**MINI PROJECT REPORT**

**On**

**VOICE ASSISTANT**

**(PYTHON)**

****

**GLA University**

**Mathura- 281406, INDIA**

**2019**

**Submitted To**

Asst. Prof. Sharad Gupta

Department of Computer Engineering & Applications

Institute of Engineering & Technology

**Submitted By**

Swapnil Saxena (171500353)

Priyanka Gupta (171500241)

Palak Agarwal (171500213)

**Student Information**

|  |  |
| --- | --- |
| Name:  Swapnil Saxena,  Palak Agarwal,  Priyanka Gupta | University Roll. No.-  171500353,  171500213,  171500241 |
| Mobile: 8299201750 | Email:  [swapnil.saxena\_cs17@gla.ac.in](mailto:swapnil.saxena_cs17@gla.ac.inpalak.agarwal_cs17@gla.ac.in priyanka.gupta_cs17@gla.ac.in)  [palak.agarwal\_cs17@gla.ac.in priyanka.gupta\_cs17@gla.ac.in](mailto:swapnil.saxena_cs17@gla.ac.inpalak.agarwal_cs17@gla.ac.in priyanka.gupta_cs17@gla.ac.in) |

**Information about Industry/Organization:**

|  |  |
| --- | --- |
| Organization | GLA university |
| Contact Person | Asst. Prof. Sharad Gupta  8505839515 |

**Project Information:**

|  |  |
| --- | --- |
| Title Of Project | Voice Assistant |
| Role & Responsibility |  |
| Technical Details | |  |  | | --- | --- | | Hardware Requirements: |  | | Main Processor | Core I3 | | Hard-disk Capacity | 1 G.B | | RAM | 2 GB | | Clock Speed | 2.8 Hz | | Keyboard | 104 Key | | Software Requirements: |  | | Operating System | Windows 10 | | Language | Python 3.7 | |
| Project Implementation Details | Successfully completed |

**Summary of the Project Work**

|  |
| --- |
| In this project we learned to work on virtual assistant to query voice commands of a program user, so to do this we learnt the interrelation using python programming. During this Training we developed a project on Voice Assistant for a general store. And have learnt many uses of core java in our daily life for making our life simpler and easily managing our data.  The project entitled Voice Assistant was completed successfully. The system has been developed with much care and free of errors and at the same time it is efficient and less time consuming. The entire code is error free. Also the project helped us in understanding about the development phases of a project and software development life cycle. We also learned how to test different features of a project. There is a scope for further development in our project to a great extent. A number of features can be added to this project in future. |

ACKNOWLEDGEMENT

It is our immense pleasure to acknowledgement the assistance of number of people without whose help this would not have been possible. Firstly, we would like to express our respect and gratitude towards **Mr. Sharad Gupta**, our guide for giving us this opportunity and providing us with full support and invaluable guidance in every way possible, throughout the project.

We would also like to thank the group members for their constant support extended for the completion of this project. It would not have been possible without the efforts and co-operation of our group members. This project has provided us technical as well as general knowledge and given maturity of thought and vision. Lastly, I would like to thank our friends and respondents for their support and willingness to help is through the project.

Swapnil Saxena (171500353)

Palak Agarwal (171500213)

Priyanka Gupta (171500241)

Department of Computer Engineering and Applications

GLA University, Mathura

17 km Stone NH#2, Mathura-Delhi Road, P.O. Chaumuhan

Mathura, 281406

DECLARATION

I hereby declare that the project work entitled “Voice Assitant**”** submitted to the GLA University Mathura, is a record of an original work done by me under the guidance of Mr. Sharad Gupta.

Signature of Candidate:

Name of team members:

Swapnil Saxena,

Palak Agarwal,

Priyanka Gupta

Roll No:

171500353,

171500213,

171500241

Course: Computer Science and Engineering

Year: III

Semester: VI

ABSTRACT

To call any technology that makes our lives easier by one name is almost impossible. There are a variety of terms that refer to agents that can perform tasks or services for an individual, and they are almost interchangeable — but not quite. They differ mainly based on how we interact with the technology, the app, or a combination of both.

**Here are some basic definitions, similarities, and differences:**

**Intelligent Personal Assistant**: This is software that can assist people with basic tasks, usually using natural language. Intelligent personal assistants can go online and search for an answer to a user’s question. Either text or voice can trigger an action.

**Automated Personal Assistant**: This term is synonymous with intelligent personal assistant.

**Smart Assistant**: This term usually refers to the types of physical items that can provide various services by using smart speakers that listen for a wake word to become active and perform certain tasks. Amazon’s Echo, Google’s Home, and Apple’s Home Pod are types of smart assistants.

