

**A**  
**Project Report**  
**On**  
**VIRTUAL VOICE ASSISTANT USING PYTHON**  
Submitted to  
**RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES,**  
**RK VALLEY**  
in partial fulfilment of the requirements for the award of the Degree of  
**BACHELOR OF TECHNOLOGY**  
**IN**  
**ELECTRONICS AND COMMUNICATION ENGINEERING**

**Submitted by**

**S.RIZWANA BANU**

**R170278**

**M.RAHAMATH BEE**

**R171110**

**S.AYESHA**

**R170011**

**Under the Guidance of**

**Mr.T.Naresh,M.tech**

**Assistant Professor**

**Department Of Electronics and Communication Engineering**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES**

**(RGUKT KADAPA is approved by UGC, AICTE, established in 2008, provide Education opportunities for rural people), Vempalli, Kadapa-516330**

**2019-2023**

**RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES**  
**(RGUKT KADAPA is approved by UGC, AICTE, established in 2008, provide Education opportunities for rural people)**  
**Vempalli, Kadapa-516330**  
**2019-2023**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**



**CERTIFICATE**

This is to certify that the project report entitled **“VIRTUAL VOICE ASSISTANT USING PYTHON”** a bonafide record of the project work done and submitted by

**S.RIZWANA BANU**

**R170278**

**M.RAHAMATH BEE**

**R171110**

**S.AYESHA**

**R170011**

for the partial fulfilment of the requirements for the award of B.Tech Degree in **ELECTRONICS AND COMMUNICATION ENGINEERING** RGUKT , Kadapa.

**GUIDE**

Mr.T.Naresh  
Assistant Professor

**Head of the Department**

Mr.B.Madhan Mohan  
Assistant Professor,

External Viva-Voice Exam Held on \_\_\_\_\_

**INTERNAL EXAMINER**

**EXTERNAL EXAMINER**

## **DECLARATION**

We hereby declare that the project report entitled **“VIRTUAL VOICE ASSISTANT USING PYTHON”** submitted to the Department of **ELECTRONICS AND COMMUNICATION ENGINEERING** in partial fulfillment of requirements for the award of the degree of **BACHELOR OF TECHNOLOGY**. This project is the result of our own effort and that it has not been submitted to any other University or Institution for the award of any degree or diploma other than specified above.

S.RIZWANA BANU	R170278
S.AYESHA	R170011
M.RAHAMATH BEE	R171110

## ACKNOWLEDGEMENTS

We are thankful to our guide **Mr. T.Naresh** for his valuable guidance and encouragement. His helping attitude and suggestions have helped us in the successful completion of the project.

We would like to express our gratefulness and sincere thanks to **Mr. M. Madhan Mohan** ,Assistant Professor of the Department of **ELECTRONICS AND COMMUNICATION ENGINEERING**, for his kind help and encouragement during the course of our study and in the successful completion of the project work.

We have great pleasure in expressing our hearty thanks to our beloved Director **Dr. SANDHYA RANI** for spending his valuable time with us to complete this project.

Successful completion of any project cannot be done without proper support and encouragement. We sincerely thanks to the Management for providing all the necessary facilities during the course of study.

We would like to thank our parents and friends, who have the greatest contributions in all our achievements, for the great care and blessings in making us successful in all our endeavors.

S. RIZWANA BANU	R170278
S. AYESHA	R170011
M.RAHAMATH BEE	R171110

## **ABSTRACT**

In this modern era, day to day life became smarter and interlinked with technology. We already know some voice assistance like google, Siri. Etc. Now in our voice assistance system, it can act as a basic medical prescriber, daily schedule reminder, note writer, calculator and a search tool.

This project works on voice input and give output through voice and displays the text on the screen. The main agenda of our voice assistance makes people smart and give instant and computed results. The voice assistance takes the voice input through our microphone (Bluetooth and wired microphone) and it converts our voice into computer understandable language gives the required solutions and answers which are asked by the user. This assistance connects with the world wide web to provide results that the user has questioned. Natural Language Processing algorithm helps computer machines to engage in communication using natural human language in many forms.

