

JARVIS A.I Assistant

A Project Work

Submitted in Partial fulfillment for the award of
Graduate Degree of Bachelor of Technology (B.Tech.)

In

Computer Science & Engineering (Session: 2021-22)



Submitted By:

Abhay Soni

Ashok kumar Kushwaha

Anand Kushwaha

Under the Guidance of

Dr. Shiwangi Mishra

(Computer Sc. & Engineering)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING SHRI RAM INSTITUTE OF
SCIENCE&TECHNOLOGY, JABALPUR (M.P.)

RAJIVV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P.)

SHRI RAM INSTITUTE OF SCIENCE AND TECHNOLOGY

APPROVED BY AICTE NEW DELHI & GOVT OF M.P. (AFFILIATED TO RGPV UNIVERSITY OF TECHNOLOGY OF M.P)

Near ITI MADHOTAL, JABALPUR-482002(M.P) TELE FAX: 0761-2641707, EMAIL: SRISTJBP@REDIFFMAIL.COM

CERTIFICATE

This is to certify that

Abhay Soni (0215CS101010)

Ashok Kumar Kushwaha (0215CS101011)

Anand Kushwaha (0215CS101012)

Students of 6th semester, Computer Sc. & Engg. S.R.I.S.T., Jabalpur have duly completed their Minor Project entitled "JARVIS A.I ASSISTANT " for the Completion of 6th Semester examination under the requirement for the degree of Bachelor of Technology as per R.G.P.V., Bhopal.

They have successfully implemented and tested this project, which meets all the requirements specified under my guidance.

Dr. Shiwangi Mishra

(Project Guide)

Computer Sc.& Engg Deptt, SRIST,

Jabalpur

Prof. Prateek Gupta (H.O.D)

Computer Sc. & Engg Deptt

SRIST, Jabalpur

SHRI RAM INSTITUTE OF SCIENCE AND TECHNOLOGY

APPROVED BY AICTE NEW DELHI & GOVT OF M.P.(AFFILIATED TO RGPV UNIVERSITY OF TECHNOLOGY OF M.P)

Near ITI MADHOTAL ,JABALPUR-482002(M.P) TELE FAX : 0761-2641707 ,EMAIL: SRISTJBP@REDIFFMAIL.COM

CERTIFICATE

This is to certify that

Abhay Soni (0213CS191003)

Ashok Kumar Kushwaha (0213CS191021)

Anand Kushwaha (0213CS191017)

Students of 6 semester, Computer Sc. & Engg. S.R.I.S.T., Jabalpur have duly completed their Minor Project entitled " **JARVIS A.I Assistant** " for the Completion of 6 Semester examination under the requirement for the degree of Bachelor of Technology as per R.G.P.V Bhopal.

They have successfully implemented and tested this project.

INTERNAL EXAMINER

Date:

EXTERNAL EXAMINER

Date:

SHRI RAM INSTITUTE OF SCIENCE AND TECHNOLOGY

APPROVED BY AICTE NEW DELHI & GOVT OF M.P.(AFFILIATED TO RGPV UNIVERSITY OF TECHNOLOGY OF M.P.)

Near ITI MADHOTAL ,JABALPUR-482002(M.P) TELE FAX : 0761-2641707 ,EMAIL: SRISTJBP@REDIFFMAIL.COM

ACKNOWLEDGEMENT

I express my sincere thanks to Asst. Dr. Shiwangi Mishra , my project guide, who guided me through the project and also gave valuable suggestions and guidance for completing the project. She helped me to understand the intricate issues involved in project making besides effectively presenting it. These intricacies would have been lost otherwise. My project has been success only because of her guidance.

We are highly grateful to Prof. Mr. PRATEEK GUPTA (H.O.D, Dept. of Computer Science &Engg.) for providing us their valuable, suggestions, motivation, &ideas during whole project work.

We are also thankful to the whole computer science department for providing us the technical support to carry out the project work, to let us utilize all the necessary facilities of the institute and guidance at each and every step during the project work.

Abhay Soni (0213CS191003)

Ashok Kumar Kushwaha (0213CS191021)

Anand Kushwaha (0213CS191017)

TABLE OF CONTANT

1. Introduction

- 1.1. Overview
- 1.2. Problem Statement
- 1.3. Scope

2. Analysis

- 2.1. Objective of project
- 2.2. Working
- 2.3. Software Requirement Specification
- 2.4. Feasibility Study
- 2.5. Software Engineering Paradigm
- 2.6. Software Development Life Cycle

3. Design

- 3.1 SYSTEM DESIGN AND IMPLEMENTATION
- 3.2 DATA FLOW DIAGRAM DFD
- 3.3 FLOW CHART

4. Implementation

- 4.1 HARDWARE AND SOFTWARE REQUIREMENTS
- 4.2 Tools Description
- 4.3. Process Logic of Modules

5. Result

- 5.1. Snapshot

6. Source Code

ABSTRACT

The Most famous application of iPhone is “SIRI” which helps the end user to communicate end user mobile with voice and it also responds to the voice commands of the user. Same kind of application is also developed by the Google that is “Google Voice Search” which is used for in Android Phones. But this Application mostly works with Internet Connections. But our Proposed System has capability to work with Internet Connectivity. It is named as "JARVIS A.I Assistant “ with Voice Recognition Intelligence, which takes the user input in form of voice or text and process it and returns the output in various forms like action to be performed or the search result is dictated to the end user. In addition, this proposed system can change the way of interactions between end user and the computer . The system is being designed in such a way that all the services provided by the Computer are accessible by the end user on the user's voice commands.

Keywords: JARVIS

1 . Introduction

1.1 OVERVIEW

Today, we can ask voice assistants like Apple's Siri, Google Now to perform simple tasks like, "What's the weather", "Remind me to take pills in the morning", etc. in our own natural language. The next evolution of natural language interaction with voice assistants is in the form of task automation such as "turn on the air conditioner whenever the temperature rises above 30 degrees Celsius", or "if there is motion on the security camera after 10pm, call Bob". A voice assistant is a digital assistant that uses voice recognition, speech synthesis and natural language processing (NLP) to provide a service through a particular application. Now everyone wants to have an assistant who listen our call, anticipates our needs and can take necessary action when needed. This luxury life is now available with the help of Artificial Intelligence based on voice assistant. Voice assistants come's in small packages and can perform a variety of actions after hearing our commands. They can launch apps, open web browser, answer basic informational queries, tell horoscope, calculate your BMI, answer our queries, play music, send email, set reminders, make lists, and do basic math calculations, etc.

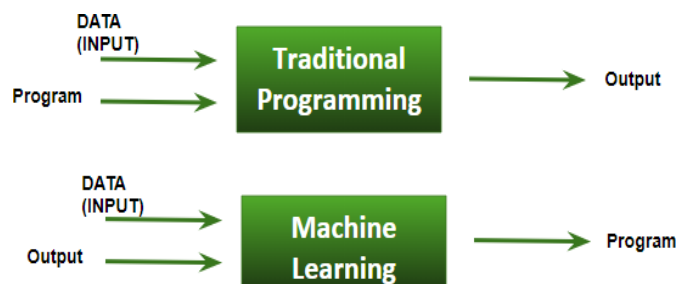
Example: Training of students during exam.

While preparing for the exam students don't actually cram the subject but try to learn it with complete understanding. Before the examination, they feed their machine(brain) with a good amount of high-quality data (questions and answers from different books or teachers notes or online video lectures).