**Virtual Digital Assistants**: These are automated software applications or platforms that assist the user by understanding natural language in either written or spoken form. Technology is constantly advancing and changing, and the voice assistant market will progress along with it. In April 2015, the research firm Gartner predicted that by the end of 2018, 30 percent of interactions with technology would be through “conversations” with smart machines, many of them by voice. Their reports say unique consumer users for virtual digital assistants (which they define as automated software applications or platforms that assist the human user through understanding natural language in written or spoken form) will grow from more than 390 million worldwide users in 2015 to 1.8 billion by the end of 2021. The growth in the business world is expected to increase from 155 million users in 2015 to 843 million by 2021. With that kind of projected growth, revenue is forecasted to grow from $1.6 billion in 2015 to $15.8 billion in 2021.According to Global Market Insights, Inc., between 2016 and 2024, the market share for the technology will grow at an annual rate of almost 35 percent. More and more sectors of the economy, like healthcare and the automotive industry, are finding uses for the speech recognition technology in addition to those found in devices like smart speakers and phones.

**TABLE OF CONTENT:**

**Abstract…………...........................…………………...………. i**

**Acknowledgement………......................……………………… ii**

**Declaration………………………………..................………… iii**

**1. Introduction………………………………......................…… 1**

1.1. Overview……………………...…………….................. 2

1.2. Motivation ……………………....................………….. 3

1.3. Problem Statement…………………….…..................... 4

1.4. Objective…………………………………..................... 5

**2. Software Requirement Analysis…………………………..... 6**

2.1. System Analysis………………………………………. 7

2.2. Classification……………………………………......... 8

2.3. Create database……………………………………….. 9

2.4. Connecting microphone……………………………… 10

**3. Requirements…………….……………..………................... 11**

3.1. Hardware………………………………....................... 11

3.2. Software………………………………........................ 12

3.3. Operating system………………………....................... 13

3.4. Application…………………………............................ 14

3.5. Technology…………………………......................... .. 15

**4. Results……………….…………………………..............….. 16**

**Introduction:**

Speech is the most basic means of adult human communication. The basic goal of speech processing is to provide an interaction between a human and a machine. Speech recognition allows the machine to catch the words, phrases and sentences. Natural language processing allows the machine to understand what we speak, and speech synthesis to allow the machine to speak. An intelligent virtual assistant (IVA) or intelligent personal assistant (IPA) is a software agent that can perform tasks or services for an individual based on commands or questions. Sometimes, the term "chatbot” is used to refer to virtual assistants generally or specifically accessed by online chat. In some cases, online chat programs are exclusively for entertainment purposes. Some virtual assistants are able to interpret human speech and respond via synthesized voices. Users can ask their assistants questions, control home automation devices and media playback via voice, and manage other basic tasks such as email, to-do lists, and calendars with verbal (spoken?) commands.[[1]](https://en.wikipedia.org/wiki/Virtual_assistant#cite_note-1)As of 2017, the capabilities and usage of virtual assistants are expanding rapidly, with new products entering the market and a strong emphasis on both email and voice user interfaces. Apple and Google have large installed bases of users on smart phones. Microsoft has a large installed base of Windows-based personal computers, smart phones and smart speakers. Amazon has a large install base for smart speakers.[[2]](https://en.wikipedia.org/wiki/Virtual_assistant#cite_note-2) Conversica has over 100 million engagements via its email and SMS interface Intelligent Virtual Assistants for business. In this project we make use of the above feature to create an Automatic Speech Recognition System to make our systems make use of our voice commands to perform real life tasks. For example sending an E-mail, telling what time is it, Searching for something on Wikipedia and opening files present on our Systems etc.

We were motivated to create this project through inbuilt AI like Microsoft Cortana, Google Assistant and we are quite interested in creating our own AI Voice Recognition System using Python as our primary language. We were also intrigued to work on this project because of its many future prospects and applications.

**1.2 Benefits / Drawbacks**

* **Benefits**

### In-car systems

* Simple voice commands may be used to initiate phone calls, select radio stations or play music from a compatible smart phone, MP3 player or music-loaded flash drive.

### Military

* In these programs, speech recognizers have been operated successfully in fighter aircraft, with applications including: setting radio frequencies, commanding an autopilot system, setting steer-point coordinates and weapons release parameters, and controlling flight display.

### Usage in education and daily life User:

* Speech recognition can allow students with learning disabilities to become better writers. By saying the words aloud, they can increase the fluidity of their writing, and be alleviated of concerns regarding spelling, punctuation, and other mechanics of writing.