# TABLE OF CONTENT

Abstract	i
Table of Contents	ii
List of Figures	iii

Chapter No.	Description	Page No
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 Objective	1
<b>2</b>	<b>VIRTUAL VOICE ASSISTANT</b>	<b>2</b>
	2.1 Purpose	2
	2.2 Scope	2
	2.3 Applicability	2
<b>3</b>	<b>SYSTEM OF TECHNOLOGY</b>	<b>3</b>
	3.1 Python	3
<b>4</b>	<b>REQUIREMENT AND ANALYSIS</b>	<b>4-5</b>
	4.1 Package Required	4-5
	4.2 Problem Definiton	5
<b>5</b>	<b>SERVICES OF VIRTUAL ASSISTANTS</b>	<b>6</b>
<b>6</b>	<b>BLOCK DIAGRAM</b>	<b>7</b>
<b>7</b>	<b>HARDWARE AND SOFTWARE REQUIREMENTS</b>	<b>8</b>
	7.1 Hardware	8
	7.2 Software	8
<b>8</b>	<b>SOFTWARE CODE</b>	<b>9-16</b>
	8.1 Implemetation	9
	8.1.1 Setting up the speech engine :	9
	8.1.2 Run and Wait:	10
	8.1.3 Initiate a Function to Greet the User	10-11
	8.1.4 Main Function	11-12
	8.2 SKILL 1 -Fetching Data from Wikipedia	12
	8.3 SKILL 2 -Accessing the web Browsers	12-13
	8.4 SKILL 3- To Forecast The Weather	13-14
	8.5 SKILL 4 -Predicting time	14
	8.6 SKILL 5- Extra Features	14-15
	8.7 SKILL 6 -To fetch the News	15
	8.8 SKILL 7-Capturing the Photo	15
	8.9 SKILL 8-Searching from Web	15
	8.10 SKILL 9- Setting yout AI to answer the computational and geographical questions	16
	8.11 SKILL 10- TO LOG OFF YOUR PC	16
<b>9</b>	<b>CONCLUSION</b>	<b>17</b>
References		18

## LIST OF FIGURES

FIGURE NO.	DESCRIPTION	PAGE NO.
<b>1</b>	Flow chart of Virtual Voice Assistant	<b>7</b>

# Chapter 1

## INTRODUCTION

In today's era almost all tasks are digitalized. We have Smartphone in hands and it is nothing less than having world at your fingertips. These days systems where we can say Text Dad, "I'll be late today." And the text is sent. That is the task of a Voice Assistant. It also supports specialized task such as booking a flight, or finding cheapest book online from various ecommerce sites and then providing an interface to book an order are helping automate search, discovery and online order operations.

Voice Assistants are software programs that help you ease your day-to-day tasks, such as showing weather report, creating reminders, making shopping lists etc. They can take commands via text (online chat bots) or by voice. Voice based intelligent assistants need an invoking word or wake word to activate the listener, followed by the command. We have so many virtual assistants, such as Apple's Siri, Amazon's Alexa and Microsoft's Cortana.

This system is designed to be used efficiently on desktops. Personal assistant software improves user productivity by managing routine tasks of the user and by providing information from online sources to the user. It is effortless to use.

Voice searches have dominated over text search. Web searches conducted via mobile devices have only just overtaken those carried out using a computer and the analysts are already predicting that 50% of searches will be via voice by 2020. voice assistants are turning out to be smarter than ever. Allow your intelligent assistant to make email work for you. Detect intent, pick out important information, automate processes, and deliver personalized responses.

This project was started on the premise that there is sufficient amount of openly available 1 data and information on the web that can be utilized to build a virtual assistant that has access to making intelligent decisions for routine user activities.

### 1.1 OBJECTIVE

Main objective of building personal assistant software (a virtual assistant) is using semantic data sources available on the web, user generated content and providing knowledge from knowledge databases. The main purpose of an intelligent virtual assistant is to answer questions that users may have. This may be done in a business environment, for example, on the business website, with a chat interface. Virtual assistants can tremendously save you time. We spend hours in online research and then making the report in our terms of understanding.

