement:

Availability issues:

The only issue with ALEXA or SIRI is that they are premium products that only have the ability to do what's attached to it across AMAZON's web service and not a completely OPEN-source design, which is fair enough but the normal user, low cost/ budget user won't have any access to what can and cannot be done. So, ALAINA aims to fill that void by enabling any user to use her base code and modify it the way any user wants to in their server instead of using AMAZON's or APPLE's cloud service. Just like in the marvel movies, Friday can be accessed by only two people, based on voice commands, the whole organization runs with the help of the AI used by tony stark, similarly we aspired to find the fine line between privacy and open source. Themselves, while doing everything the user wants, exactly the way they want it and without any privacy breach.

Reason to be open source:

ALAINA was made keeping in the mind, the common user, the naïve user, the basic developer and also the intrigued individual and curious minds with no resources, which is why we decided to keep ALAINA open source and made sure that it is available for prototyping. Any user can add their own custom commands to add more functionality making ALAINA even smarter day-by-day.

II. STUDIES AND FINDINGS

ALAINA was made using basics of python, its vast libraries, and use of KERAS and TensorFlow, NLTK package of python the current generation of AI related packages that help in using neural networks, and architecture. ALAINA itself stands for **All Language** processing **Artificial Intelligence** using **Neural Networks.**

There have been many issues regarding lagging of text to speech conversion, audio delay and no response from the system altogether due to the vast amount of neural processing that works in the background, in low powered processors, hence we had to tweak the whole system from being completely neural architecture based to just using bag of words algorithm and using keywords to define commands given to ALAINA.

Other voice assistant systems usually work with mobile devices, ALAINA was kept simple so every developer can make it for any kind of device, and be it IOT, Windows, Linux, and Android installed on Raspberry PI, custom android ROM based systems.

Most of the AI voice assistant systems are based on the user input and will just return the data asked for through Google, or prebuilt app installed in the device or what they are coded upon and cannot be integrated with more functions as its owned by their respective companies, but with ALAINA we have the freedom to do whatever we want it to act like and/or respond to us while performing the functions we want it to. Nonetheless, all the current voice assistant systems are one of the most loved and used systems, but they can do much more than they do if they were open source, which is what we plan to do with ALAINA.

The current libraries i.e., NLTK, has embedded iNLTK in it, due to which the process of understanding Indian accents and speech becomes easier for the program to understand and bridge the gap between human and machine communication. Alaina has been made open source for the very purpose that every programmer can have the power to define their own functions for ALAINA to perform the custom functions that every user can get for their own, by just coding the program in the existing function. As Alaina even has a way to perform inside programs, we could be looking into coding without using our hands in VS Code, if we choose to do so, in that way we could be just speaking the code and then ALAINA would write it down, of course we made sure that there is human intervention so as to not be able to find errors for ourselves and rectify.

The base code will be provided to everyone for the fact that open- source community, filled with millions of programmers can just add their code snippets, which will add functionality to ALAINA even more. There is a possibility that other user's code can create malfunctions in the main source code, which is why the main source code will be made available but not editable online, only iterations of the code, when verified by the code author will be published back in the form of upgrades or iterations.

ALAINA being made on python is a major necessity, as the AI part of Alaina is basically using TensorFlow and Keras and CuNN- Cuda Neural Network package by Nvidia for its neural network architecture, due to which the ever-evolving libraries by Nvidia, python community will always be helpful in further upgrading ALAINA by the open-source community.

Because of this, the control is always in the hands of the users, and also the means to add the variations of problems that she can solve, and a solution based on the problem. If there is an issue, or something ALAINA is not able to perform or does not understand the user's query, it shall repeat the user with the query and the output has been made to feel like a normal human asking a question.

III. Literature Surveyed.

Our first survey was done on paper Human-Chatbot Interaction using NLTK. This paper proposes the mechanism of creating knowledgeable chat application which cannot permit the user to send inappropriate or improper messages to the participants by incorporating bottom level implementation of natural language processing (NLP). We have implemented INLTK, KERAS Packages in our project by referring this paper.

The next Survey was done on paper Chatbot Design Techniques in Speech Conversation Systems. In this paper, a survey of Chatbot design techniques in speech conversation between the human and the computer is presented. We have referred this paper to implement NLTK and KERAS package in our project so that voice of chatbot feel like real human and not just talking to robot.

The last paper we surveyed was Chatbots and Virtual Assistant in Indian Banks. The main motive of choosing this paper was it was an attempt to analyze the different chatbots used by different Indian banks with a view to answer above research questions. The focus has been only on Indian private and public sector banks. We have referred this paper to get to know how the user can fire p query and what are possible answers we can provide to user query along with keeping user engage in using our APP.

IV. Design Methodology

The design methodology is based on the aspect of Neural Networks used in the basic techniques of AI application. Neural Networks and Bags of words techniques are used to build up the neural architecture of ALAINA, which helps it in understand the basic way to respond to a query to the user.

We have used Selenium as a secondary suite to interact with various functionalities, like tracking IP, interacting with web- based controls, and all major components related to web activities.

ALAINA was made in python for the very reason that python is a multi-platform software, so it works with any kind of system, not just windows, but also Android and Ubuntu. With some tinkering to the source code, it can even work with Raspberry Pi and IOT devices. We can embed it with Raspberry pi / Arduino. As engineers, anyone who is a fan of Iron Man from the movies, they would love to have a voice assistant that not only listens to them, or responds, but does everything around the house with the help of just a voice command. By enabling it to be embedded to an IOT device and Blynk App, it is possible to control all the devices in your house with your voice by embedding ALAINA in the IOT device.

ALAINA was made open source and in python to enable the users/ the open-source community have the ease to make their own devices without worrying or depending on some paid company devices and be able to embed it in anything they invent, reinvent. Discover. ALAINA can be the way to enable home automation devices with their own embedded voice assistant.

IF enabled on to a cloud server, you can control everything in your house, your devices, your PC's, all the devices you own with your own custom commands.

V. Hardware and Software Used

HARDWARE	SOFTWARE
1. Windows 10 Operating System compatible system.	1. Python 3.7
	2. Keras and TensorFlow package for neural networking of understanding the query.

Windows 10 Operating System compatible system is used to check and deploy code.

Python - We are using Python 3.7, as 3.8 and 3.9 versions don't work well with TensorFlow modules currently, 3.7 is a more stable version of Python as of when this project was made.

Keras - Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent & simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear & actionable error messages. It also has extensive documentation and developer guides. So we are using Keras API for reducing the load and also making sure the learning process is iterated and rechecked for maximum understanding by the Voice assistant.