### Keep your home safe

* You can connect your virtual assistant to your smart home security devices to make it easier to keep your property safe. For example, you can connect the Blink security camera system to any Amazon Alexa device and once the Blink 'skill' is enabled on your Alexa app you can give voice commands such as 'Alexa, ask Blink to arm my home security system'.

### Save money on your energy bills

* Virtual assistants can control smart home devices, including next-gen lighting options. You could hook up a smart thermostat such as the Nest to Google Home or Alexa and tell your VPA when you're leaving the house, so that your heating gets turned off. Tests suggest that UK users can reduce their energy usage by as much 16.5 per cent, saving cash in the long run.

**What’s causing the Shift towards Voice?**

The main driver of the shift towards voice user interfaces is the changing user demands. 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, the mass adoption of artificial intelligence in users’ everyday lives is also fueling the shift towards voice applications. The number of IoT devices such as smart thermostats, appliances, 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 however, it only starts there. Many industry experts even [predict that nearly every application will integrate voice technology in some way in the next 5 years.](https://www.forbes.com/forbes/welcome/?toURL=https://www.forbes.com/sites/groupthink/2017/09/29/your-mobile-strategy-needs-to-include-voice/&refURL=https://www.google.ca/&referrer=https://www.google.ca/)Applications of this technology are seen everywhere, so where will it take us in 2020 and beyond? We provide a high-level overview of the potential that voice has and 7 key predictions we think will take off in the coming years.

**Application Area**

### Streamlined Conversations

Both Google and Amazon recently announced that both assistants will no longer require the use of repeated “wake” words. Previously both assistants were dependent on a wake word (Alexa or Ok, Google) to initiate a new line of conversation. For example, one would have to ask “Alexa, what’s the current temperature at the hallway thermostat?” and then have to say, “Alexa” again before requesting that the voice assistant to “set the hallway thermostat to 23 degrees.” It would be more convenient and natural for the user to say, “Alexa, what’s the current temperature at the hallway thermostat?” and then simply say “set my hallway thermostat to 23 degrees,” without requiring the wake word again, and now that’s possible. Consumers use voice assistants in specific locations, usually while multitasking, and can either be alone or amongst a group of people when using them. Having devices that can decipher these

Contextual factors make a conversation more convenient and efficient with these devices, but it also shows that developers behind the technology are aiming to provide a more user-centric experience.

### 2. Compatibility and Integration

### When it comes to integrating voice technology with other products, Amazon has been ahead of the game. Those who use Alexa will be familiar with the fact that the voice assistant is already integrated into a vast array of products including Samsung’s Family Hub refrigerators. Google has finally caught on and has announced Google Assistant Connect. The idea behind this technology is for manufacturers to create custom devices that serve specific functions and are integrated with the Assistant. In 2020, we will see a greater interest in the development of voice-enabled devices. This will include an increase in mid-level devices: devices that have some assistant functionality but aren’t full-blown smart speakers. Instead, they communicate with your smart speaker, display or even perhaps your phone over Bluetooth where the processing happens on those devices. Amazon is already well on its way with an Alexa-enabled wall clock.

### Search Behaviors Will Change

Voice search has been a hot topic of discussion. Visibility of voice will undoubtedly be a challenge. This is because the visual interface with voice assistants is missing. Users simply cannot see or touch a voice interface unless it is connected to the Alexa or Google Assistant app. Search behaviors, in turn, will see a big change. In fact, if tech research firm [Juniper Research is correct, voice-based ad revenue could reach $19 billion by 2022](https://techcrunch.com/2017/11/08/voice-enabled-smart-speakers-to-reach-55-of-u-s-households-by-2022-says-report/), thanks in large part to the growth of voice search apps on mobile devices. Brands are now experiencing a shift in which touch points are transforming to listening points, and organic search will be the main way in which brands has visibility. ComScore data even reveals that [50% of all search will be via voice tech by 2020.](https://searchengineland.com/voice-search-explosion-will-change-local-search-251776) As voice search grows in popularity, advertising agencies and marketers expect Google and Amazon will open their platforms to additional forms of paid messages.

### Individualized Experiences

Voice assistants will also continue to offer more individualized experiences as they get better at differentiating between voices. Google Home is able to support up to six user accounts and detect unique voices, which allows Google Home users to customize many features. Users can ask “What’s on my calendar today?” or “tell me about my day?” and the assistant will dictate commute times, weather, and news information for individual users. It also includes features such as nicknames, work locations, payment information, and linked accounts such as Google Play, Spotify, and Netflix. Similarly, for those using Alexa, simply saying “learn my voice” will allow users to create separate voice profiles so the technology can detect who is speaking for more individualized experiences.