One of the main advantages of voice searches is their rapidity. In fact, voice is reputed to be four times faster than a written search: whereas we can write about 40 words per minute, we are capable of speaking around 150 during the same period of time<sup>15</sup>. In this respect, the ability of personal assistants to accurately recognize spoken words are a prerequisite for them to be adopted by consumers.

## **Chapter 2**

# **VIRTUAL VOICE ASSISTANT**

### **2.1 PURPOSE**

Purpose of virtual assistant is to being capable of voice interaction, music playback, making to-do lists, setting alarms, streaming podcasts, playing audiobooks, and providing weather, traffic, sports, and other real-time information, such as news. Virtual assistant sense able users to speak natural language voice commands in order to operate the device and its apps. There is an increased overall awareness and a higher level of comfort demonstrated specifically by millennial consumers. In this ever-evolving digital world where speed, efficiency, and convenience are constantly being optimized, it's clear that we are moving towards less screen interaction.

### **2.2 SCOPE**

Voice assistants will continue to offer more individualized experiences as they get better at differentiating between voices. However, it's not just developers that need to address the complexity of developing for voice as brands also need to understand the capabilities of each device and integration and if it makes sense for them specific brand. They will also need to focus on maintaining a user experience that is consistent within the coming years as complexity becomes more of a concern. This is because the visual interface with voice assistants is missing. Users simply cannot see or touch a voice interface.

### **2.3 APPLICABILITY**

The mass adoption of artificial intelligence in users' everyday lives is also fuelling the shift towards voice. The number of IoT devices such as smart thermostats and speakers are giving voice assistants more utility in a connected user's life. Smart speakers are the number one way we are seeing voice being used. Many industry experts even predict that nearly every application will integrate voice technology in some way in the next 5 years. The use of virtual assistants can also enhance the system of IoT (Internet of Things). Twenty years from now, Microsoft and its competitors will be offering personal digital assistants that will offer the services of a full-time employee usually reserved for the rich and famous.



## **Chapter 3**

### **SYSTEM OF TECHNOLOGY**

#### **3.1 PYTHON**

Python is an OOPs (Object Oriented Programming) based, high level, interpreted programming language. It is a robust, highly useful language focused on rapid application development (RAD). Python helps in easy writing and execution of codes. Python can implement the same logic with as much as 1/5 code as compared to other OOPs languages.

Python provides a huge list of benefits to all. The usage of Python is such that it cannot be limited to only one activity. Its growing popularity has allowed it to enter into some of the most popular and complex processes like Artificial Intelligence (AI), Machine Learning (ML), natural language processing, data science etc.

Python has a lot of libraries for every need of this project. For this project we used libraries used are speech recognition to recognize voice, Pytsx for text to speech.

Python is reasonably efficient. Efficiency is usually not a problem for small examples. If your Python code is not efficient enough, a general procedure to improve it is to find out what is taking most the time, and implement just that part more efficiently in some lower-level language. This will result in much less programming and more efficient code (because you will have more time to optimize) than writing everything in a low-level language.

## Chapter 4

### REQUIREMENT AND ANALYSIS

#### 4.1 PACKAGES REQUIRED

To build a personal voice assistant it's necessary to install the following packages in your system using the pip command.

System Analysis is about complete understanding of existing systems and finding where the existing system fails. The solution is determined to resolve issues in the proposed system. It defines the system. The system is divided into smaller parts. Their functions and inter relation of these modules are studied in system analysis. The complete analysis is followed below.

**a. PYTTSX**

Pytttsx stands for Python Text to Speech. It is a cross-platform Python wrapper for text-to-speech synthesis. It is a Python package supporting common text-to-speech engines on MacOS X, Windows, and Linux. It works for both Python2.x and 3.x versions. Its main advantages that it works offline.

