VIRTUAL PERSONAL ASSISTANT USINGSPEECH RECOGNITION

A Project Report submitted in partial fulfillment of the requirements for the award of the degree,

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

Y KRISHNAKAANTH REDDY	321910306039
K PRAMOD REDDY	321910306008
C SAI SUMANTH	321910306041
R PRASANNA KUMAR REDDY	321910306059
K VISHNU	321910306038

Under the esteemed guidance of

DR. DAYANANDA LAL . N

Assistant Professor

Department of Computer Science & Engineering

GITAM SCHOOL OF TECHNOLOGY

GANDHI INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Deemed to be University)

Bengaluru Campus.

November 2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GITAM SCHOOL OF TECHNOLOGY

GITAM

(Deemed to be University)

CERTIFICATE

This is to certify that the project report entitled **"Virtual Personal Assistant Using Recognition"** is a bonified record of work carried out by **Y. Krishnakaanth Reddy – 321910306039, K. Pramod Reddy -321910306008, C. Sai Sumanth – 321910306041, R. Prasanna Kumar Reddy – 321910306059, K. Vishnu - 321910306038** submitted in partial fulfillment of requirement for the award of degree of **Bachelors of Technology in Computer Science and Engineering.**

9|11|22

Project Guide.

SIGNATURE OF THE GUIDE

Dr. Dayananda Lal. N

Assistant Professor

y. Vamshi Dhar

Head of the Department.

SIGNATURE OF THE HoD

Dr. Vamsidhar Yendapalli,

Professor

Head of the Department
Computer Science & Engineering
GITAM School of Technology
GITAM University (Bengaluru Campus)
Nagadenehalli - 562 163
Bengaluru Rural Dist. Karnataka

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GITAM SCHOOL OF TECHNOLOGY

GITAM

(Deemed to be University)

DECLARATION

We, hereby declare that the project report entitled **"Virtual Personal Assistant Using SpeechRecognition"** is an original work done in the **Department of Computer Science and Engineering, GITAM School of Technology, GITAM (Deemed to be University)** submitted in partial fulfillment of the requirements for the award of the degree of **B.Tech.** in **Computer Science and Engineering**. The work has not been submitted to any other college or University for the award of any degree.

Date:

Registration No's	**Name**	**Signature**
321910306039	Y. KRISHNAKAANTH REDDY	Y.krishnakaanthreddy
321910306008	K. PRAMOD REDDY	Pramod
321910306041	C. SAI SUMANTH	C.
321910306059	R. PRASANNA KUMAR REDDY	R. Prasanna kumar Re
321910306038	K. VISHNU	

ACKNOWLEDGEMENT

The Satisfaction and Euphoria that accompany the successful completion of any Major Project would be incomplete without mentioning the people who made it possible, whose constant guidance and encouragement crowned our efforts with success. We take this opportunity to express the deepest gratitude and appreciations to all those who held us directly or indirectly towards the successful completion of the Project.

I would like to thank **Dr. Nageshwara Rao T**, Ph.D., Director, GITAM School of Technology for all the facilities provided.

I would like to thank HOD **Dr. Vamsidhar Yendapalli**, Ph.D., Professor, Department of Computer Science and Engineering for his support and encouragement that went a long way in successful completion of this Project.

I consider that as a privilege to express our gratitude and respect to **Dr. Dayananda lal . N**, Assistant professor, department of Computer Science and Engineering for being our internal guide for his integral and incessant support offered to us throughout the course of this project and for constant source of inspiration throughout the Seminar.

I would like to thank our project coordinator **Prof. Viswa Bharathy A M** for his integral and incessant support offered to us throughout the course of this project and for constant source of inspiration throughout the Project.

I would like to thank my parents and my friends for their support and encouragement in the completion of project in time.

Finally, I would like to thank all the teaching and non-teaching staff members of the Computer Science and Engineering Department, for the support in completion of the project in time.