### Voice Push Notifications

We’ve previously discussed the method of using [user-centric push notifications](https://clearbridgemobile.com/how-to-design-a-user-centric-push-notifications-strategy/) as a means to re-engage users with your app, voice technology presents a unique means of distributing push notifications. As a way to increase user engagement and retention, push notifications simply remind users of the

app and display relevant messaging to the user. Now that both Google Assistant and Amazon’s Alexa allow the user to enable spoken notifications for any third-party app that has the compatibility, users can hear notifications rather than read them. These notifications are generally related to calendar appointments or new content from core features.

### Touch Interaction

CES 2019 continued to prove that voice and visual displays are merging into one seamless experience. This year Google showcased what is being called the [E Ink screen](https://www.theverge.com/2019/1/8/18172195/google-assistant-connect-e-ink-screen-gadget-smarthome-ecosystem-ces-2019). This display can show the weather, local traffic information, or calendar events. The push to bring visual and voice capabilities together allow users to further interact with the assistant.

### Security Will Be a Focus

Forty-one percent of voice assistant users are concerned about trust and privacy according to a [report](https://about.ads.microsoft.com/en-us/blog/post/april-2019/new-report-tackles-tough-questions-on-voice-and-ai) from Microsoft. With news from Google I/O and Amazon’s MARS conferences announcing that assistants will essentially be able to plan an entire evening, for example, find local movie times, buy tickets, book a restaurant reservation and schedule an Uber, concerns regarding payments and sensitive information are valid. Voice payments, in particular, will become more secure and convenient for users to make purchases. Speaker verification and ID will also become a paramount as part of the voice assistant experience with more security being built around the user.

## Why Adopt A Mobile Voice Strategy?

Mobile phones are already personalized, more so than any website. Additionally, there is very little screen space on mobile, making it more difficult for users to search, or navigate. With larger product directories and more information, voice applications enable consumers to use natural language to eliminate or reduce manual effort, making it a lot faster to accomplish tasks. [Rogers has introduced voice commands to their remotes](https://www.rogers.com/customer/support/article/using-voice-commands) allowing customers to quickly browse and find their favorite shows or the latest movies with certain keywords, for example, an actor’s name. Brands need to focus on better mobile experiences for their consumers and voice is the way to do so. Users are searching for quicker and more efficient ways of accomplishing tasks and voice is quickly becoming the ideal channel for this. Whether that’s finding out information, making a purchase, or achieving a task, voice is the new mobile experience. It’s clear that brands are racing to figure out their voice strategy. With [over 100 million Alexa devices being sold](https://techcrunch.com/2019/01/04/more-than-100-million-alexa-devices-have-been-sold/) alone, there’s a reason why businesses are looking to catch up.

## Voice User Interface (VUI) Will Continue to Advance

Even with just that handful of simple scenarios, it’s easy to see why voice assistants are shaping up to become the hubs of our connected homes and increasingly connected lives .Voice technology is becoming increasingly accessible to developers. For example, Amazon offers [Transcribe](https://aws.amazon.com/transcribe/), an automatic speech recognition (ASR) service that enables developers to add speech-to-text capability to their applications. Once the voice capability is integrated into the application, users can analyze audio files and in return, receive a text file of the transcribed speech. Google has made moves in making Assistant more ubiquitous by opening the software development kit through [Actions](https://developers.google.com/actions/), which allows developers to build voice into their own products that support artificial intelligence.

Another one of Google’s speech-recognition products is the AI-driven [Cloud Speech-to-Text](https://cloud.google.com/speech-to-text/) tool which enables developers to convert audio to text through [deep learning](https://www.techemergence.com/deep-learning-in-the-enterprise-current-traction-and-challenges/)neural network algorithms. This is only the beginning of voice technology as we will see major advancements in the user interface in the years to come. With the advancements in VUI, companies need to start educating themselves on how they can best leverage voice to better interact with their customers. It’s important to ask what the value of adding voice will be as it doesn’t always make sense for every brand to adopt.