**b. SPEECH RECOGNITION**

This is a library for performing speech recognition, with support for several engines and APIs, online and offline. It supports APIs like Google Cloud Speech API, IBM Speech to Text, Microsoft Bing Voice Recognition etc.

**c. WIKIPEDIA**

Wikipedia is a multilingual online encyclopedia used by many people from academic community ranging from freshmen to students to professors who want to gain information over a particular topic. This package in python extracts data's required from Wikipedia.

**d. ECAPTURE**

This module is used to capture images from your camera

**e. DATETIME**

This is an inbuilt module in python and it works on date and time

**f. OS**

This module is a standard library in python and it provides the function to interact with operating system

**g. TIME**

The time module helps us to display time

**h. WEB BROWSER**

The time module helps us to display time

#### **i. SUBPROCESS**

This is a standard library use to process various system commands like to log off or to restart your PC.

#### **j. JSON**

The json module is used for storing and exchanging data.

#### **k. REQUEST**

The request module is used to send all types of HTTP request. Its accepts URL as parameters and gives access to the given URL'S.

#### **WOLFRAM ALPHA**

Wolfram Alpha is an API which can compute expert-level answers using Wolfram's algorithms, knowledge base and AI technology. It is made possible by the Wolfram Language.

### **4.2 PROBLEM DEFINITION :**

Usually, user needs to manually manage multiple sets of applications to complete one task. For example, a user trying to make a travel plan needs to check for airport codes for nearby airports and then check travel sites for tickets between combinations of airports to reach the destination. There is need of a system that can manage tasks effortlessly. we already have multiple virtual assistants. But we hardly use it. There are number of people who have issues in voice recognition. These systems can understand English phrases but they fail to recognize in our accent.

Our way of pronunciation is way distinct from theirs also they are easy to use on mobile devices than desktop systems. There is need of a virtual assistant that can understand English in Indian accent and work on desktop system. when a virtual assistant is not able to answer questions accurately, it's because it lacks the proper context or doesn't understand the intent of the question. Its ability to answer questions relevantly only happens with rigorous optimization, involving both humans and machine learning. Continuously ensuring solid quality control strategies will also help manage the risk of the virtual assistant learning undesired bad behaviours. They require large amount of information to be fed in order for it to work efficiently. virtual assistant should be able to model complex task dependencies and use these models to recommend optimized plans for the user. It needs to be tested for finding optimum paths when a task has multiple sub-tasks and each sub-task can have its own sub-tasks.