Student Name's	**Registration No.**
Y KRISHNAKAANTH REDDY	321910306039
K PRAMOD REDDY	321910306008
C SAI SUMANTH	321910306041
R PRASANNA KUMAR REDDY	321910306059
K VISHNU	321910306038

TABLE OF CONTENTS

ABSTRACT

Modern society has increased the innovation and technological integration of daily life. We have developed the concept of virtual assistants for laptops in the suggested voice help system. It functions as a useful calendar reminder, note-taker, calculator, and search engine. There isn't a suitable virtual assistant for Microsoft computers when it comes to laptops. We have suggested utilizing Python to create a speech-to-text virtual assistant for the Windows operating system. The proposed work accepts voice input, produces voice output, and displays text. The major goals of voice help are to encourage innovation and provide quick, accurate results. This aid links to the internet to deliver results for the user. has inquired. The fundamental structure of the work is to incorporate as many tasks as is practical and construct it so that our voice commands can do it. Finally, a graphical representation of the outcome is shown.

LIST OF FIGURES

1. INTRODUCTION

Today the advancement of AI systems that are able to create a natural Human-Machine-Interaction (through voice, communication, gestures, facial expressions, etc.) are gaining popularity. One of the mostobserved and popular was the direction of interaction, based on the understanding of the machine by the machine of the natural language processing. It is no longer a human self-learns to communicate with a machine, but a machine self-learns to communicate with a human, exposing his actions, habits, behaviour and trying to become his best personalized asassistant. The work on making and improvising such personalized assistants has been going on for a long period of time. These systems are constantly improving and improving, go beyond personal computers and have already firmly established themselves in differentmobile devices and gadget.

One of the most popular voice assistants are Siri, from Apple, Amazon Echo, which responds to the name of Alexa from Amazon, Cortana from Microsoft, Google Assistant from Google, and therecently appeared intelligent assistant under the name AIVA. This paper presents a brief introduction to the architecture and construction of voice assistants. It provides proposed plan of work and also provides methodology of the work of a voice assistant. It also describes the test results of the voice assistant. The primary goal of this work is to build a local voice assistant that performs the work of human and the daily task that a human needed to do in day-to-day life. It hassome new features like posting comments on the social media websites such as Facebook, Instagram, etc. by just few simple commands. You can also know the weather around and can getthe climate conditions in your local region. It can open and launch web applications and the localstorage of the user computer.

2. LITERATURE SURVEY

Giancarlo Iannizzotto[1] have proposed A vision and speech enabled, customizable, virtual assistant for smart environments which describes about Popular Virtual currently include Amazon Alexa, Apple's Siri. Google Assistant and Microsoft's Cortina and The digital assistant built into windows phone 8.1 and windows 10. It is used for Speech enable Virtual assistants (Often named smart speakers), Offer a wide variety of network-oriented services and In some cases, can connect to smart environments, thus enhancing them with new and effective user interfaces. It has also some limitations like Most virtual assistants are smart enough to understand natural language, but you have to be specific. For instance, if you connect Amazon Echo with the Uber app, Alexa can request a ride, but you have to phrase the command correctly. You have to say, "Alexa, ask Uber to request a ride.

Veton Këpuska[1][2] have proposed Next-Generation of Virtual Personal Assistants (Microsoft Cortana,Apple Siri, Amazon Alexa and Google Home) which described about Google has improved the GoogleAssistant by using the Deep Neural Networks (DNN) method which highlights the main components ofdialogue systems and new deep learning architectures used for these components, Microsoft used the Microsoft Azure Machine Learning Studio with other Azure components to improve the Cortana dialogue system.

Gamal Bohouta[2][2] have proposed Next-Generation of Virtual Personal Assistants (Microsoft Cortana,Apple Siri, Amazon Alexa and Google Home). It is used for automatic extracted features in three modalities: text, audio and video. It has also had some limitations like we need to be very specific in voice and image clarity. If dim light was their image will be hardly recognized.