How can you provide value to your customers? How are you solving their [pain points](https://clearbridgemobile.com/finding-opportunity-market-identifying-user-pain-points/) with voice? Will voice enhance the user experience or frustrate the user. In 2020, voice-enabled apps will not only accurately understand what we are saying, but how we are saying it and the context in which the inquiry is made. However, there are still a number of barriers that need to be overcome before voice applications will see mass adoption. Technological advances are making voice assistants more capable particularly in AI, natural language processing (NLP), and machine learning. To build a robust speech recognition experience, the artificial intelligence behind it has to become better at handling challenges such as accents and background noise. And as consumers are becoming increasingly more comfortable and reliant upon using voice to talk to their phones, cars, smart home devices, etc., voice technology will become a primary interface to the digital world and with it, [expertise for voice interface design and voice app development will be in greater demand.](https://clearbridgemobile.com/services/)

**About Python 3.7**

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). Python is named after a TV Show called ëMontyPythonís Flying Circusí and not after Python-the snake.

Python 3.0 was released in 2008. Although this version is supposed to be backward incompatibles, later on many of its important features have been back ported to be compatible with version 2.7.This tutorial gives enough understanding on Python 3 version programming language. Please refer to this link for our Python 2 tutorial.

**Why to Learn Python 3?**

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

Python is a MUST for students and working professionals to become a great Software Engineer especially when they are working in Web Development Domain. I will list down some of the key advantages of learning Python:

Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

**Characteristics of Python**

Following are important characteristics of python −

It supports functional and structured programming methods as well as OOP.

It can be used as a scripting language or can be compiled to byte-code for building large applications.

It provides very high-level dynamic data types and supports dynamic type checking.

It supports automatic garbage collection.

It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

|  |
| --- |
|  |
|  |

**Project Implementation**

**Implementation Details:**

**Pyttsx3 module:**

Pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and 3.

**Speech recognition module:**

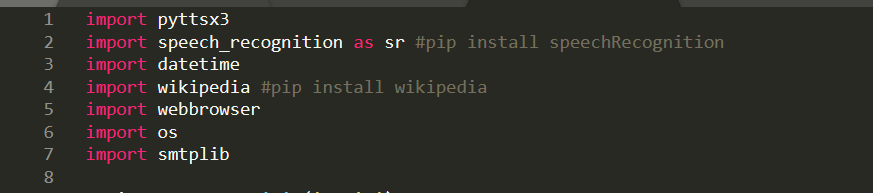
Library for performing speech recognition, with support of several engines and APIs, online and offline.

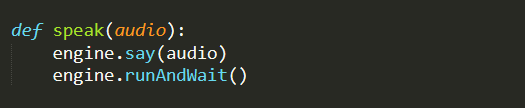
**Os module:**

The OS module in python provides functions for interacting with the operating system. OS, comes under Python’s standard utility modules. This module provides a portable way of using operating system dependent functionality. The \*os\* and \*os.path\* modules include many functions to interact with the file system.

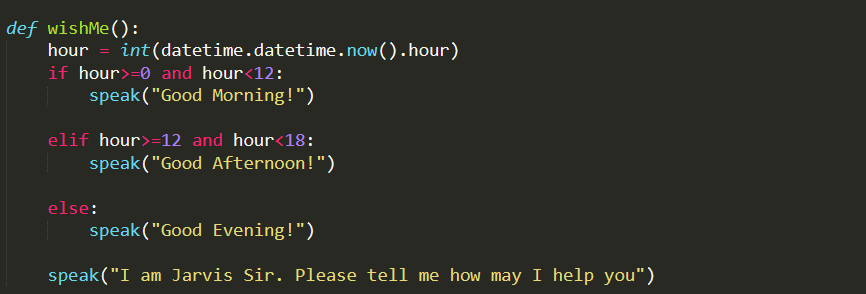
**Smtplib module:**

This script simply checks whether a website is up or not. If it is up then it will send an email about this, if it is down then it will keep checking and when the site will be up, it will send an email and terminate.





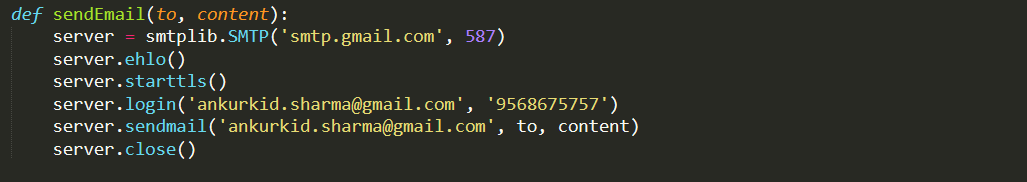
Speak function take input as audio in audio format from microphone



Wishme function will automaticall called on the run time and returns output as audio. Automatically wishes condition based on global time