Actually, they are training their brain with input as well as output i.e. what kind of approach or logic do they have to solve a different kind of questions. Each time they solve practice test papers and find the performance (accuracy /score) by comparing answers with answer key given, Gradually, the performance keeps on increasing, gaining more confidence with the adopted approach.

That's how actually models are built, train machine with data (both inputs and outputs are given to model) and when the time comes test on data (with input only) and achieves our model scores by comparing its answer with the actual output which has not been fed while training.

Researchers are working with assiduous efforts to improve algorithms, techniques so that these models perform even much better



Basic Difference in ML and Traditional Programming?

Traditional Programming

We feed in DATA (Input) + PROGRAM (logic), run it on machine and get output.

Machine Learning

We feed in DATA(Input) + Output, run it on machine during training and the machine creates its own program(logic), which can be evaluated while testing.

Advantages

Voice Assistant allows you to gain the perks of high-end technology and its functionalities. Our proposed application points to many advantages: 1. Our proposed application provides security to the user as it can authenticate the authorized user using Face Recognition technique. 2. The face recognition technology make the system secure and robust for the user as this does not required any input from the user through keyboard or mouse. 3. The application provides flexibility to the user as it can send email just listening the command given by the user. 4. Our proposed application stores personal information such as location data, reminders and contacts in the notebook. 5. This application includes the functions and services such as: opening system application, event handler, location services, music player service, checking weather, Google search, Wikipedia search, tell horoscope, general conversation and help menu

1.1 PROBLEM STATEMENT

Virtual Personal Assistant

The voice assistant is design to make the work easier of the user. As user can give command to them without making visual access to the screen. The biggest disadvantage of this system is that confidential data can be accessed by unauthorized user so the privacy can be breached. Due to this, the confidentiality, integrity and availability (CIA) of user data is affected. Looking to this problem the security features of "Face Recognition" is designed so that it can detect the authorized user face and take user command as input and provide response via a synthesis voice. Facial recognition technology (FRT) is one of the most controversial new tools. It was first developed in the 1960s. It has recently become accessible to the mass market-to both law enforcement and private consumers. Automatic face recognition involves:

KEY TAKEAWAYS

- A virtual assistant is a self-employed worker who specializes in offering administrative services to clients from a remote location, usually a home office.
- Typical tasks a virtual assistant might perform include scheduling appointments, making phone calls, making travel arrangements, and managing email accounts.

- Some virtual assistants specialize in offering graphic design, blog writing, bookkeeping, social media, and marketing services.
- For an employer, one advantage of hiring a virtual assistant is the flexibility to contract for just the services they need.

1.3 SCOPE

Future Scope of Machine Learning

FUTURE SCOPE Using this system as a framework, the system can be expanded to features security. Security is important these days so it can be combined with this system to give more advanced security features. In this, the voice authentication technology can be implemented for more security. More advancement are possible like operating on various tones or accents from different regions that mean it should be able to perform operations on various voice tones and accents.. Further modifications are possible like learning the answer of questions that are not known by the voice assistant and replying whenever next time the same question is put up by the user.

Automotive Industry



The automotive industry is one of the areas where Machine Learning is excelling by changing the definition of 'safe' driving. There are a few major companies such as Google, Tesla, Mercedes Benz, Nissan, etc.

that have invested hugely in Machine Learning to come up with novel innovations. However, Tesla's self-driving car is the best in the industry. These self-driving cars are built using Machine Learning, IoT sensors, high-definition cameras, voice recognition systems, etc.

You just need to sit in the car and enter the location. It will find the best possible route to that location and will ensure to safely drive you to the specified destination. How wonderful it would be to experience such a great creation by humans! This is all possible with the help of Machine Learning

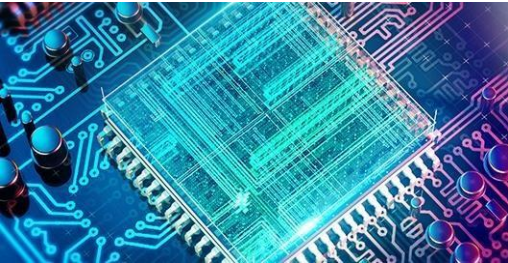
Robotics



Robotics is one of the fields that always gain the interest of researchers as well as the common. In 1954, George Devol invented the first robot that was programmable and it was named as Animate . After that, in the 21st century, Hanson Robotics created the first AI-robot, Sophia. These inventions were possible with the help of Machine Learning and Artificial Intelligence.

Researchers all over the world are still working on creating robots that mimic the human brain. They are using neural networks, AI, ML, computer vision, and many other technologies in this research. In the future, we may come across robots that would be capable of performing various tasks similar to a human.

Quantum Computing



We are still at an infant state in the field of Machine Learning. There are a lot of advancements to achieve in this field. One of them that will take Machine Learning to the next level is Quantum Computing. It is a type of computing that uses the mechanical phenomena of quantum such as entanglement and superposition. By using the quantum phenomenon of superposition, we can create systems (quantum systems) that can exhibit multiple states at the same time. On the other hand, entanglement is the phenomenon where two different states can be referenced to each other. It helps in describing the correlation between the properties of a quantum system.

These quantum systems are built using advanced quantum algorithms that process data at high speed. Fast processing enhances the processing power of Machine Learning models. Thus, the future scope of Machine Learning will accelerate the processing power of the automation system used in various technologies.

Computer Vision

AS the name suggests, [computer vision](#) gives a vision to a computer or a machine. Here comes into our minds what the Head of AI at Google, Jeff Dean, has once said, 'The progress we've made from 26% error in 2011 to 3% error in 2016 is hugely impactful. The way I like to think is, computers have now evolved eyes that work



Giving the ability to a machine to recognize and analyse images, videos, graphics, etc. is the goal of computer vision. The progress in the field of Artificial Intelligence and Machine Learning has made it possible to achieve the goal of computer vision faster.

2. Analysis

2.1 Objective of project

Voice Assistant using python

As we know Python is a suitable language for scriptwriters and developers. Let's write a script for Voice Assistant using Python. The query for the assistant can be manipulated as per the user's need. Speech recognition is the process of converting audio into text. This is commonly used in voice assistants like Alexa, Siri, etc. Python provides an API called [Speech Recognition](#) to allow us to convert audio into text for further processing. In this article, we will look at converting large or long audio files into text using the Speech Recognition API in python.

2.2 How does voice assistant work?

The assistant needs a keyword in order to begin the conversation. The assistant jumps into action when you say a specific keyword. In general, keyword includes an assistant name along with greeting.

In the case of Jay voice assistant, it's simply 'JARVIS'. In a similar fashion, these assistants need a keyword to begin listening.

The assistant then listens to what you say by the microphone (mic) in your laptop. Furthermore, the assistant then interprets or understand what you said by natural language processing. In most of the cases, voice assistants are connected to neural networks via the internet. Neural networks and artificial intelligence together help assistant to understand the context of human language.