## **Chapter 5**

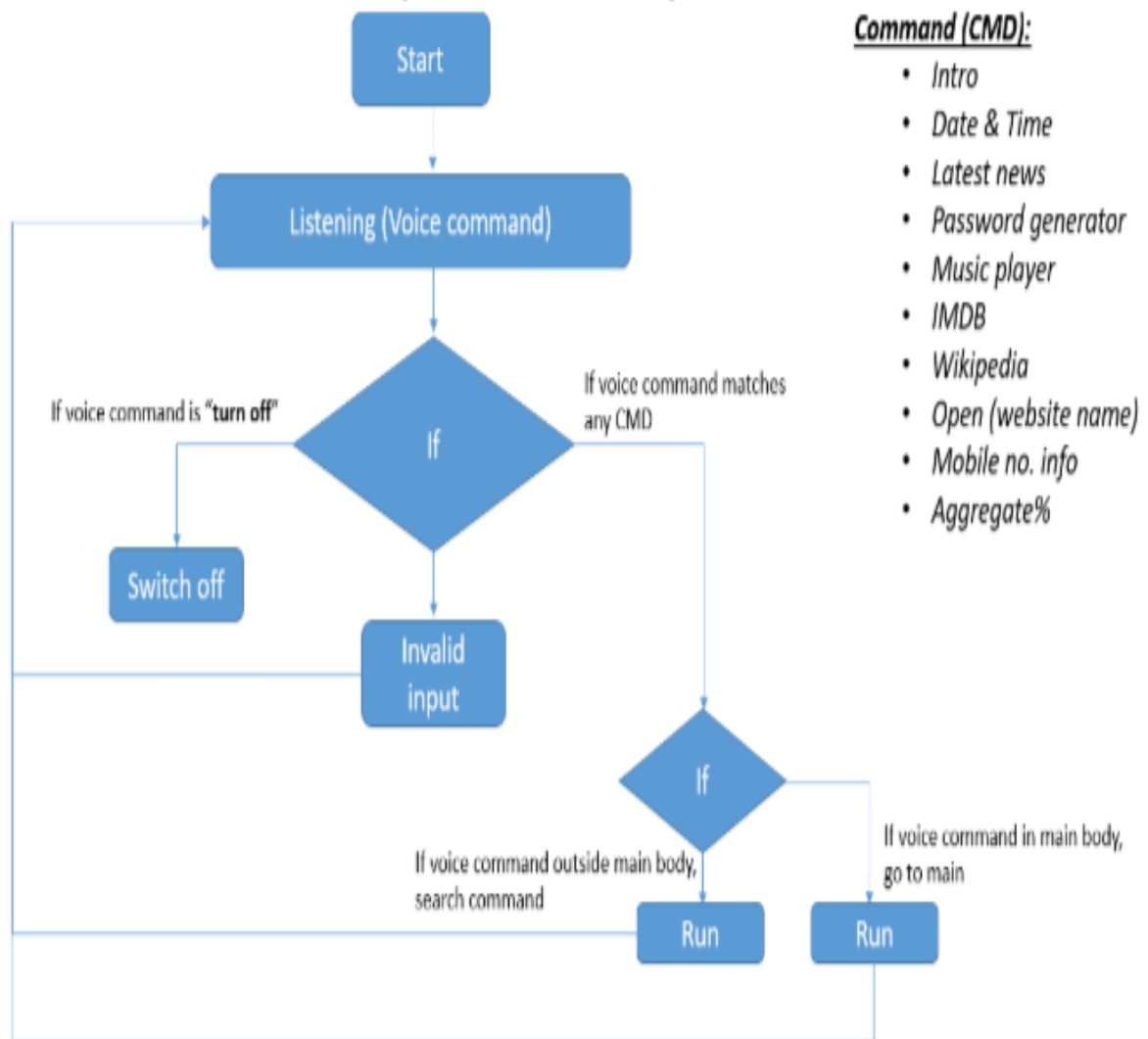
### **SERVICES OF VIRTUAL ASSISTANTS**

The following are the services provided by Virtual Assistants:

- Providing information such as weather, facts from e.g., Wikipedia etc.
  - Set an alarm or make to-do lists and shopping lists. \*Remind you of birthdays and meetings.
  - Play music from streaming services such as Saavn and Gaana.
  - Play videos, TV shows or movies on televisions, streaming from e.g., Netflix or Hotstar.
- Book tickets for shows, travel and movies

## Chapter 6

### BLOCK DIAGRAM



**Fig 1: Flow chart of Virtual Voice Assistant**

## **Chapter 7**

### **HARDWARE AND SOFTWARE REQUIREMENTS**

The software is designed to be light-weighted so that it doesn't be a burden on the machine running it. This system is being build keeping in mind the generally available hardware and software compatibility.

Here are the minimum hardware and software requirement for virtual assistant.

#### **7.1 HARDWARE:**

- \*Intel processor or Anyother.

- \*RAM 512MB or more.

#### **7.2 SOFTWARE:**

- \*Windows 7(32-bit) or above.

- \*Python 3.9 or later

- \*Chrome Driver

## Chapter 8

### SOFTWARE CODE

#### 8.1 IMPLEMENTATION

Import the following libraries

```
import speech_recognition as sr
import pyttsx3
import datetime
import wikipedia
import webbrowser
import os
import time
import subprocess
from ecapture import ecapture as ec
import wolframalpha
import json
import requests
```

```
print('Loading AI personal assistant - Go One')
```

##### 8.1.1 Setting up the speech engine :

The pyttsx3 module is stored in a variable name engine.

Sapi5 is a Microsoft Text to speech engine used for voice recognition.