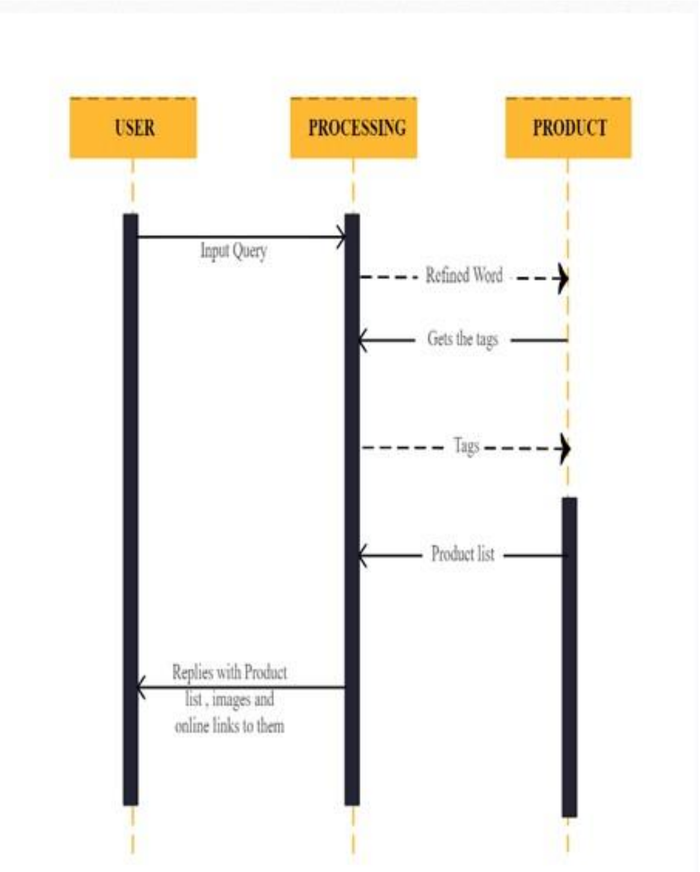
TensorFlow - TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

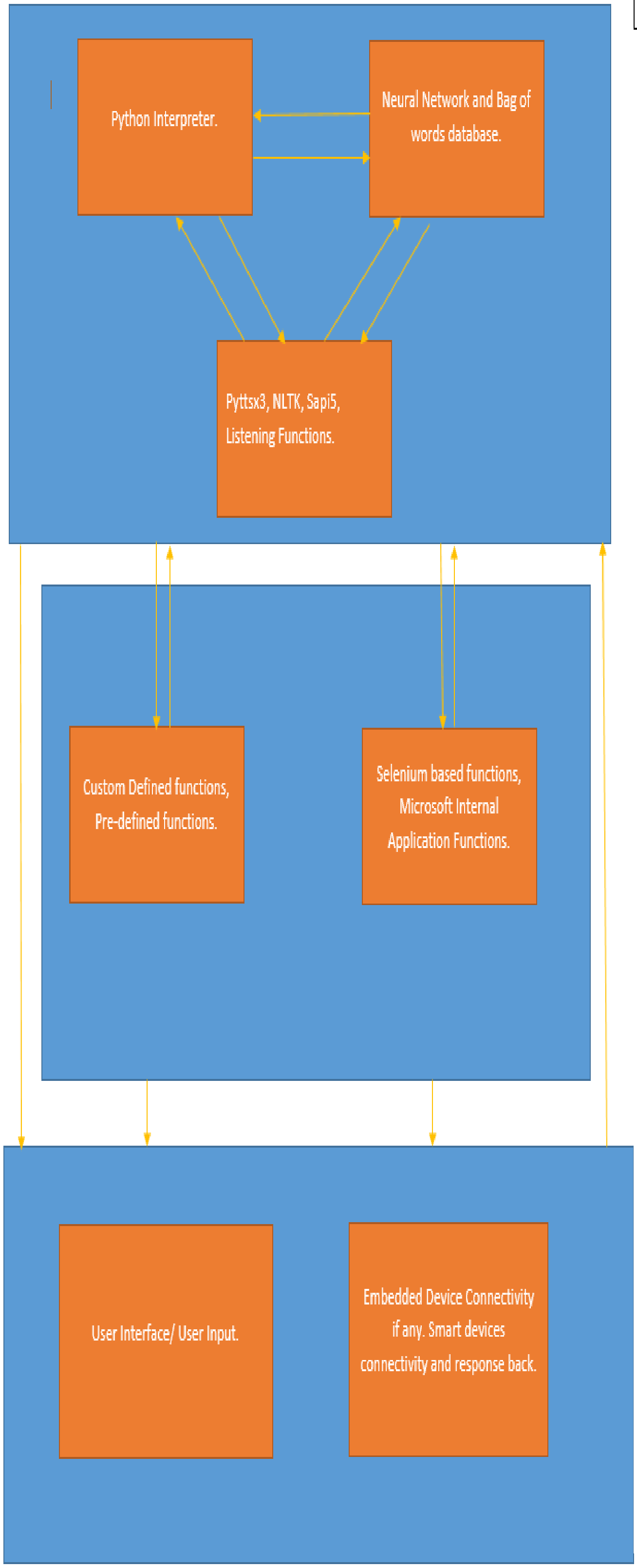
The core open-source library to help you develop and train ML models.

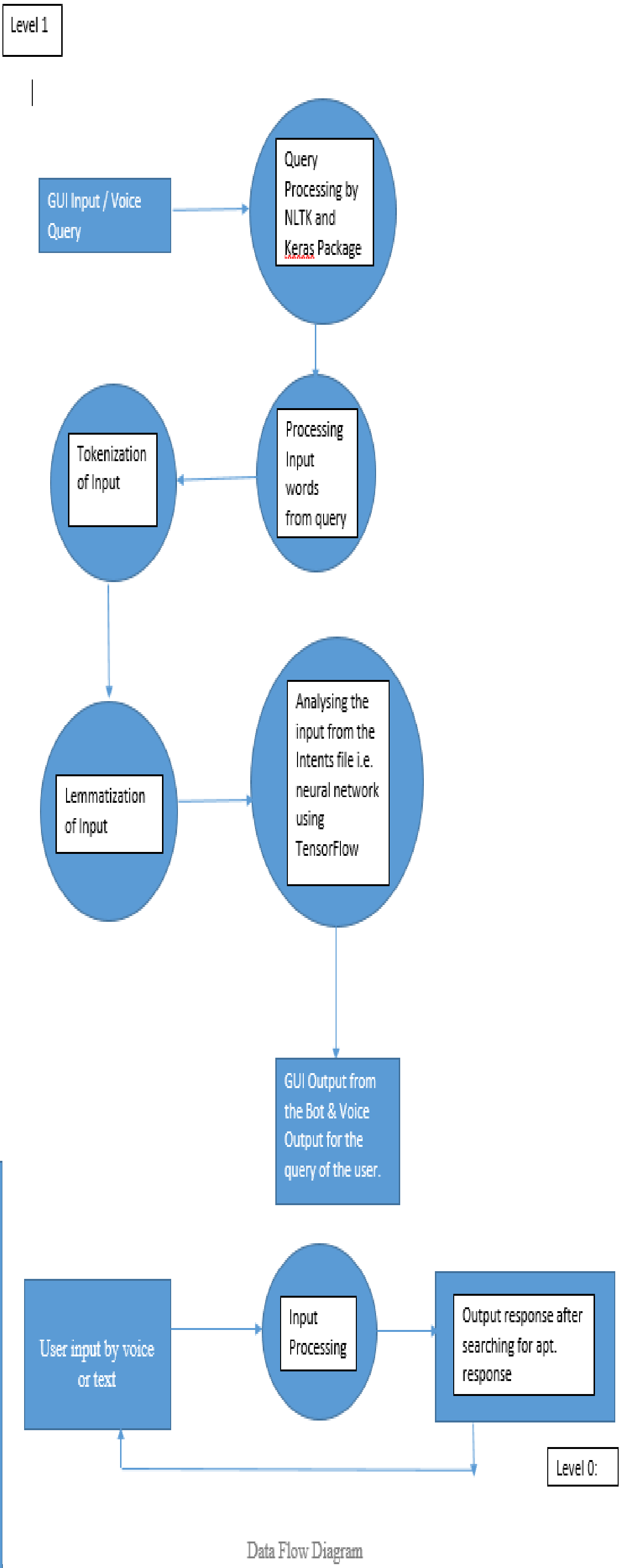
Visual Studio Code - Visual Studio Code is a **streamlined code- editing software** that supports development operations like debugging, task running, and version control. It aims to provide just the tools that you would need for a **quick code-build-debug cycle** and it leaves the more complex workflows to fuller featured IDEs like Visual Studio. Thus, it's much easier and less of a hassle to program your work—plus, you can pretty much **customize your interface** the way you want.