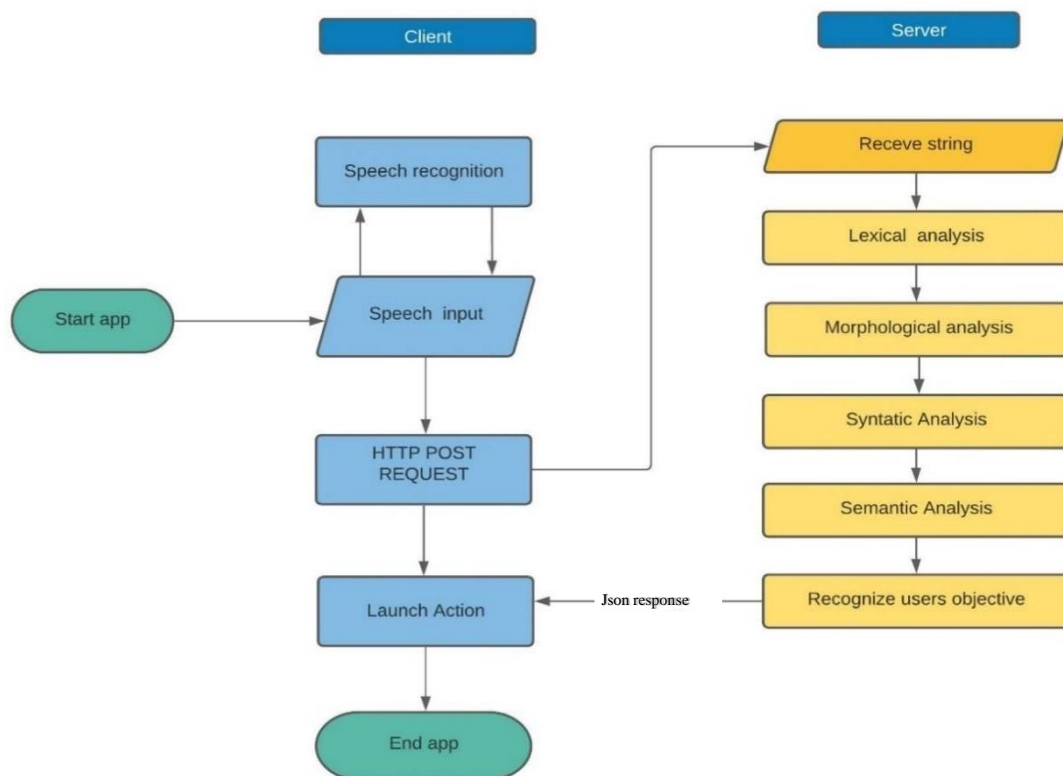
2.3 Software requirement specification

JARVIS Assistant will:

- Welcome note
- Searching on Wikipedia
- Searching on Google and play YouTube video

- Play your music
- Show time
- Run timers and reminders
- Open apps on your windows
- Open website on your windows
- Real-time spoken assistant
- Find place on google maps
- Answer some normal conversation
- Play Audio book
- Tell some Jokes

2.4 Working diagram of JARVIS A.I assistant



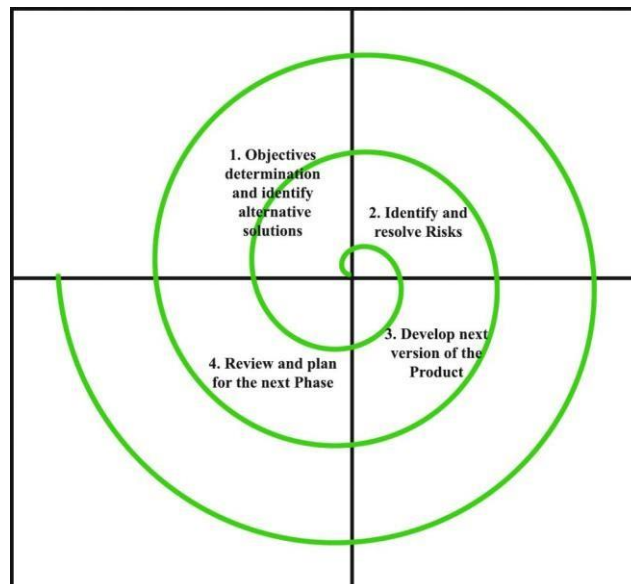
2.5 SDLC MODELS

Spiral Model

Spiral model is one of the most important Software Development Life Cycle models, which provides support for Risk Handling. In its diagrammatic representation, it looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a Phase of the software development process. The exact number of phases needed to develop the product can be varied by the project manager depending upon the project risks. As the project manager dynamically determines the number of phases, so the project manager has an important role to develop a product using the spiral model.

The Radius of the spiral at any point represents the expenses(cost) of the project so far, and the angular dimension represents the progress made so far in the current phase.

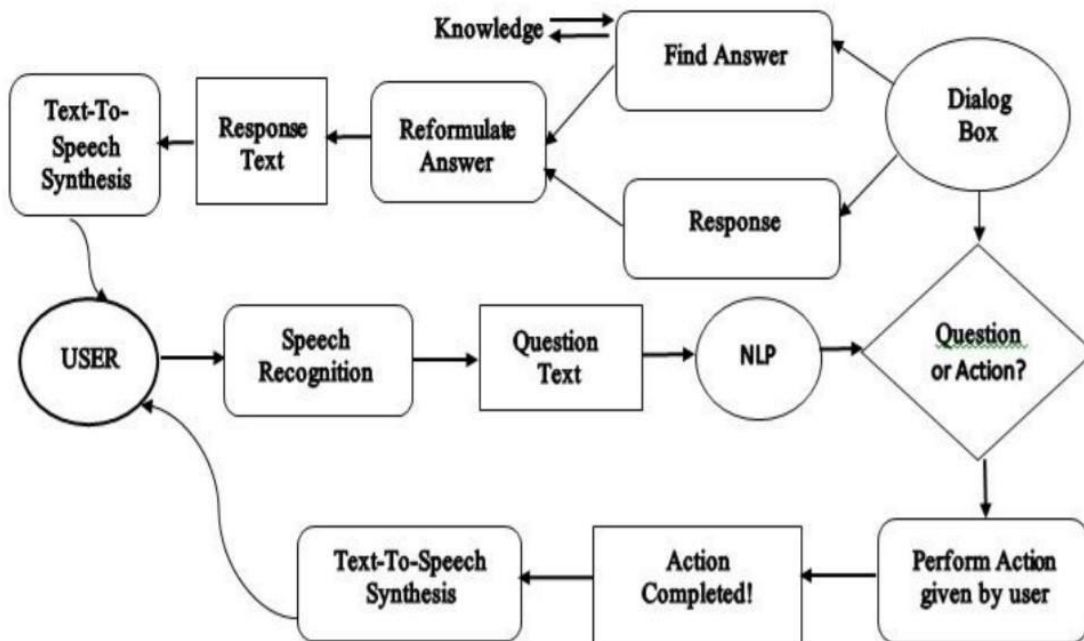
The below diagram shows the different phases of the Spiral Model: –



3. Design

3.1 SYSTEM DESIGN AND IMPLEMENTATION

The proposed model of the voice assistant is as shown in the above figure 1. The model consists of user input through microphone to accept commands from the user. These commands are then go through Speech Recognition, it is the ability of a machine or program to identify words and phrases in spoken languages and convert them to a machine-readable format. On these input Natural Language Processing is applied, it is a field which is created by amalgamating computer science and artificial intelligence. Using NLP, we are concerned with interactions between computers and human natural languages. Then the BRAIN check whether it is a question or an action, if it is a action than the action is performed by the voice assistant and acknowledgment is given to the user via a synthesis voice or if it is a question than it is search in dialog box or knowledge base and then response via a synthesis voice to the user. Our Voice assistant uses google text-to-speech API to understand all the words spoken by the user, and based on certain conditions that satisfy being a command the voice assistant sends responses to the user.