Jonathan Leung have proposed Goal-Oriented Modelling for Virtual Assistants which describes for GoalNetand MADE as the basis of framework. It is used for Users can interact with the assistant by either typing a message or using spoken language. It has also some limitations. They are limited to specific types of applications such as quizzes an flashcard, The task specifications are written in C++ and must be compiled before usage, making it difficult for end users to personalize their virtual assistants and Python library aimedat providing machine learning methods for developers less familiar with dialogue systems. However, such libraries are targeted at developers and are not suitable for users without programming experience.

3. SURVEY OF TECHNOLOGY

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). Pythonhelps in easy writing and execution of codes. Python can implement the same logic with as much as 1/5thcode ascompared 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 JIA, libraries used are speech recognitionto recognize voice, Pyttsx for text to speech, selenium for web automation etc.

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 tooptimize) than writing everything in a low-level language.

4. SOFTWARE REQUIREMENTS

Software requirements describes the intended purpose, requirements, and nature of software/application /project to be developed.

- **Windows OS.**

  It acts as an interface between the computer and the user. It is designed in such a manner that itoperates, controls, and executes various applications on the computer.

  A 32-bit system can access 232 different memory addresses, i.e. 4 GB of RAM or physicalmemory ideally, it can access more than 4 GB of RAM also.

- **Internet:**

  Internet is required to Access information from Wikipedia and other websites. The Internet makesit possible to quickly find information, listen to music, watch videos, and much, much more

- **Python:**

  It is mandatory to have python 3.6 or above. Since few of the modules or the latest updates fromthe python 3.7

- **Integrated development Environment:**

  Here we use PyCharm Community Version as it is most widely used for python Projects.

4.1. HARDWARE REQUIREMENTS

- **Microphone :**

  Microphones are an essential part of any audio recording system. The microphone picks up the sound and converts it into electrical energy that can then be processed by electronic amplifiers and audio processing systems. To this project one microphone is needed either External or Inbuilt.

- **RAM :**
  Minimum 4 GB is required.

5. PROBLEM STATEMENT

We are all aware about Cortana, Siri, Google Assistant and many other virtual assistants which are designed to aid the tasks of users in Windows, Android and iOS platforms. But to our surprise, there's no such complete virtual assistant available for Core Windows platform consisting of 70% of the users.So, this is actually a major problem for users where there could be Three step process. so, we came up with Developing a virtual personal assistant using Speech recognition through machine learning techniques.

5.1. 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 thandesktop 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 behaviors.

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 plansfor 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. In such a case there can be multiple solutions to paths, and the it should be able to consider user preferences, other active tasks, priorities in order to recommend a particular plan.

5.2. Problem Objectives

- To build Hands free Application which works on Voice Commands
- Used in Personal Desktop and PC

6. PURPOSE AND SCOPE

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 assistants enable 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.

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 convenienceare constantly being optimized, it’s clear that we are moving towards less screen interaction.

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 deviceand integration and if it makes sense for their specific brand.

They will also need to focus on maintaining a user experience that is consistent within the comingyears 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.