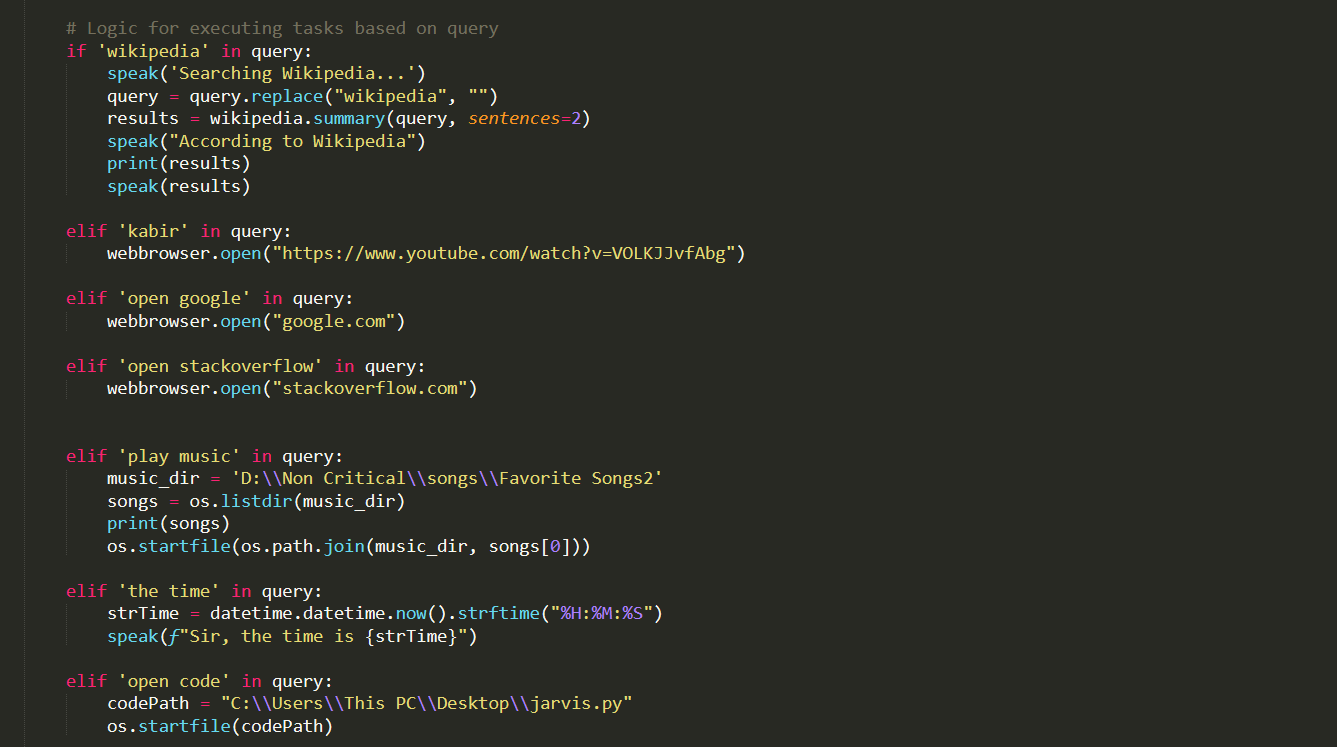
Takecommand take audio or command given to compiler stored in a variable query and Return it and that particular variable is further used in upcoming condition to perform Some task depend on condition user want And also try exception rule is used whether speech is recognized accurately otherwise it Will show message to say again to user to give again clear voice input

****

This sendEmail function is main definition of program to send a email to which person And what content user want to send by just calling by name and content of email This function must be hided from other because it content user privacy email and Password through which user send his mail from mentioned email address