3.2 DATA FLOW DIAGRAM DFD

is a graphical representation which provides information flow between input and output data. It is also known as "Data Flow Chart or Bubble Chart". A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated.

Level 0 DFD:

The user gives the input in the form of voice; this voice command is recognized by the application. Then it will check whether it is the authorized user, then action is performed as per the command given by the user. Command given is compared as a form of action and question and response with the dialog box or search through the knowledge base.

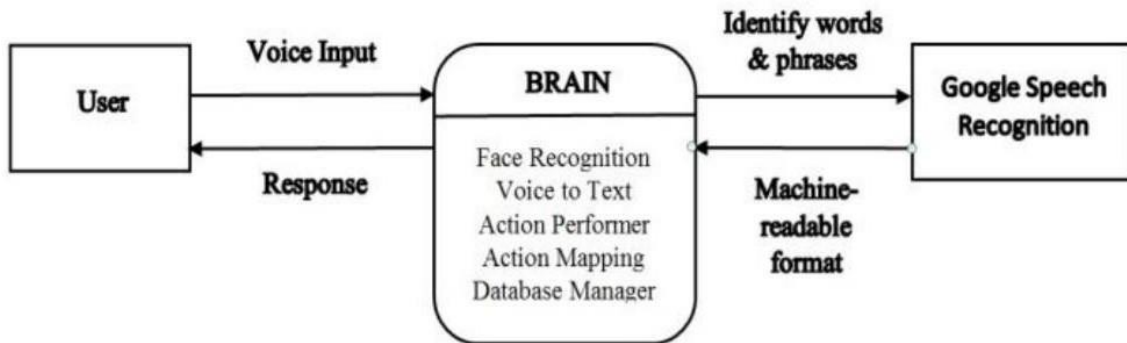
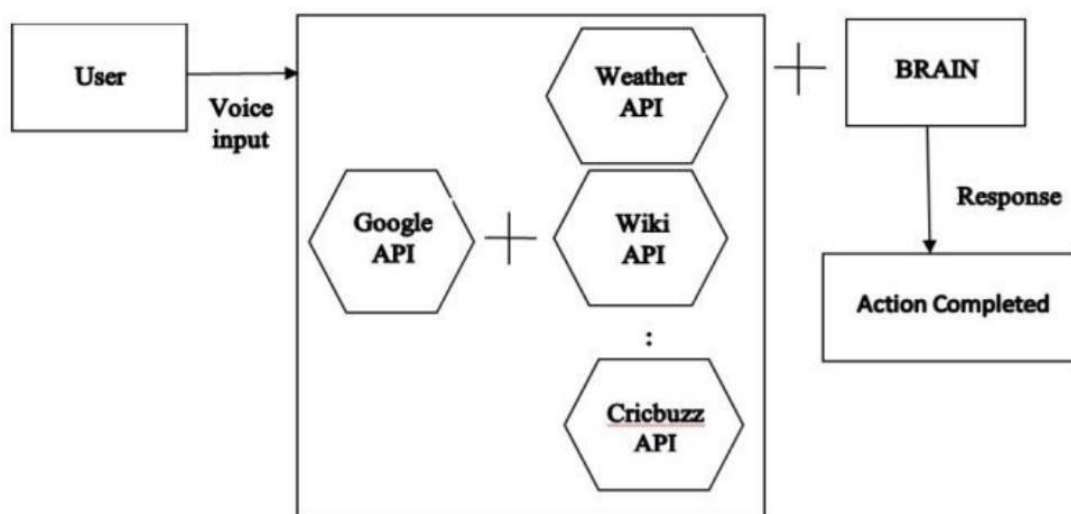


Fig -2: DFD Diagram Level 0

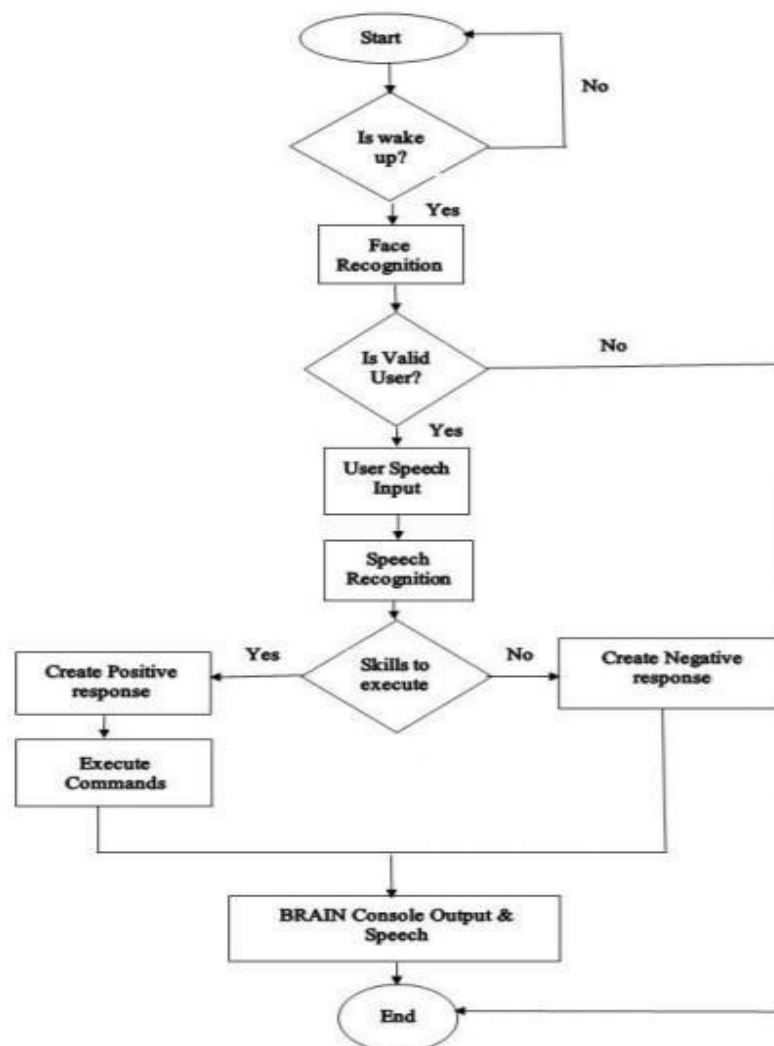
Level 1 DFD:

Input is given by user in the form of voice. [Google Voice API](#) will convert this voice data in text form and then the action is performed by the voice assistant according to the command given by the user by comparing with the dialog box and knowledge base.



3.3 FLOW CHART

Flow chart is the graphical representation of algorithms. Different symbols are used to represent flow chart. As the system is started, it first authenticates the authorized user, then voice assistant is on running in the background listening for available voice commands; once the user gives a command, based on the conditions provided to the voice assistant, the voice assistant gives the necessary output. This output is sent to the Speech Recognition which is convert the speech into machine-readable form. Based on the input received the personal voice assistant then performs the desired task.



4 Implementation

4.1 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.

Hardware:

I3 processor or later.

RAM 4 GB or more.

Mic & speaker

Software:

Windows 10(64-bit).