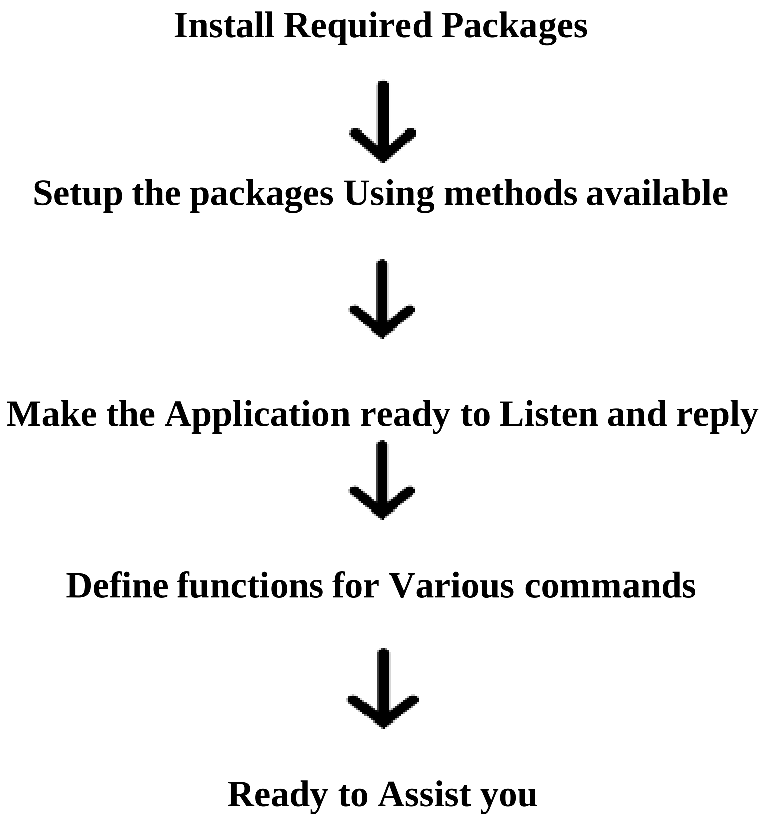
6.1. Applicability

The mass adoption of artificial intelligence in users' everyday lives is also fueling the shift towardsvoice. The number of IoT devices such as smart thermostats and speakers are giving voice assistantsmore utility in a connected user's life. Smart speakers are the number one way we are seeing voicebeing used.

Many industry experts even predict that nearly every application will integrate voice technology insome 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.

7. DESIGNING

7.1. METHODOLOGY



Install Required Packages

Setup the packages Using methods available

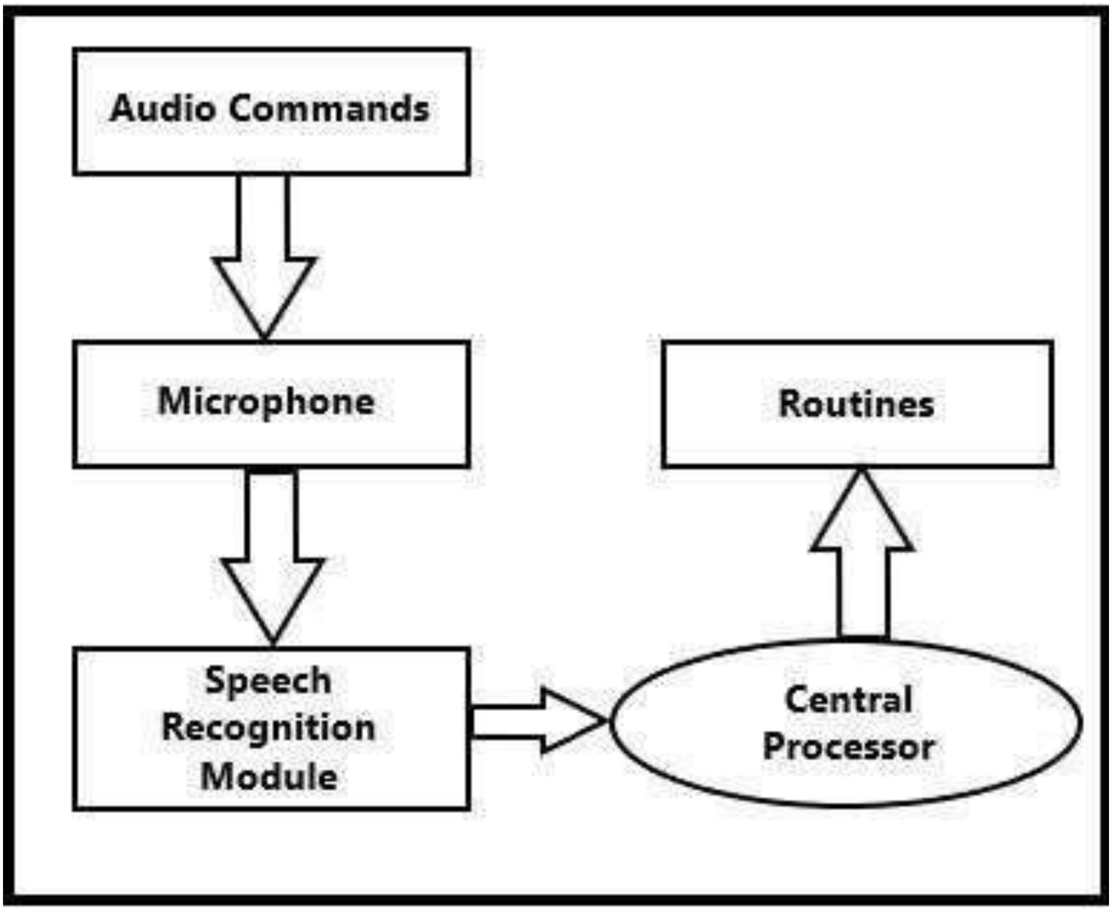
Make the Application ready to Listen and reply