VI. Design Process, Data Flow, Sequence Diagram of the internal processing:

Sequence Diagram:




Python Interpreter.
Neural Network and Bag of words database.
Pyttsx3, NLTK, Sapi5, Listening Functions.
Custom Defined functions, Pre-defined functions.
Selenium based functions, Microsoft Internal Application Functions.
User Interface/ User Input.
Embedded Device Connectivity if any. Smart devices connectivity and response back.

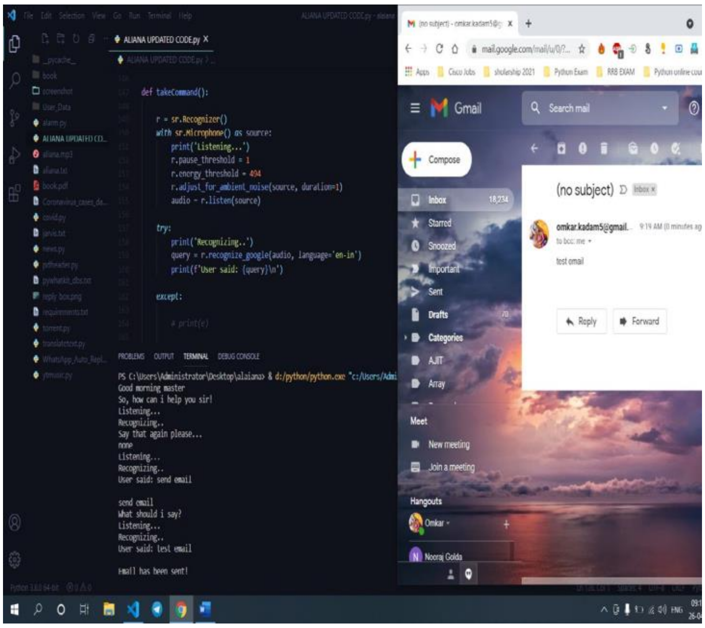

Level 1
GUI Input / Voice Query
Query Processing by NLTK and Keras Package
Processing Input words from query
Tokenization of Input
Lemmatization of Input
Analysing the input from the Intents file i.e. neural network using TensorFlow
GUI Output from the Bot & Voice Output for the query of the user.
User input by voice or text
Input Processing
Output response after searching for apt. response
Level 0:
Data Flow Diagram

Design Process.

Data Flow Diagram

VII. Screenshots of Our App:

1. Sending Email: For sending email we have used open source library of python stmplib library.



```
def takeCommand():

    r = sr.Recognizer()
    with sr.Microphone() as source:
        print('Listening...')
        r.pause_threshold = 1
        r.energy_threshold = 494
        r.adjust_for_ambient_noise(source, duration=1)
        audio = r.listen(source)

    try:
        print('Recognizing..')
        query = r.recognize_google(audio, language='en-in')
        print(f'User said: {query}\n')

    except:

        # print(e)
```

```
PS C:\Users\Administrator\Desktop\alaiana> & d:/python/python.exe "c:/Users/Admi
Good morning master
So, how can i help you sir!
Listening...
Recognizing..
Say that again please...
none
Listening...
Recognizing..
User said: send email

send email
What should i say?
Listening...
Recognizing..
User said: test email

Email has been sent!
```

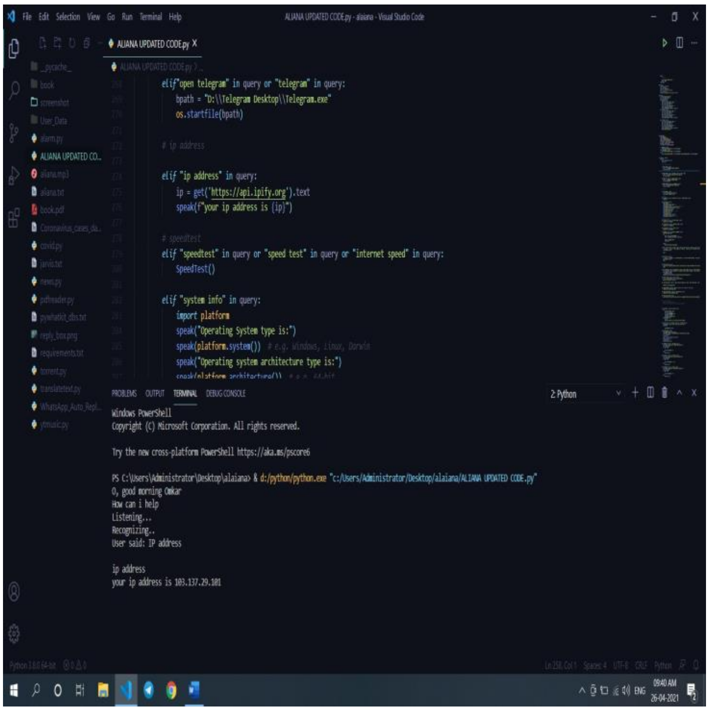
(no subject)

omkar.kadam5@gmail...

to bcc: me

test email

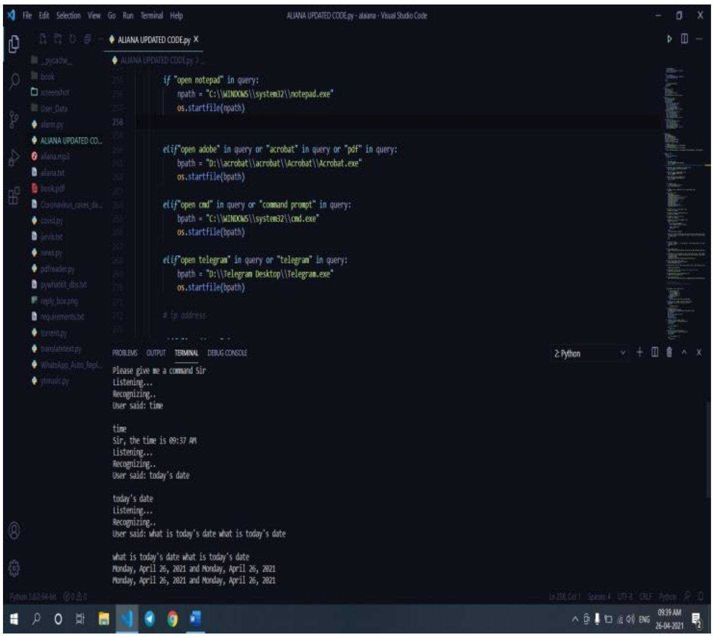
2. Opening of Notepad: To open and close System and Third-party apps in our ALIANA we have used OS module of python.



```
        elif"open telegram" in query or "telegram" in query:
            bpath = "D:\\Telegram Desktop\\Telegram.exe"
            os.startfile(bpath)

        # ip address

        elif "ip address" in query:
            ip = get('https://api.ipify.org').text
            speak(f"your ip address is {ip}")

        # speedtest
        elif "speedtest" in query or "speed test" in query or "internet speed" in query:
            SpeedTest()

        elif "system info" in query:
            import platform
            speak("Operating System type is:")
            speak(platform.system())  # e.g. Windows, Linux, Darwin
            speak("Operating system architecture type is:")
            speak(platform.architecture())  # e.g. 64-bit
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Desktop\alaiana> & d:/python/python.exe "c:/Users/Administrator/Desktop/alaiana/ALIANA UPDATED CODE.py"
O, good morning Omkar
How can i help
Listening...
Recognizing..
User said: IP address

ip address
your ip address is 103.137.29.101
```