The voice Id can be set as either 0 or 1,

0 indicates Male voice

1 indicates Female voice

```
engine=pyttsx3.init('sapi5')
```

```
voices=engine.getProperty('voices')
```

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

Now define a function **speak** which converts text to speech. The speak function takes the text as its argument,further initialize the engine.

### 8.1.2 Run and Wait:

This function Blocks while processing all currently queued commands. It Invokes callbacks for engine notifications appropriately and returns back when all commands queued before this call are emptied from the queue.

```
def speak(text):  
    engine.say(text)  
    engine.runAndWait()
```

### 8.1.3 Initiate a Function to Greet the User :

Define a function wishMe for the AI assistant to greet the user.

The now().hour function abstract's the hour from the current time.

If the hour is greater than zero and less than 12, the voice assistant wishes you with the message "Good Morning".

If the hour is greater than 12 and less than 18, the voice assistant wishes you with the following message "Good Afternoon".

Else it voices out the message "Good evening"

```
def wishMe():  
    hour=datetime.datetime.now().hour  
  
    if hour>=0 and hour<12:  
        speak("Hello,Good Morning")  
        print("Hello,Good Morning")  
  
    elif hour>=12 and hour<18:  
        speak("Hello,Good Afternoon")  
        print("Hello,Good Afternoon")  
  
    else:  
        speak("Hello,Good Evening")  
        print("Hello,Good Evening")
```

Define a function **wishMe** for the AI assistant to greet the user.

The **now().hour** function abstract's the hour from the current time.



If the hour is greater than zero and less than 12, the voice assistant wishes you with the message “Good Morning”.

If the hour is greater than 12 and less than 18, the voice assistant wishes you with the following message “Good Afternoon”.

Else it voices out the message “Good evening”

```
def takeCommand():  
    r=sr.Recognizer()  
    with sr.Microphone() as source:  
        print("Listening...")  
        audio=r.listen(source)  
  
    try:  
        statement=r.recognize_google(audio,language='en-in')  
        print(f"user said: {statement}\n")  
  
    except Exception as e:  
        speak(" I an not understand , please say that again")  
        return "None"  
    return statement  
  
speak("Loading your AI personal assistant Go-One")  
wishMe()
```

#### 8.1.4 Main Function:

The main function starts from here,the commands given by the humans is stored in the variable **statement**.

```
if __name__=='__main__':  
  
    while True:  
        speak("Tell me how can I help you now?")  
        statement = takeCommand().lower()  
    if statement==0:  
        continue
```

If the following trigger words are there in the statement given by the users it invokes the virtual assistant to speak the below following commands.

if "good bye" in statement or "ok bye" in statement or "stop" in statement:

```
    speak('your personal assistant G-one is shutting down, bye ,Have a nice day')
    print('your personal assistant G-one is shutting down,Good bye,Have a nice day')
    break
```

## 8.2 SKILL 1 -FETCHING DATA FROM WIKIPEDIA :

The following commands helps to extract information from wikipedia. The **wikipedia.summary()** function takes two arguments, the statement given by the user and how many sentences from wikipedia is needed to be extracted is stored in a variable **result**.

if 'wikipedia' in statement:

```
    speak('Searching Wikipedia...')
    statement =statement.replace("wikipedia", "")
    results = wikipedia.summary(statement, sentences=3)
    speak("According to Wikipedia")
    print(results)
    speak(results)
```

## 8.3 SKILL 2 -ACCESSING THE WED BROWSERS — GOOGLE CHROME , G-MAIL AND YOUTUBE:

The web browser extracts data from web. The **open\_new\_tab** function accepts **URL** as a parameter that needs to be accessed.