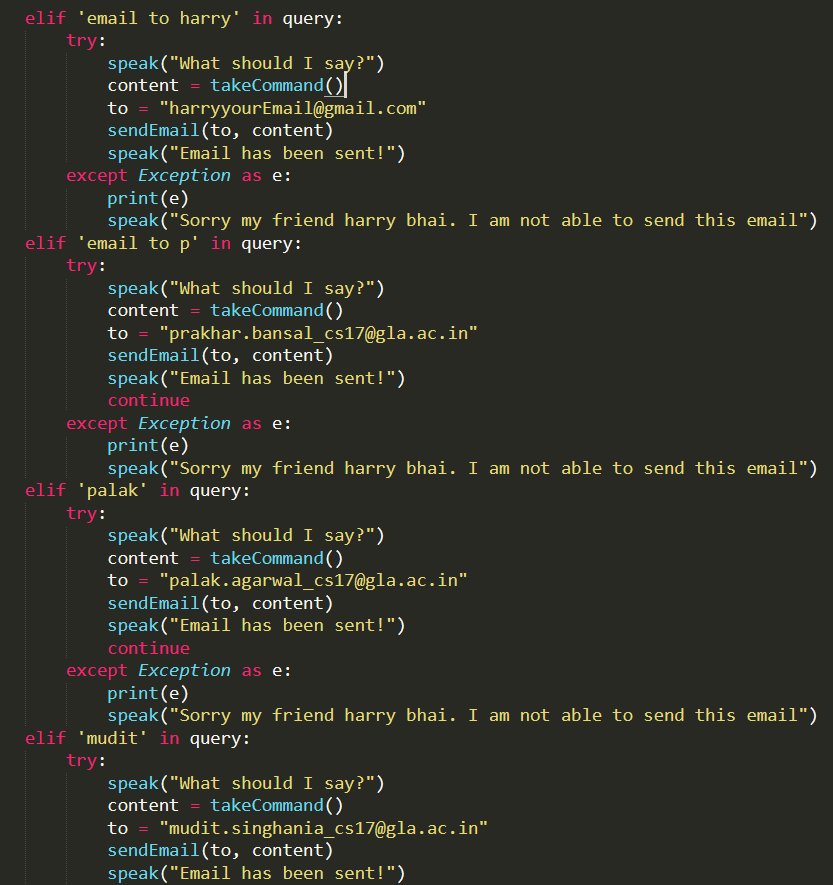
As our voice recognition system open with wish by calling function wishme() And above we explain *takecommand* return a voice into a string format stored in query Variable



Given above condition user want to perform some task by system

Above condition is checked in *elseif* condition with the help of query variable

If query matched with condition than particular task is performed

****

Here is above condition for query name mentioned people to whom email is send Using try exception condition and using content also what message is to be send to a particular person with their proper email address with a message email has been sent

**Project Code**

import pyttsx3

importspeech\_recognition as sr #pip install speechRecognition

importdatetime

importwikipedia #pip install wikipedia

importwebbrowser

importos

importsmtplib

from selenium import webdriver

frompynput.mouse import Button , Controller

m = Controller()

engine = pyttsx3.init('sapi5')

voices = engine.getProperty('voices')

# print(voices[1].id)

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

def speak(audio):

engine.say(audio)

engine.runAndWait()

defwishMe():

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

if hour>=0 and hour<12:

speak("Good Morning!!!")

elif hour>=12 and hour<18:

speak("Good Afternoon!!!!")

else:

speak("Good Evening!!!!!")

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

deftakeCommand():

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

r = sr.Recognizer()

withsr.Microphone() as source:

print("Listening...")

r.pause\_threshold = 1

audio = r.listen(source)

try:

print("Recognizing...")

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

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

if 'how are you' not in query and 'tell me about yourself' not in query and "sing for me" not in query:

speak(query)

except Exception as e:

# print(e)

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

return "None"

return query

def sendEmail(to, content):

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

server.ehlo()

server.starttls()

server.login('swapnilsaxena1998@gmail.com', '830@yaswapnil')

server.sendmail(‘swapnilsaxena1998@gmail.com', to, content)

server.close()

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

wishMe()

while True:

# if 1:

query = takeCommand().lower()

# Logic for executing tasks based on query

if 'wikipedia for' in query:

speak('Searching Wikipedia...')

query = query.split(" ")

location = query[2:]

location=' '.join(location)

webbrowser.open('https://en.wikipedia.org/wiki/'+ location)

text = wikipedia.summary(query)

print(text)

speak(text)

elif 'this' in query:

query = query.split(" ")

c=query[1:]

location=' '.join(c)

webbrowser.open('https://www.youtube.com/results?search\_query=' + location)

elif 'open google' in query:

webbrowser.open("google.com")

elif 'open stackoverflow' in query:

webbrowser.open("stackoverflow.com")

elif 'open gla website' in query:

webbrowser.open("www.gla.ac.in")

elif 'how are you' in query:

speak('i am fine what can i do for you')

elif 'tell me about yourself' in query:

speak('i am jarvis your personal virtual assistant created to help you')

elif 'open youtube on' in query:

query=query.split(" ")

location = query[3:]

location=' '.join(location)

webbrowser.open("https://www.youtube.com/results?search\_query=" + location)

m.position = (395,306)

for \_ in range(10000):

for i in range(10000):

pass

m.press(Button.left)

m.release(Button.left)

elif "where is" in query:

query = query.split(" ")

location = query[2:]

location=' '.join(location)

speak("Hold on Sir, I will show you where " + location + " is.")