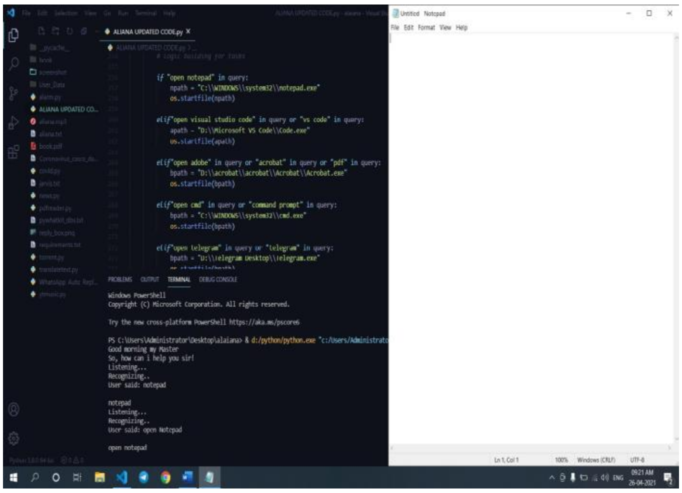
3. Date and time: To implement this we have used date and time module of python.


File Edit Selection View Go Run Terminal Help
ALIANA UPDATED CODE.py - alaiana - Visual Studio Code
ALIANA UPDATED CODE.py
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
2: Python
09:39 AM
26-04-2021

```
if "open notepad" in query:
    npath = "C:\\WINDOWS\\system32\\notepad.exe"
    os.startfile(npath)


elif"open adobe" in query or "acrobat" in query or "pdf" in query:
    bpath = "D:\\acrobat\\acrobat\\Acrobat\\Acrobat.exe"
    os.startfile(bpath)

elif"open cmd" in query or "command prompt" in query:
    bpath = "C:\\WINDOWS\\system32\\cmd.exe"
    os.startfile(bpath)

elif"open telegram" in query or "telegram" in query:
    bpath = "D:\\Telegram Desktop\\Telegram.exe"
    os.startfile(bpath)

# ip address
```

```
Please give me a command Sir
Listening...
Recognizing..
User said: time

time
Sir, the time is 09:37 AM
Listening...
Recognizing..
User said: today's date

today's date
Listening...
Recognizing..
User said: what is today's date what is today's date

what is today's date what is today's date
Monday, April 26, 2021 and Monday, April 26, 2021
Monday, April 26, 2021 and Monday, April 26, 2021
```

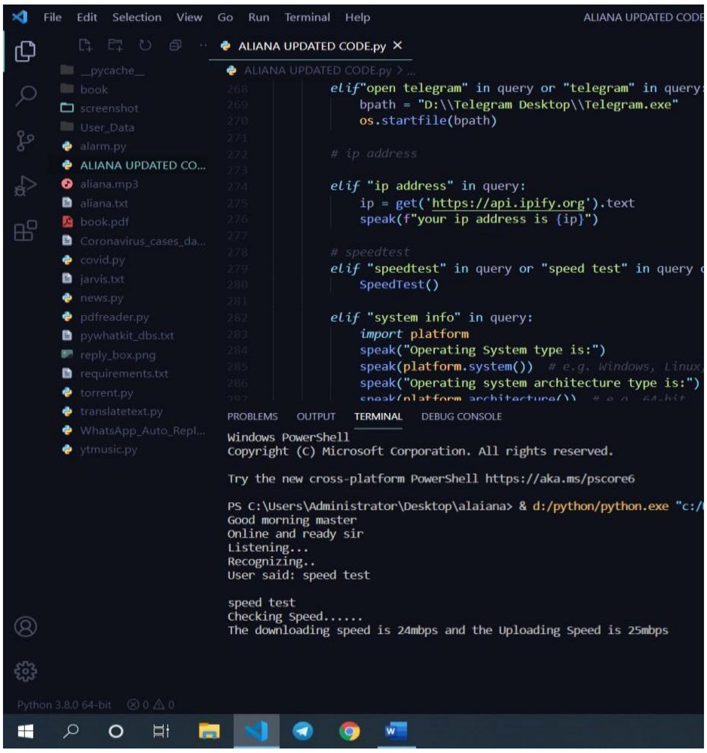
4. IP Address: Here we have used api of ipfy to find ip address of our system.


ALIANA UPDATED CODE.py
Untitled - Notepad
File Edit Format View Help
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
09:21 AM
26-04-2021

```
# logic building for tasks

if "open notepad" in query:
    npath = "C:\\WINDOWS\\system32\\notepad.exe"
    os.startfile(npath)

elif"open visual studio code" in query or "vs code" in query:
    apath = "D:\\Microsoft VS Code\\Code.exe"
    os.startfile(apath)

elif"open adobe" in query or "acrobat" in query or "pdf" in query:
    bpath = "D:\\acrobat\\acrobat\\Acrobat\\Acrobat.exe"
    os.startfile(bpath)

elif"open cmd" in query or "command prompt" in query:
    bpath = "C:\\WINDOWS\\system32\\cmd.exe"
    os.startfile(bpath)

elif"open telegram" in query or "telegram" in query:
    bpath = "D:\\Telegram Desktop\\Telegram.exe"
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Desktop\alaiana> & d:/python/python.exe "c:/Users/Administrato
Good morning my Master
So, how can i help you sir!
Listening...
Recognizing..
User said: notepad

notepad
Listening...
Recognizing..
User said: open Notepad

open notepad
```