Python 3.or later

Chrome Driver

Google weather API

4.2 Tools Description

Tools for Making project:

Visual Studio Code

Python

4.3. Process Logic of Modules

Using Python modules

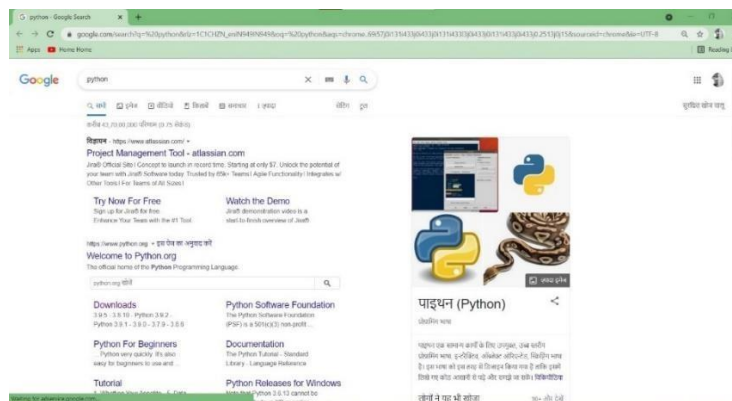
```
from ast import operator
from asyncio import Task
from ctypes.wintypes import HGDIOBJ
from email.mime import audio
from multiprocessing.context import SpawnContext
from re import search
import sys
import operator
import datetime
import pyttsx3
import speech_recognition as sr
import webbrowser
import pywhatkit
import PyPDF4
import os
import requests
import pyautogui
from bs4 import BeautifulSoup
import wikipedia
import pyjokes
from PyQt5 import QtWidgets, QtCore, QtGui
from PyQt5.QtCore import QTimer, QTime, QDate, Qt
from PyQt5.QtGui import QMovie
from PyQt5.QtCore import*
from PyQt5.QtGui import*
from PyQt5.QtWidgets import*
from PyQt5.uic import loadUiType
from QTGUI import Ui_gui1
import AI
```

5. Result

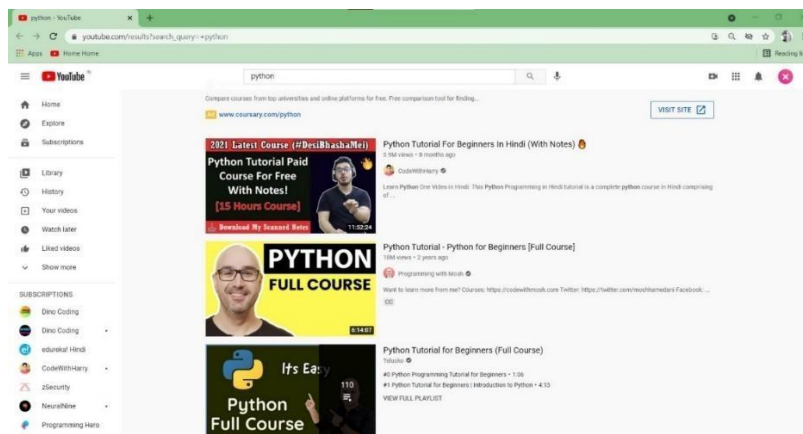
Launch a Web Browser using web browser module

In Python, **web browser module** provides a high-level interface which allows displaying Web-based documents to users. The web browser module can be used to launch a browser in a platform-independent manner as shown below:

Search Python in google



Play Python in google



Wikipedia module in Python

The Internet is the single largest source of information, and therefore it is important to know how to fetch data from various sources. And with Wikipedia being one of the largest and most popular sources for information on the Internet.

Wikipedia is a multilingual online encyclopaedia created and maintained as an open collaboration project by a community of volunteer editors using a wiki-based editing system.

In this article, we will see how to use Python's Wikipedia module to fetch a variety of information from the Wikipedia website.

```
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

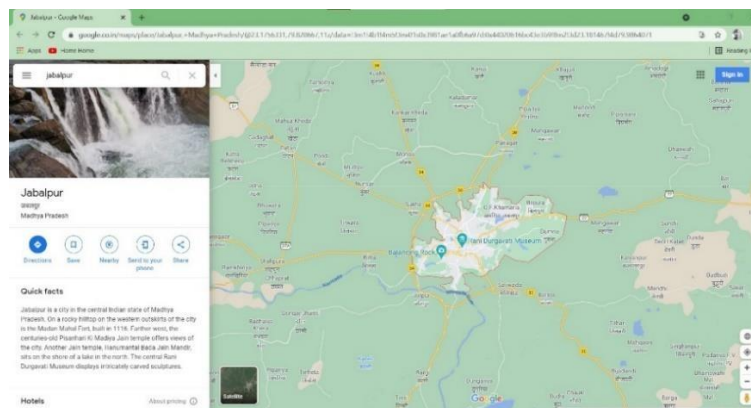
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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

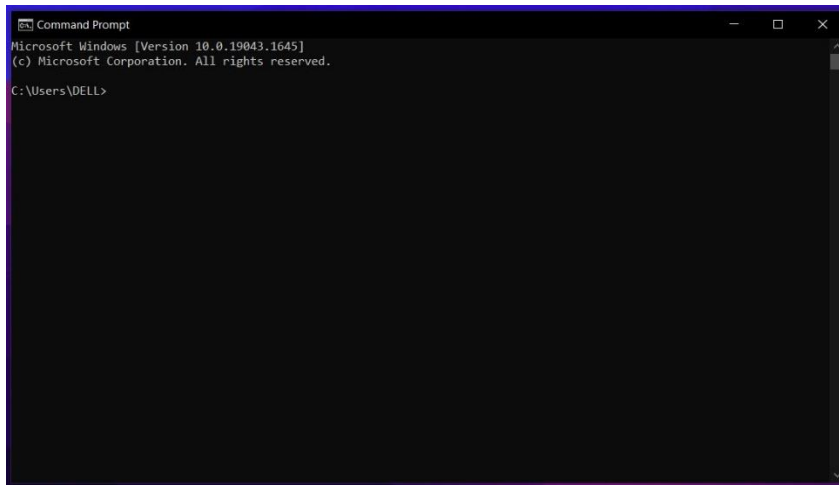
Source Control (Ctrl+Shift+G)
PS C:\python project> & "C:/Program Files/Python39/python.exe" "d:/python project/Jay.py"
Listening.....
Recognizing.....
User said: Samsung Wikipedia

In the Hebrew Bible, Samson (; Hebrew: שִׁמְשׁוֹן, Šimšōn, "man of the sun") was the last of the judges of the ancient Israelites mentioned in the Book of Judges (chapters 13 to 16) and one of the last leaders who "judged" Israel before the institution of the monarchy. He is sometimes considered to be an Israelite version of the popular Near Eastern folk hero also embodied by the Sumerian Enkidu and the Greek Heracles. The biblical account states that Samson was a Nazirite, and that he was given immense strength to aid him against his enemies and allow him to perform superhuman feats, including slaying a lion with his bare hands and massacring an entire army of Philistines using only the jawbone of a donkey.
```

Python script to open a Google Map location on chrome



Open System Software using Os Module



Assistant using Wolfram Alpha API.

The Wolfram Alpha Webservice API provides a web-based API allowing the computational and presentation capabilities of Wolfram Alpha to be integrated into web, mobile, desktop, and enterprise applications. Wolfram Alpha is an API which can compute expert-level answers using Wolfram's algorithms, knowledgebase and AI technology. It is made possible by the Wolfram Language. This article tells how to create a simple assistant application in Python which can answer simple questions like the ones listed below.

```
Listening.....
Recognizing.....
User said: what is the capital of India

New Delhi, Delhi, India
█

Listening.....
Recognizing.....
User said: who is the Prime Minister of India

Narendra Modi (from 26/05/2014 to present)
```

Write a note & show note

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

Listening.....
Recognizing.....
User said: my name is Krishna Vishwakarma

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

Find current weather of any city using Open Weather map API in Python

Open weather map is a service that provides weather data, including current weather data, forecasts, and historical data to the developers of web services and mobile applications.