The **Python time sleep function** is used to add delay in the execution of a program. We can use this function to halt the execution of the program for given **time** in seconds.

elif 'open youtube' in statement:

```
    webbrowser.open_new_tab("https://www.youtube.com")
    speak("youtube is open now")
    time.sleep(5)
```

elif 'open google' in statement:

```
    webbrowser.open_new_tab("https://www.google.com")
    speak("Google chrome is open now")
    time.sleep(5)
```

elif 'open gmail' in statement:

```
webbrowser.open_new_tab("gmail.com")
speak("Google Mail open now")
time.sleep(5)
```

#### 8.4 SKILL 3- TO FORECAST THE WEATHER:

Now to program your AI assistant to detect weather we need to generate an API key from Open Weather map.

Open weather map is an online service which provides weather data. By generating an API ID in the official website you can use the APP\_ID to make your voice assistant detect weather of all places whenever required. The necessary modules needed to be imported for this weather detection is json and request module.

The **city\_name variable** takes the command given by the human using the **takeCommand()** function. The **get** method of **request** module returns a **response** object. And the **json** methods of response object converts json format data into python format.

The variable **X** contains list of nested dictionaries which checks whether the value of 'COD' is 404 or not that is if the city is found or not.

The values such as temperature and humidity is stored in the main key of variable **Y**.

elif "weather" in statement:

```
api_key="8ef61edcf1c576d65d836254e11ea420"
base_url="https://api.openweathermap.org/data/2.5/weather?"
speak("whats the city name")
city_name=takeCommand()
complete_url=base_url+"appid="+api_key+"&q="+city_name
response = requests.get(complete_url)
x=response.json()
if x["cod"]!="404":
    y=x["main"]
    current_temperature = y["temp"]
    current_humidiy = y["humidity"]
    z = x["weather"]
    weather_description = z[0]["description"]
    speak(" Temperature in kelvin unit is " +
          str(current_temperature) +
          "\n humidity in percentage is " +
          str(current_humidiy) +
          "\n description  " +
```

```

        str(weather_description))
print(" Temperature in kelvin unit = " +
      str(current_temperature) +
      "\n humidity (in percentage) = " +
      str(current_humidity) +
      "\n description = " +
      str(weather_description))
else:
    speak(" City Not Found ")

```

Human: Hey G-One ,I want to get the weather data

G-One: What is the city name?

Human: Himachal Pradesh

G-One: Temperature in kelvin unit is 301.09 , Humidity in percentage is 52 and Description is light rain.

## 8.5 Skill 4 -PREDICTING TIME:

The current time is abstracted from **datetime.now()** function which displays the hour, minute and second and is stored in a variable name **strTime**.

elif 'time' in statement:

```

strTime=datetime.datetime.now().strftime("%H:%M:%S")
speak(f"the time is {strTime}")

```

## 8.6 SKILL 5- EXTRA FEATURES:

It would be interesting to program your AI assistant to answer the following questions like what it can and who created it, isn't it?

elif 'who are you' in statement or 'what can you do' in statement:

```

speak('I am G-one version 1 point O your personal assistant. I am programmed to minor tasks like'
      'opening youtube,google chrome,gmail and stackoverflow ,predict time,take a photo,search
wikipedia,predict weather'

```

```

      'in different cities , get top headline news from times of india and you can ask me computational
or geographical questions too!')

```

elif "who made you" in statement or "who created you" in statement or "who discovered you" in statement:

```

speak("I was built by ece team")
print("I was built by ece team")
elif "open stackoverflow" in statement:

```

```

webbrowser.open_new_tab("https://stackoverflow.com/login")

```

```
speak("Here is stackoverflow")
```

## 8.7 SKILL 6 -TO FETCH LATEST NEWS:

If the user wants to know the latest news , The voice assistant is programmed to fetch top headline news from Time of India by using the web browser function.

## 8.8 SKILL 7-CAPTURING PHOTO:

The **ec.capture()** function is used to capture images from your camera. It accepts 3 parameter.