5. Do a Speed test: To do speed test we have used both os module and API of speedtest.

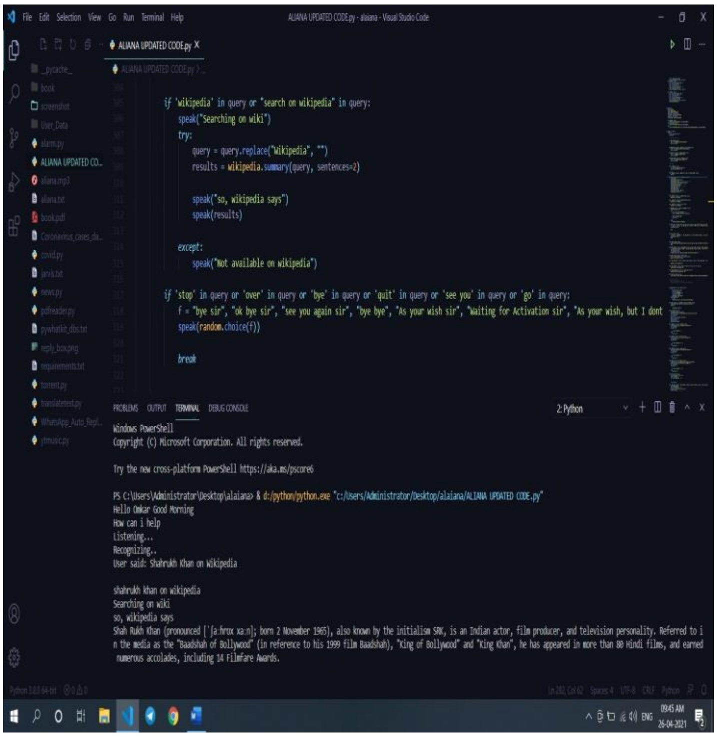


```
        elif"open telegram" in query or "telegram" in query
            bpath = "D:\\Telegram Desktop\\Telegram.exe"
            os.startfile(bpath)

        # ip address

        elif "ip address" in query:
            ip = get('https://api.ipify.org').text
            speak(f"your ip address is {ip}")

        # speedtest
        elif "speedtest" in query or "speed test" in query o
            SpeedTest()

        elif "system info" in query:
            import platform
            speak("Operating System type is:")
            speak(platform.system())  # e.g. Windows, Linux,
            speak("Operating system architecture type is:")
            speak(platform.architecture())  # e.g. 64-bit
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Desktop\alaiana> & d:/python/python.exe "c:/
Good morning master
Online and ready sir
Listening...
Recognizing..
User said: speed test

speed test
Checking Speed......
The downloading speed is 24mbps and the Uploading Speed is 25mbps
```

6. System information: To get System information we have used python OS module.

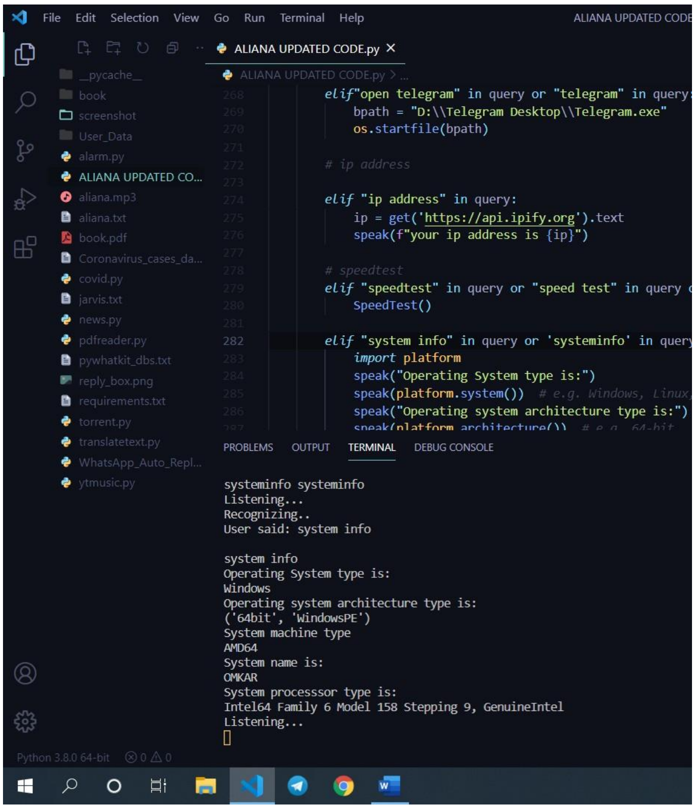
7. Search anything using Wikipedia: to do search on Wikipedia we have used Wikipedia API.



```
    if 'wikipedia' in query or "search on wikipedia" in query:
        speak("Searching on wiki")
        try:
            query = query.replace("Wikipedia", "")
            results = wikipedia.summary(query, sentences=2)

            speak("so, wikipedia says")
            speak(results)

        except:
            speak("Not available on wikipedia")

    if 'stop' in query or 'over' in query or 'bye' in query or 'quit' in query or 'see you' in query or 'go' in query:
        f = "bye sir", "ok bye sir", "see you again sir", "bye bye", "As your wish sir", "Waiting for Activation sir", "As your wish, but I dont
        speak(random.choice(f))

        break
```

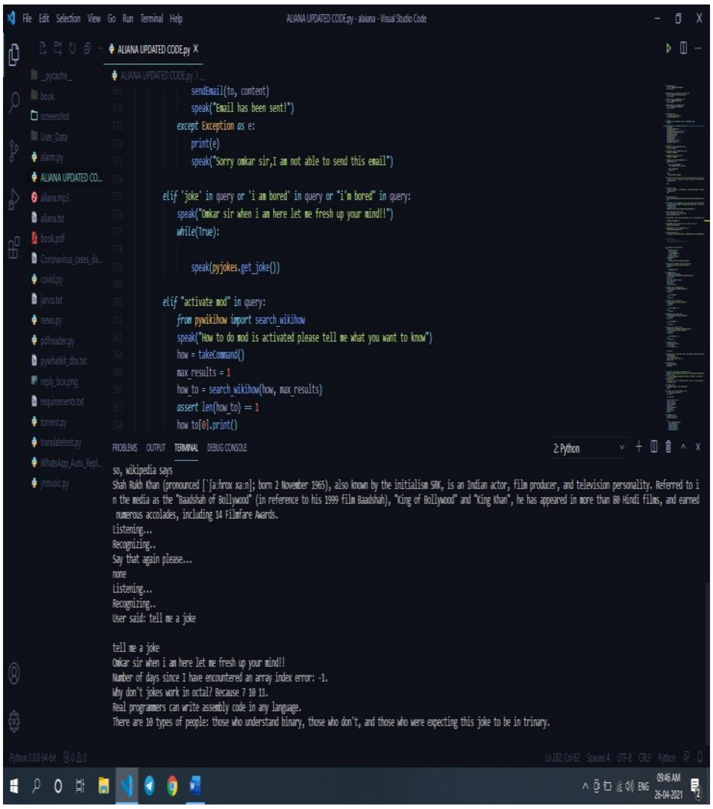
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Desktop\alaiana> & d:/python/python.exe "c:/Users/Administrator/Desktop/alaiana/ALIANA UPDATED CODE.py"
Hello Omkar Good Morning
How can i help
Listening...
Recognizing..
User said: Shahrukh Khan on Wikipedia

shahrukh khan on wikipedia
Searching on wiki
so, wikipedia says
Shah Rukh Khan (pronounced [ˈʃaːɦrʊx xaːn]; born 2 November 1965), also known by the initialism SRK, is an Indian actor, film producer, and television personality. Referred to i
n the media as the "Baadshah of Bollywood" (in reference to his 1999 film Baadshah), "King of Bollywood" and "King Khan", he has appeared in more than 80 Hindi films, and earned
 numerous accolades, including 14 Filmfare Awards.
```

7. Tell me a Joke: For telling jokes to user, we have used pyjoke module in ALIANA app.