It provides an API with JSON, XML and HTML endpoints and a limited free usage tier. Making more than 60 calls per minute requires a paid subscription starting at *USD 40* per month. Access to historical data requires a subscription starting at *150 USD* per month. Users can request current weather information, extended forecasts and graphical maps (showing cloud cover, wind speed, pressure and precipitation)

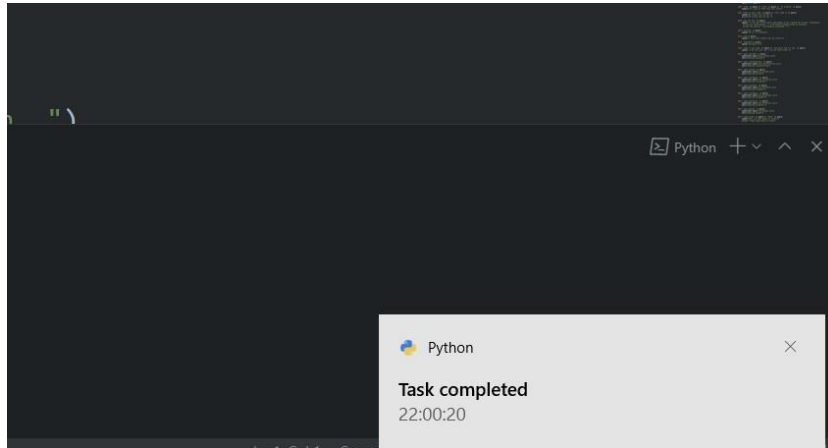
```
Listening.....
Recognizing.....
User said: Jabalpur weather

Temperature is 28°C
Time: Saturday 10:02 pm
Sky Description: Partly cloudy
e
[]
```

Get Current Date and Time using Python

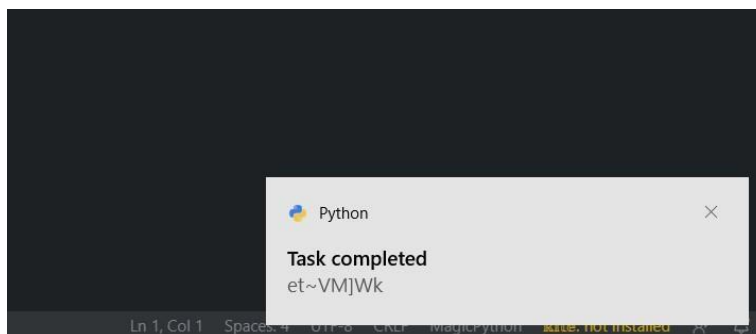
In Python, date and time are not a data type of its own, but a module named `datetime` can be imported to work with the date as well as time. Datetime module comes built into Python, so there is no need to install it externally.

To get both current date and time `datetime.now()` function of `datetime` module is used. This function returns the current local date and time.



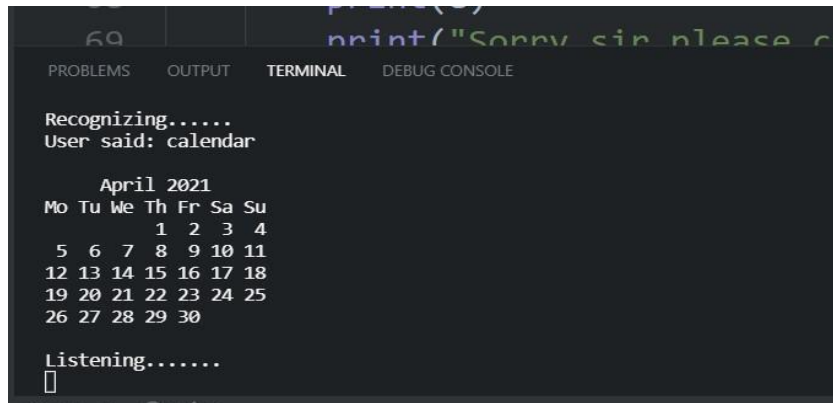
Generating Strong Password using Random Module

Having a weak password is not good for a system that demands high confidentiality and security of user credentials. It turns out that people find it difficult to make up a strong password that is strong enough to prevent unauthorized users from memorizing it.



Show Calendar

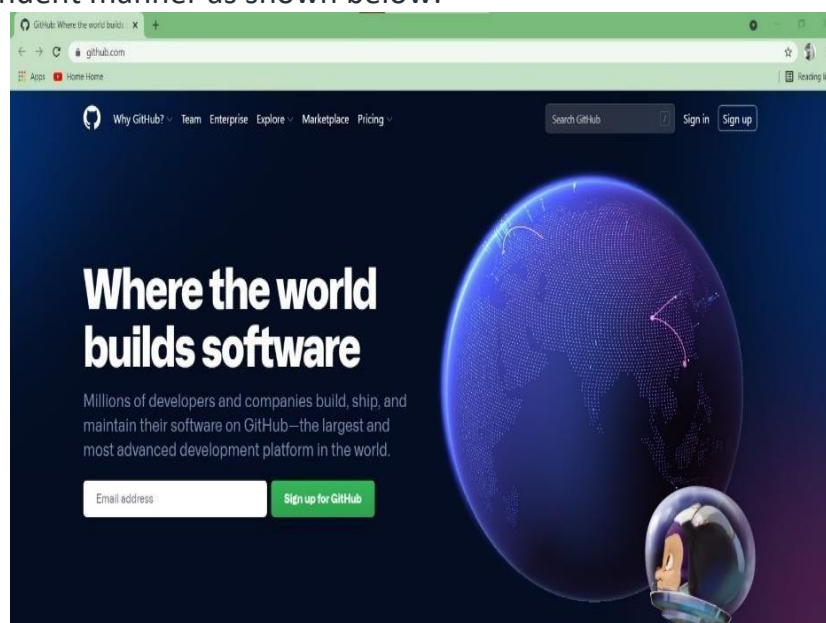
Showing Calendar using Calendar Module



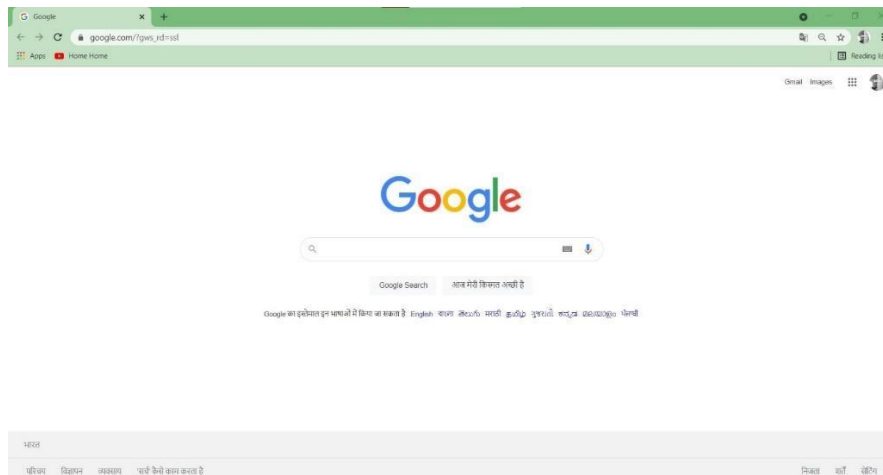
```
69  
print("Sorry sir please ca  
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE  
Recognizing.....  
User said: calendar  
April 2021  
Mo Tu We Th Fr Sa Su  
      1  2  3  4  
5  6  7  8  9 10 11  
12 13 14 15 16 17 18  
19 20 21 22 23 24 25  
26 27 28 29 30  
Listening.....  
█
```