**Camera index** — The first connected webcam will be indicated as index 0 and the next webcam will be indicated as index 1

**Window name** — It can be a variable or a string. If you don't wish to see the window, type as False

**Save name** — A name can be given to the image and if you don't want to save the image, type as false

elif 'news' in statement:

```
news = webbrowser.open_new_tab("https://timesofindia.indiatimes.com/home/headlines")
```

```
speak('Here are some headlines from the Times of India,Happy reading')
```

```
time.sleep(6)
```

elif "camera" in statement or "take a photo" in statement:

```
ec.capture(0,"robo camera","img.jpg")
```

## 8.9 SKILL 8-SEARCHING DATA FROM WEB :

From the **web browser** you can **search** required data by passing the user statement (command) to the **open\_new\_tab()** function.

*User: Hey G-One, please search images of butterfly*

*The Voice assistant opens the google window & fetches butterfly images from web.*

elif 'search' in statement:

```
statement = statement.replace("search", "")
```

```
webbrowser.open_new_tab(statement)
```

```
time.sleep(5)
```

## 8.10 SKILL 9- SETTING YOUR AI ASSISTENT TO ANSWER GEOGRAPHICAL AND COMPUTATIONAL QUESTIONS:

Here we can use a third party API called **Wolfram alpha API** to answer computational and geographical questions. It is made possible by the Wolfram Language. The **client** is an instance (class) created for wolfram alpha. The **res** variable stores the response given by the wolfram alpha

elif 'ask' in statement:

    speak('I can answer to computational and geographical questions and what question do you want to ask now')

    question=takeCommand()

    app\_id="R2K75H-7ELALHR35X"

    client = wolframalpha.Client('R2K75H-7ELALHR35X')

    res = client.query(question)

    answer = next(res.results).text

    speak(answer)

    print(answer)

To access the wolfram alpha API an unique App ID is required which can be generated by the following ways:

1. Login to the official page of wolfram alpha and create an account if you do not possess one.
2. Sign in using your wolfram ID
3. Now you will view the homepage of the website. Head to the account section in the top right corner where you see your email. In the drop down menu, select the My Apps (API) option.
4. You will see this following window, now click Get APP\_ID button
5. Now you will get the following dialog box, give a suitable name and description and click the App ID button, an App ID will be generated and this is an unique ID. Using the App Id use can access the Wolfram alpha API.

Human: Hey G-One ,what is the capital of California?

G-One Voice assistant: Sacramento, United States of America

## 8.11 SKILL 10- TO LOG OFF YOUR PC:

The **subprocess.call()** function here is used to process the system function to log off or to turn off your PC. This invokes your AI assistant to automatically turn off your PC.

elif "log off" in statement or "sign out" in statement:

    speak("Ok , your pc will log off in 10 sec make sure you exit from all applications")

    subprocess.call(["shutdown", "/l"])

time.sleep(3)

## **Chapter 9**

### **CONCLUSION**

In this paper “Virtual Assistant Using Python” we discussed the design and implementation of Digital Assistance. The project is built using open source software modules with PyCharm community backing which can accommodate any updates shortly. The modular nature of this project makes it more flexible and easy to add additional features without disturbing current system functionalities.

It not only works on human commands but also give responses to the user based on the query being asked or the words spoken by the user such as opening tasks and operations. It is greeting the user the way the user feels more comfortable and feels free to interact with the voice assistant. The application should also eliminate any kind of unnecessary manual work required in the user life of performing every task. The entire system works on the verbal input rather than the next one.

**REFERENCE :**

1. <https://towardsdatascience.com/how-to-build-your-own-ai-personal-assistant-using-python-f57247b4494b>
2. <https://www.geeksforgeeks.org/voice-assistant-using-python>
3. Zwass V. Speech Recognition [Internet]. Encyclopedia Britannica Online: 2016; [cited 2019 April 7]
4. Markowitz J. Toys That Have a Voice. Speech Technology Magazine [Internet]. 2003 March [cited 2019 April 7]
5. Pinola M. Speech Recognition Through the Decades: How We Ended Up with Siri. PC World [Internet]. 2011 November 2 [Cited 2019 April 7].
6. Swapnil Saurav, Python Programming-learn and practice (2nd Edition). Ingram short title; 2018 January
7. Grant Smith, Everything You Need To Ace Computer Science in One Big Fat Notebook, Workman Publishing Co. Inc., 2020 March