```
elif"open telegram" in query or "telegram" in query
    bpath = "D:\\Telegram Desktop\\Telegram.exe"
    os.startfile(bpath)

# ip address

elif "ip address" in query:
    ip = get('https://api.ipify.org').text
    speak(f"your ip address is {ip}")

# speedtest
elif "speedtest" in query or "speed test" in query
    SpeedTest()

elif "system info" in query or 'systeminfo' in query
    import platform
    speak("Operating System type is:")
    speak(platform.system())  # e.g. Windows, Linux,
    speak("Operating system architecture type is:")
    speak(platform.architecture())  # e.g. 64-bit
```

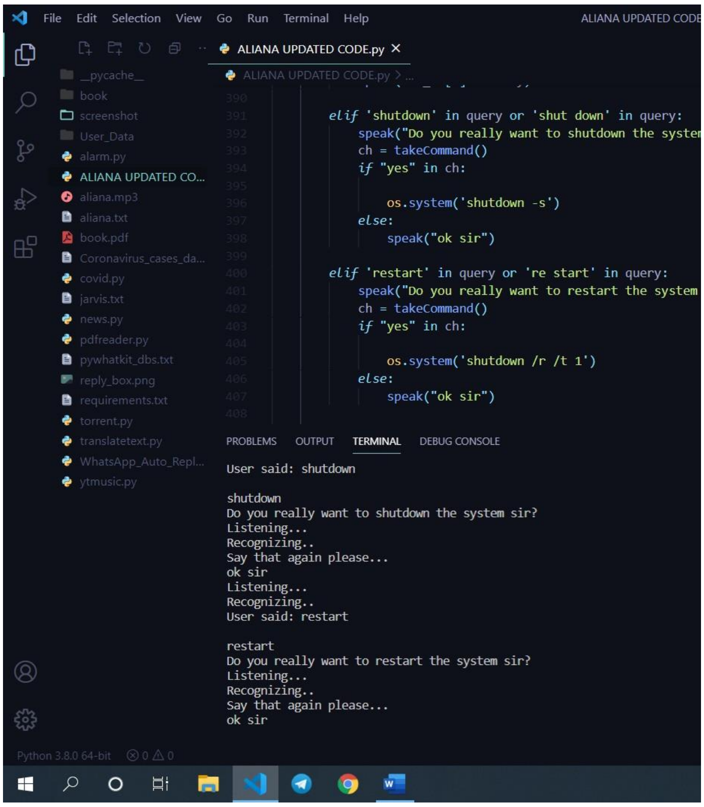
```
systeminfo systeminfo
Listening...
Recognizing..
User said: system info

system info
Operating System type is:
Windows
Operating system architecture type is:
('64bit', 'WindowsPE')
System machine type
AMD64
System name is:
OMKAR
System processsor type is:
Intel64 Family 6 Model 158 Stepping 9, GenuineIntel
Listening...
```



```
        sendEmail(to, content)
        speak("Email has been sent!")
    except Exception as e:
        print(e)
        speak("Sorry omkar sir,I am not able to send this email")

elif 'joke' in query or 'i am bored' in query or "i'm bored" in query:
    speak("Omkar sir when i am here let me fresh up your mind!!")
    while(True):

        speak(pyjokes.get_joke())

elif "activate mod" in query:
    from pywikihow import search_wikihow
    speak("How to do mod is activated please tell me what you want to know")
    how = takeCommand()
    max_results = 1
    how_to = search_wikihow(how, max_results)
    assert len(how_to) == 1
    how to[0].print()
```

```
so, wikipedia says
Shah Rukh Khan (pronounced [ˈʃaːɦrʊx xaːn]; born 2 November 1965), also known by the initialism SRK, is an Indian actor, film producer, and television personality. Referred to i
n the media as the "Baadshah of Bollywood" (in reference to his 1999 film Baadshah), "King of Bollywood" and "King Khan", he has appeared in more than 80 Hindi films, and earned
 numerous accolades, including 14 Filmfare Awards.
Listening...
Recognizing..
Say that again please...
none
Listening...
Recognizing..
User said: tell me a joke

tell me a joke
Omkar sir when i am here let me fresh up your mind!!
Number of days since I have encountered an array index error: -1.
Why don't jokes work in octal? Because 7 10 11.
Real programmers can write assembly code in any language.
There are 10 types of people: those who understand binary, those who don't, and those who were expecting this joke to be in trinary.
```

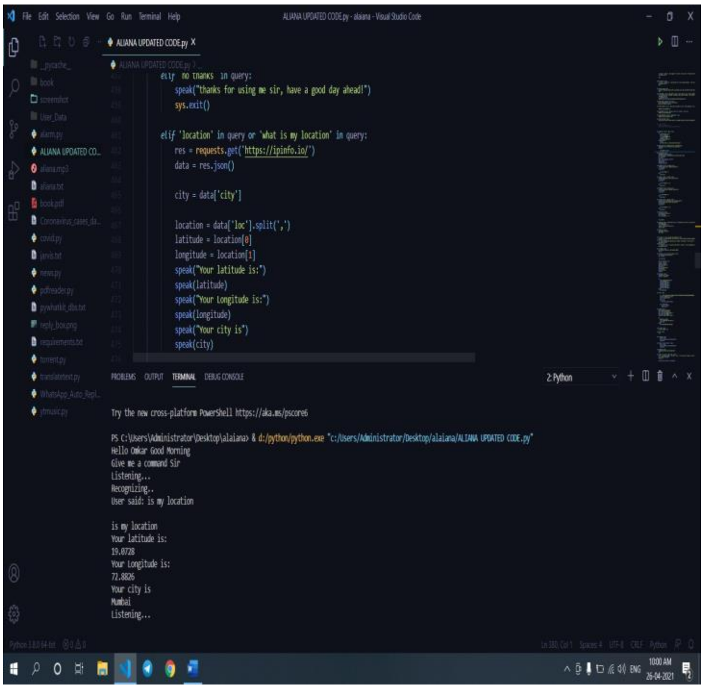
8. Shut down and Restart the System: To do this task we have implemented python OS module.



```python
        elif 'shutdown' in query or 'shut down' in query:
            speak("Do you really want to shutdown the syster
            ch = takeCommand()
            if "yes" in ch:

                os.system('shutdown -s')
            else:
                speak("ok sir")

        elif 'restart' in query or 're start' in query:
            speak("Do you really want to restart the system
            ch = takeCommand()
            if "yes" in ch:

                os.system('shutdown /r /t 1')
            else:
                speak("ok sir")
```