Launch a Web Browser using web browser module

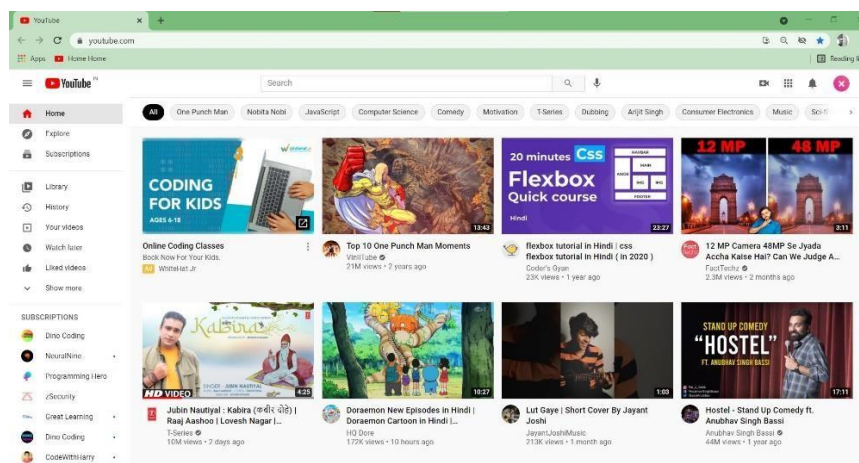
In Python, **web browser module** provides a high-level interface which allows displaying Web-based documents to users. The web browser module can be used to launch a browser in a platform-independent manner as shown below:



Open google



Open You-tube



Python Script to create random jokes using pyjokes

Python supports creation of random jokes using one of its libraries. Let us explore it a little more, **Pyjokes** is a python library that is used to create one-line jokes for programmers. Informally, it can also be referred as a fun python library which is pretty simple to use. Let us see how you can actually use it to perform the required task,

```
Listening.....
Recognizing.....
User said: joke
```

```
Complaining about the lack of smoking shelters, the nicotine addicted Python programmers said there ought to be 'spaces for tabs'.
```

Exit Command

This Command Use for Exit python Program

```
Listening.....
Recognizing.....
User said: exit
[]
```

Python
Task completed
Thanks for giving me your time

GUI Interface



6 . Source code

```
from ast import operator
from asyncio import Task
from ctypes.wintypes import HGDIOBJ
from email.mime import audio
from multiprocessing.context import SpawnContext
from re import search
import sys
import operator
import datetime
import pyttsx3
import speech_recognition as sr
import webbrowser
import pywhatkit
import PyPDF2
import os
import requests
import pyautogui
from bs4 import BeautifulSoup
import wikipedia
import pyjokes
from PyQt5 import QtWidgets, QtCore, QtGui
from PyQt5.QtCore import QTimer, QTime, QDate, Qt
from PyQt5.QtGui import QMovie
from PyQt5.QtCore import*
from PyQt5.QtGui import*
from PyQt5.QtWidgets import*
from PyQt5.uic import loadUiType
from QTGUI import Ui_gui1
import AI

engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
print(voices)
engine.setProperty('voices', voices[1].id)
# text to speech

def Speak(audio):
    engine.say(audio)
    print(audio)
    engine.runAndWait()
```

```

def wish():

    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 can i help you ")

class MainThread(QThread):
    def __init__(self):
        super(MainThread, self).__init__()

    def run(self):
        self.TaskExe()
        # To convert voice into text

    def takecommand(self):
        command = sr.Recognizer()
        with sr.Microphone() as source:
            print("Listening....")
            command.pause_threshold = 1
            command.energy_threshold = 3500
            audio = command.listen(source)

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

            except Exception as Error:
                Speak("Say that again please.....")
                return "none"

        return query.lower()

```

```
#Speak("hello sir i am jarvis")
```

```
def TaskExe(self):
```

```
    def celcius():
```

```
        search = "temperature in jabalpur"
```

```
        url = f"https://www.google.com/search?q={search}"
```

```
        r = requests.get(url)
```

```
        data = BeautifulSoup(r.text, "html.parser")
```

```
        temperature = data.find("div", class_="BNeawe").text
```

```
        Speak(f"The temperature Outside is {temperature} celcius")
```

```
    def pdf_reader():
```

```
        book = open('E:\\Books\\1book.pdf','rb')
```

```
        pdfReader = PyPDF4.PdfFileReader(book) # pip install PyPDF2
```

```
        pages = pdfReader.numPages
```

```
        Speak(f"Total numbers of pages in this book {pages}")
```

```
        Speak("sir please enter the page number i have to read")
```

```
        pg = int(input("Please enter the page number: "))
```

```
        page = pdfReader.getPage(pg)
```

```
        text = page.extractText()
```

```
        Speak(text)
```

```
    def Whatsapp():
```

```
        Speak("Tell Me The Name Of the Person!")
```

```
        Speak("Tell Me The Message!")
```

```
        msg = self.takecommand()
```

```
        Speak("Tell Me The Time Sir!")
```

```
        Speak("Time In Hour!")
```

```
        hour = int(self.takecommand())
```

```
        Speak("Time In Minutes!")
```

```
        min = int(self.takecommand())
```

```
        pywhatkit.sendwhatmsg("+918XXXXXXXXXX", msg, hour, min, 20)
```

```
        Speak("Ok Sir, Sending Whatsapp Message!")
```

```
wish()
```

```
while True:
```

```
    self.query = self.takecommand()
```

```
    if 'who are you' in self.query:
```

```
        Speak("Hello sir,I Am Jarvis. ")
```

```
        Speak("You Personal Ai Assistant! ")
```

```
        Speak("How May I Help You?")
```

```
    elif 'google search' in self.query:
```

```
        query = self.query.replace('google search', '')
```

```
        Speak('searching on google')
```

```
        webbrowser.open("https://www.google.com/search?q="+query+"&rlz=1C1CHZN_enIN949IN949&oq="+query  
                        + "&aqs=chrome..69i57j0i131i433j0i433j0i131i433l3j0i433j0i131i433j0i433j0  
                        .2513j0j15&sourceid=chrome&ie=UTF-8")
```

```
    elif 'where is' in self.query:
```

```
        query = self.query . replace('where is', '')
```

```
        location = query
```

```
        Speak('serching' + ""+location+"" 'on maps')
```

```
        webbrowser.open("https://www.google.co.in/maps/place/" + location + '')
```

```
        Speak("Serching" + location)
```

```
    # switch the window
```

```
    elif "switch the window" in self.query or "switch window" in self.query:
```

```
        Speak("Okay sir, Switching the window")
```

```
        pyautogui.keyDown("alt")
```

```
        pyautogui.press("tab")
```

```
        pyautogui.keyUp("alt")
```

```
    elif 'read book' in self.query or 'read pdf' in self.query:
```

```
        pdf_reader()
```

```
    elif 'current time' in self.query:
```