webbrowser.open('https://www.google.com/maps/place/'+ location)

elif "record audio" in query:

r = sr.Recognizer()

withsr.Microphone() as source:

speak("Say something!")

audio = r.listen(source)

# Speech recognition using Google Speech Recognition

data = ""

try:

# Uses the default API key

# To use another API key: `r.recognize\_google(audio, key="GOOGLE\_SPEECH\_RECOGNITION\_API\_KEY")`

data = r.recognize\_google(audio)

speak("i recorded: " + data)

speak("do you want to save this audio")

withsr.Microphone() as s:

ad=r.listen(s)

y=r.recognize\_google(ad)

if "yes" in y:

speak("file has been successfully saved")

exceptsr.UnknownValueError:

print("Google Speech Recognition could not understand audio")

exceptsr.RequestError as e:

print("Could not request results from Google Speech Recognition service; {0}".format(e))

elif 'open playlist' in query:

music\_dir = "C:\\Users\\This PC\\Desktop\\new\\"

songs = os.listdir(music\_dir)

print(songs)

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

elif 'the time' in query:

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

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

elif 'open code' in query:

codePath = "C:\\Users\\This PC\\Desktop\\jarvis.py"

os.startfile(codePath)

elif 'email to annan' in query:

try:

speak("What should I say?")

content = takeCommand()

to = "mohd.annanjix@gmail.com"

sendEmail(to, content)

speak("Email has been sent!")

except Exception as e:

print(e)

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

elif 'email to palak' in query:

try:

speak("What should I say?")

content = takeCommand()

to = "palak.agarwal\_cs17@gla.ac.in"

sendEmail(to, content)

speak("Email has been sent!")

continue

except Exception as e:

print(e)

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

elif 'priyanka' in query:

try:

speak("What should I say?")

content = takeCommand()

to = "priyanka.gupta\_cs17@gla.ac.in"

sendEmail(to, content)

speak("Email has been sent!")

continue

except Exception as e:

print(e)

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

elif 'email to priyanka' in query:

try:

speak("What should I say?")

content = takeCommand()

to = "priyanshu.jain\_cs17@gla.ac.in"

sendEmail(to, content)

speak("Email has been sent!")

continue

except Exception as e:

print(e)

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

elif 'email to myself' in query:

try:

speak("What should I say?")

content = takeCommand()

to = "swapnil.saxena\_cs17@gla.ac.in"

sendEmail(to, content)

speak("Email has been sent!")

continue

except Exception as e:

print(e)

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

elif 'email to ayushi' in query:

try:

speak("What should I say?")

content = takeCommand()

to = "ayushisaxenalko@gmail.com"

sendEmail(to, content)

speak("Email has been sent!")

continue

except Exception as e:

print(e)

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

elif 'tell me about' in query:

try:

speak('this is what i found')

query=query.split(" ")

content=query[3:]

webbrowser.open('https://www.google.com/search?q=' + str(\*content))

except Exception as e:

print(e)

speak("sorry")

elif 'good job jarvis' in query:

speak('thank you sir')

elif 'open word' in query:

speak("opening Microsoft word")

os.startfile('C:\\ProgramData\\Microsoft\\Windows\\Start Menu\\Programs\\Microsoft Office\\Microsoft Word 2010.lnk')

elif 'open Excel' in query:

speak("opening Microsoft Excel")

os.startfile('C:\\ProgramData\\Microsoft\\Windows\\Start Menu\\Programs\\Microsoft Office\\Microsoft Excel 2010.lnk')

elif 'open powerpoint' in query:

speak("opening Microsoft powerpoint")

os.startfile('C:\\ProgramData\\Microsoft\\Windows\\Start Menu\\Programs\\Microsoft Office\\Microsoft Powerpoint 2010.lnk')

elif 'open notepad' in query:

speak("opening windows notepad")

os.startfile('C:\\Users\\This PC\\AppData\\Roaming\\Microsoft\\Windows\\Start Menu\\Programs\\Accessories\\Notepad.lnk')

elif "shutdown" in query:

speak("do you wish to shutdown your computer")

speak("press enter to shutdown!")

os.system("shutdown /s /t 1")

**References/Bibliography**

* 1. Google ( [https://www.google.com](https://www.google.com/))
  2. GeeksforGeeks ([https://www.geeksforgeeks.org](https://www.geeksforgeeks.org/) )
  3. JavaTpoint ([https://www.javatpoint.com](https://www.javatpoint.com/) )