```
User said: shutdown

shutdown
Do you really want to shutdown the system sir?
Listening...
Recognizing..
Say that again please...
ok sir
Listening...
Recognizing..
User said: restart

restart
Do you really want to restart the system sir?
Listening...
Recognizing..
Say that again please...
ok sir
```

9. Write a note/or make a remainder to user and show it to user: To write and show note user we have used python keyboard module.
10. What is my Location: To find user location we have implemented open source API ipinfo.io .

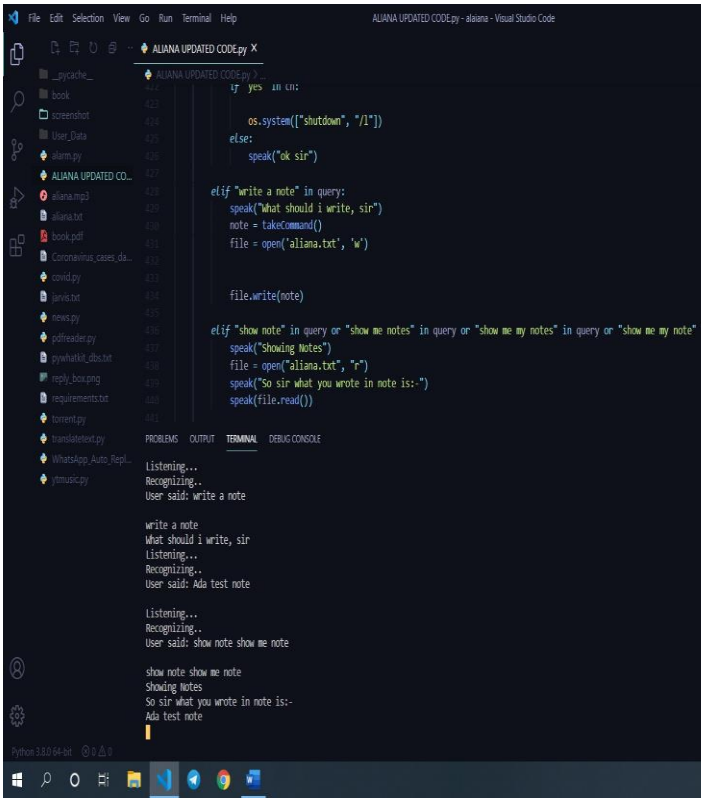


File Edit Selection View Go Run Terminal Help
ALIANA UPDATED CODE.py - alaiana - Visual Studio Code
ALIANA UPDATED CODE.py ×
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
2: Python

```python
        elif 'no thanks' in query:
            speak("thanks for using me sir, have a good day ahead!")
            sys.exit()

        elif 'location' in query or 'what is my location' in query:
            res = requests.get('https://ipinfo.io/')
            data = res.json()

            city = data['city']

            location = data['loc'].split(',')
            latitude = location[0]
            longitude = location[1]
            speak("Your latitude is:")
            speak(latitude)
            speak("Your Longitude is:")
            speak(longitude)
            speak("Your city is")
            speak(city)
```

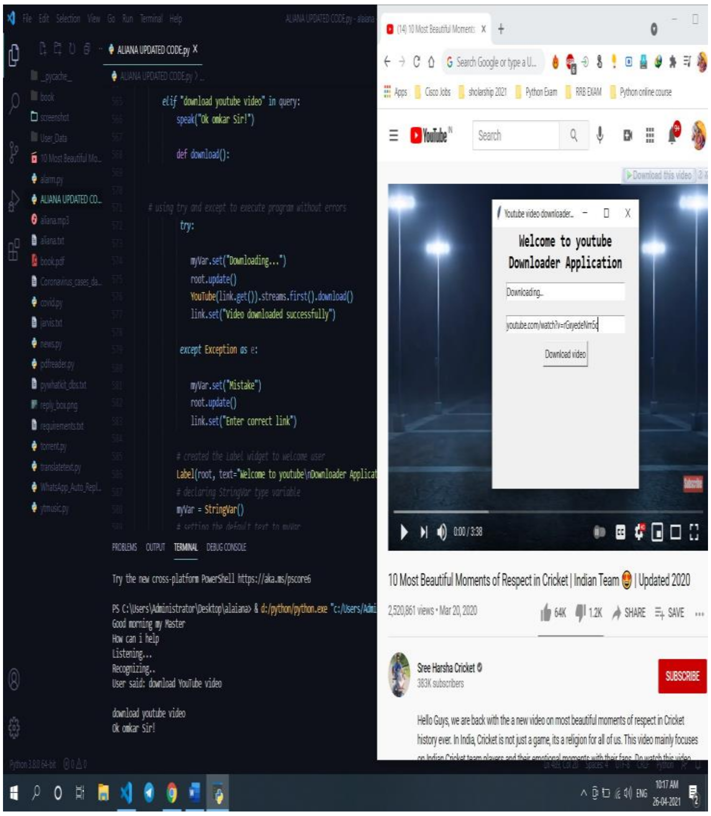
```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Desktop\alaiana> & d:/python/python.exe "c:/Users/Administrator/Desktop/alaiana/ALIANA UPDATED CODE.py"
Hello Omkar Good Morning
Give me a command Sir
Listening...
Recognizing..
User said: is my location

is my location
Your latitude is:
19.0728
Your Longitude is:
72.8826
Your city is
Mumbai
Listening...
```

11. Download Youtube Video: To make Youtube downloader we have used Tkiner module for GUI and pytube module.



```
                if "yes" in cn:

                    os.system(["shutdown", "/l"])
                else:
                    speak("ok sir")

            elif "write a note" in query:
                speak("What should i write, sir")
                note = takeCommand()
                file = open('aliana.txt', 'w')


                file.write(note)

            elif "show note" in query or "show me notes" in query or "show me my notes" in query or "show me my note"
                speak("Showing Notes")
                file = open("aliana.txt", "r")
                speak("So sir what you wrote in note is:-")
                speak(file.read())
```

```
Listening...
Recognizing..
User said: write a note

write a note
What should i write, sir
Listening...
Recognizing..
User said: Ada test note

Listening...
Recognizing..
User said: show note show me note

show note show me note
Showing Notes
So sir what you wrote in note is:-
Ada test note
```

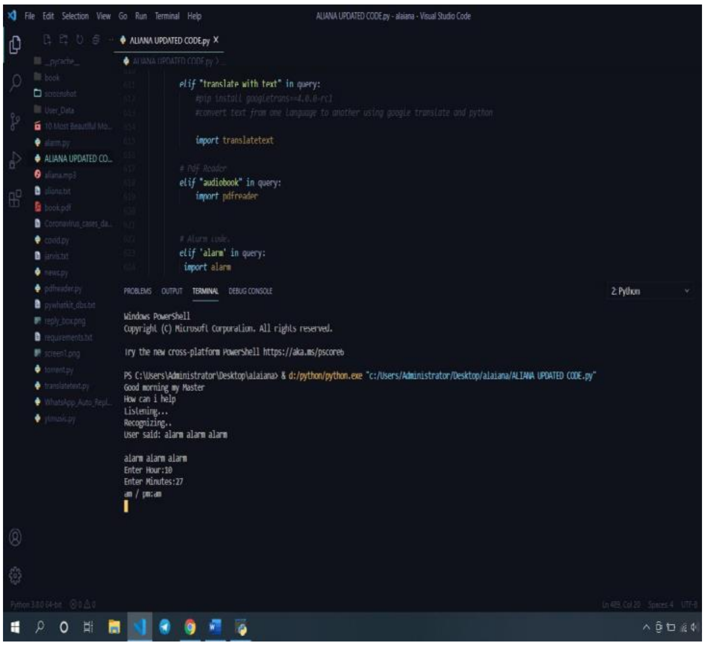


```
            elif "download youtube video" in query:
                speak("Ok omkar Sir!")

                def download():


        # using try and except to execute program without errors
                    try:

                        myVar.set("Downloading...")
                        root.update()
                        YouTube(link.get()).streams.first().download()
                        link.set("Video downloaded successfully")

                    except Exception as e:

                        myVar.set("Mistake")
                        root.update()
                        link.set("Enter correct link")

                # created the label widget to welcome user
                Label(root, text="Welcome to youtube\nDownloader Applicat
                # declaring StringVar type variable
                myVar = StringVar()
                # setting the default text to myVar
```

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Desktop\alaiana> & d:/python/python.exe "c:/Users/Admi
Good morning my Master
How can i help
Listening...
Recognizing..
User said: download YouTube video

download youtube video
Ok omkar Sir!
```