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

```
        Speak(strTime)
```

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

```
    elif "send message" in self.query:
```

```
        Whatsapp()
```

```
elif 'website' in self.query:
    Speak("Ok Sir Launching.....")
    query = self.query.replace("website", "")
    web1 = self.query.replace("open", "")
    web2 = 'https://www.' + web1 + '.com'
    webbrowser.open(web2)
    Speak("Launched!")

elif 'Launch' in self.query:
    Speak("Tell Me The name of the Website!")
    name = self.takecommand()
    web = 'https://www.' + name + '.com'
    webbrowser.open(web)
    Speak("Done Sir!")

elif 'facebook' in self.query:
    Speak("Ok Sir!")
    webbrowser.open("https://www.facebook.com")
    Speak("Ok Sir....")

elif 'open java point' in self.query:
    Speak("Ok Sir!")
    webbrowser.open("https://www.javatpoint.com")
    Speak("Ok Sir....")

elif 'youtube search' in self.query:
    Speak("Ok Sir, This is What I found for Search!")
    query = self.query.replace("jarvis", "")
    query = self.query.replace("youtube search", "")
    web = 'https://www.youtube.com/results?search_query=' + query
    webbrowser.open(web)
    Speak("Done Sir")

elif 'open stack overflow' in self.query:
    webbrowser.open("www.stackoverflow.com")

elif 'play music' in self.query:
    music = 'E:\\Music'
    songs = os.listdir(music)
    os.startfile(os.path.join(music, songs[0]))

elif 'wikipedia' in self.query:
    Speak("Searching wikipedia.....")
    query = query.replace("wikipedia", "")
    wiki = wikipedia.summary(query, 2)
    print(wiki)
    Speak(f"According to wikipedia : {wiki}")
```

```
elif "temperature" in self.query:
    celcius()

elif 'open cmd' in self.query:
    os.system('start cmd')

elif 'open notepad' in self.query:
    npath = 'C:\\Windows\\system32\\notepad.exe'
    os.startfile(npath)

elif 'close notepad' in self.query:
    Speak("OK sir, closing notepad")
    os.system("taskkill /f /im notepad.exe")

elif 'open Adobe' in self.query:
    npath = 'C:\\Program Files\\Adobe\\Acrobat
    DC\\Acrobat\\Acrobat.exe'
    os.startfile(npath)

elif 'tell me a joke' in self.query:
    joke = pyjokes.get_joke()
    Speak(joke)

elif 'how are you' in self.query:
    Speak("I Am Fine Sir!")
    Speak("What About You?")

elif 'what is my name' in self.query:
    Speak("Your name is Abhay soni")

elif 'what is your name' in self.query:
    Speak("My good name Jarvis Assistant")
```



```

elif "do some calculationns" in self.query :

    try:

        r = sr.Recognizer()

        with sr.Microphone() as source:

            Speak("Say What you want to calculate, example
                : 3 plus 3")

            print("Listening.....")

            r.adjust_for_ambient_noise(source)

            audio = r.listen(source)

        my_string = r.recognize_google(audio)

        print(my_string)


    def get_operator_fn(op):

        return {

            '+': operator.add,  # plus

            '-': operator.sub,  # minus

            'x': operator.mul,  # multiplied by

            'divided': operator.__truediv__,  # divided

        }[op]


    def eval_binary_expr(op1, oper, op2):  # 5 plus 8

        op1, op2 = int(op1), int(op2)

        return get_operator_fn(oper)(op1, op2)

    Speak("your result is")

    Speak(eval_binary_expr(*(my_string.split())))


    except Exception as error:

        Speak("try again")

        return "none"


elif 'you need a break' in self.query:

    Speak("Ok Sir,You Call Me Anytime !")

    break

```

```
startExecution = MainThread()

class Main(QMainWindow):
    def __init__(self):
        super().__init__()
        self.ui = Ui_gui1()
        self.ui.setupUi(self)
        self.ui.pushButton.clicked.connect(self.startTask)
        self.ui.pushButton_2.clicked.connect(self.close)

    def startTask(self):
        self.ui.movie = QtGui.QMovie(
            "C:\\Users\\DELL\\AppData\\Roaming\\Python\\Python39\\Scripts\\Jarvis_Gui (2).gif")
        self.ui.label.setMovie(self.ui.movie)
        self.ui.movie.start()
        self.ui.movie = QtGui.QMovie(
            "C:\\Users\\DELL\\AppData\\Roaming\\Python\\Python39\\Scripts\\Iron_Template_1.gif")
        self.ui.label_2.setMovie(self.ui.movie)
        self.ui.movie.start()
        startExecution.start()

app = QApplication(sys.argv)
AI = Main()
AI.show()
exit(app.exec_())
```

6.1 GUI code

```
from PyQt5 import QtCore, QtGui, QtWidgets

class Ui_gui1(object):
    def setupUi(self, gui1):
        gui1.setObjectName("gui1")
        gui1.resize(1239, 833)
        self.centralwidget = QtWidgets.QWidget(gui1)
        self.centralwidget.setObjectName("centralwidget")
        self.label = QtWidgets.QLabel(self.centralwidget)
        self.label.setGeometry(QtCore.QRect(0, 0, 1241, 841))
        self.label.setText("")
        self.label.setPixmap(QtGui.QPixmap("C:\\Users\\DELL\\AppData\\Roaming\\Python\\Python39\\Scripts\\Jarvis_Gui (2).gif"))
        self.label.setScaledContents(True)
        self.label.setObjectName("label")
        self.label_2 = QtWidgets.QLabel(self.centralwidget)
        self.label_2.setGeometry(QtCore.QRect(620, 360, 361, 231))
        self.label_2.setText("")
        self.label_2.setPixmap(QtGui.QPixmap("C:\\Users\\DELL\\AppData\\Roaming\\Python\\Python39\\Scripts\\Iron_Template_1.gif"))
        self.label_2.setScaledContents(True)
        self.label_2.setObjectName("label_2")
        self.pushButton = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton.setGeometry(QtCore.QRect(1070, 360, 93, 41))
        self.pushButton.setStyleSheet("background-color: rgb(235, 255, 10);\\n"
"font: 75 10pt \\\"MS Shell Dlg 2\\\";")
        self.pushButton.setObjectName("pushButton")
        self.pushButton_2 = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton_2.setGeometry(QtCore.QRect(1070, 430, 93, 41))
        self.pushButton_2.setStyleSheet("background-color: rgb(243, 255, 3);\\n"
"font: 75 10pt \\\"MS Shell Dlg 2\\\";")
        self.pushButton_2.setObjectName("pushButton_2")
        gui1.setCentralWidget(self.centralwidget)

        self.retranslateUi(gui1)
        QtCore.QMetaObject.connectSlotsByName(gui1)

    def retranslateUi(self, gui1):
        _translate = QtCore.QCoreApplication.translate
        gui1.setWindowTitle(_translate("gui1", "MainWindow"))
        self.pushButton.setText(_translate("gui1", "START"))
        self.pushButton_2.setText(_translate("gui1", "EXIT"))

if __name__ == "__main__":
    import sys
    app = QtWidgets.QApplication(sys.argv)
    gui1 = QtWidgets.QMainWindow()
    ui = Ui_gui1()
    ui.setupUi(gui1)
    gui1.show()
    sys.exit(app.exec_())
```



Thank you