Youtube video downloader...
Welcome to youtube
Downloader Application
Downloading...
youtube.com/watch?v=tGxyedeNm5o
Download video
10 Most Beautiful Moments of Respect in Cricket | Indian Team | Updated 2020
Sree Harsha Cricket
SUBSCRIBE

12. Translate sentence: Here we have used python open source translate module along with google translate API.



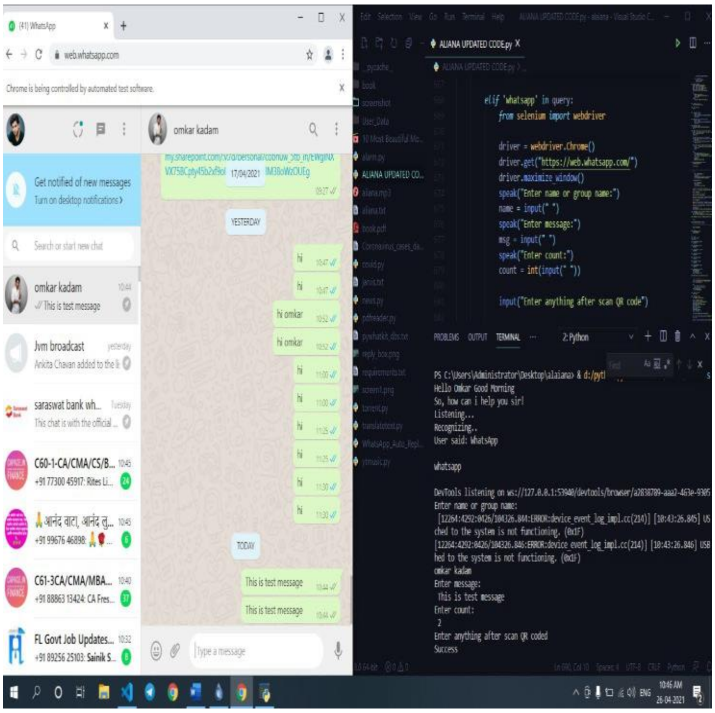
```
            elif "translate with text" in query:
                #pip install googletrans==4.0.0-rc1
                #convert text from one language to another using google translate and python

                import translatetext

            # Pdf Reader
            elif "audiobook" in query:
                import pdfreader


            # Alarm code.
            elif 'alarm' in query:
             import alarm
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator\Desktop\alaiana> & d:/python/python.exe "c:/Users/Administrator/Desktop/alaiana/ALIANA UPDATED CODE.py"
Good morning my Master
How can i help
Listening...
Recognizing..
User said: alarm alarm alarm

alarm alarm alarm
Enter Hour:10
Enter Minutes:27
am / pm:am
```

13. WhatsApp Automation: To do Whatsapp Automation we have used selenium python framework and chrome webdriver. user directly from IDE.

Test Cases	Description	Status
Accepting input from the user	To check whether ALAINA accepts and follows the instructions given by the user and actually, performs all queries by the user	Pass
Idle listening to the user	Idle listening to the user's request. For preventing privacy, we have kept the system based on user's choice.	Pass
Unit and Integration testing of all modules	Testing of Variety of application access and OS related access. Given to ALAINA, and how much she can perform based on the code given. All separate units Perform well, both separately and when integrated	Pass
Functionality Testing	All the functions used work well with each other and give desired output to the user's query.	Pass

VIII. Test Cases and Discussion:



```
omkar kadam 10:44
✓✓ This is test message
Jvm broadcast yesterday
Ankita Chavan added to the li
saraswat bank wh... Tuesday
This chat is with the official ...
C60-1-CA/CMA/CS/B... 10:45
+91 77300 45917: Rites Li... 24
🙏आनंद वाटा, आनंद लु... 10:45
+91 99676 46898: 🙏🌹... 6
C61-3CA/CMA/MBA... 10:40
+91 88863 13424: CA Fres... 37
FL Govt Job Updates... 10:32
+91 89256 25103: Sainik S... 8
```

```
omkar kadam
my.sharepoint.com/:v:/g/personal/cobnuw_5tb_in/EWgINX
VX75BCpty45b2xf9ol 17/04/2021 IM3BoWzOUEg
09:27 ✓
YESTERDAY
hi 10:47 ✓
hi 10:47 ✓
hi omkar 10:52 ✓
hi omkar 10:52 ✓
hi 11:00 ✓
hi 11:00 ✓
hi 11:25 ✓
hi 11:25 ✓
hi 11:30 ✓
hi 11:30 ✓
TODAY
This is test message 10:44 ✓
This is test message 10:44 ✓
```

```python
        elif 'whatsapp' in query:
            from selenium import webdriver

            driver = webdriver.Chrome()
            driver.get("https://web.whatsapp.com/")
            driver.maximize_window()
            speak("Enter name or group name:")
            name = input(" ")
            speak("Enter message:")
            msg = input(" ")
            speak("Enter count:")
            count = int(input(" "))

            input("Enter anything after scan QR code")
```

```
PS C:\Users\Administrator\Desktop\alaiana> & d:/pyt
Hello Omkar Good Morning
So, how can i help you sir!
Listening...
Recognizing..
User said: WhatsApp

whatsapp

DevTools listening on ws://127.0.0.1:53940/devtools/browser/a2838789-aaa2-463e-9305
Enter name or group name:
 [12264:4292:0426/104326.844:ERROR:device_event_log_impl.cc(214)] [10:43:26.845] US
ched to the system is not functioning. (0x1F)
[12264:4292:0426/104326.846:ERROR:device_event_log_impl.cc(214)] [10:43:26.846] USB
hed to the system is not functioning. (0x1F)
omkar kadam
Enter message:
 This is test message
Enter count:
 2
Enter anything after scan QR coded
Success
```

IX. Conclusion:

Through this voice assistant, we have automated various services using a single line command. It eases most of the tasks of the user like searching the web, retrieving weather forecast details, vocabulary help and medical related queries.

We aim to make this project a complete server assistant and make it smart.

Enough to act as a replacement for a general server administration. The future includes integrating ALIANA with mobile using React Native to provide a synchronized experience between the two connected devices.

X. Acknowledgements:

We sincerely express our gratitude to our Professor Miss Odilia Gonsalves and our HOD of IT branch Professor Deepali Maste for time to time guidance in making our project best. We are thankful to them as they time-time arrange virtual conference call to solve our doubts in projects as due to this Covid-19 pandemic is next to impossible to meet face to face and clear our doubts regarding the projects. We also express our gratitude to our Professor Shrikant Kallurkar and entire administrative task for arranging all online classroom effectively and time-to-time video conference and online student development programs for making students soundproof in field of technology.

XI. References:

[1] IEEE Standard Book, J. F. Curtis, (Ed.), Processes and Disorders of Human Communication. New York: Harper and Row, 1978.

[2] Journal Paper, J. Schroeterand M. M. Sondhi, "Techniques for estimating vocal abstract shapes from the speech signal,"IEEE Trans. Speech Audio Process., vol. 2, no. 1, pp. 133–150, 1994.

[3] Proceeding paper, J. M. ardo, "Vocal tract shape analysis for children," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., 1982, pp. 763– 766.