Define functions for Various commands

Ready to Assist you

7.1. Figure Of Methodology

7.2. Basic Work Flow



7.2 Figure Of Basic Workflow

8. IMPLEMENTATION

This is a project which is a Complete personal virtual assistant. In this project, we mainly focus on the package called speech recognition. Which recognizes a user’s speech and takes it as a command, just like as an input. And then, it acts as a voice command to the computer, and it generates the output.

This speech recognition only recognizes the speech. So, to convert the address into the text, we have touse another package called Python text to speech converter version 3.

Firstly, let's define a function with any variable name. Here we have taken it as talking. And here we have used a triangle except which is like. Basically, if you. A voice isn't appropriately caught then exceptwill be working there so. Maybe using try first.

1. Using speech recognition, we access the recognizer, and we assign it to a variable.
2. We access the microphone and then will be accessing the voice from the microphone, basically, inlower case.
3. Next year, we have a robust package which is a hero of the project called Pywhatkit.
4. Using pywhatkit kit, we can do many things like researching or playing YouTube channels or gettingtime or sending WhatsApp messages or many things.
5. Using the power kit, we accessed the speech command and then played the command on YouTube.
6. Similarly. There is a package called Date Time, which tells the current date and time.
7. Here, we have another package called Wikipedia. When we search for something or someone, it accesses the information from the official Wikipedia page and its printouts there. Or it speaks out the information.
8. Here, Lisa will return every uh command at the end of the program to speak out everything by the project.

So, using this project, we can do all the basic steps like getting the information from Google or Wikipedia or getting a time instantly when we do, when we are in a busy schedule. Or you can also play a song or anything on YouTube, or you can also send a WhatsApp message. So, these are like cool features whichwe don't usually find in the inbuilt system.

Virtual assistants, right? And even if you observe Google Assistant only browser, it is permitted only. Towards its boundaries, I mean inside the browser itself. So to maintain the wrapper throughout the system inside the browser and outside the browser, we require a robust and powerful. Assistant like thislikes Lisa. So this does all the things you wanted, and we are really improving much with much more things, and we're looking forward to it.

The main highlight features of this project are:

1. It is entirely a personal virtual assistant.
2. It does all the things you want to do, like for example, it tells you the time, and it gives theinformation from Google and many other things.
3. Suppose you're busy doing something, then if you want to send a WhatsApp message to someone, this virtual decision does nothing.
4. it plays any music or any video without the involvement of the hand or a moment.
5. Few more. Routine tasks as well as daily activities.

So, we are constantly improving this application to our expectations, and we are trying to achieveit. Uh, by focusing more on. Python development and as well as implementing the latest packages well. So, we try to do all the things.

We use additionally few more packages which are not installed by any external package installers, but they are pretty installed for easy usage by any PC.

1. We figured out to find the exact match of the information we need by matching the keyword.
2. We also locate the location in the Google Maps for geographical location
3. By usage of the subprocess, we perform system security functionalities which include Restart etc...
4. We also found a new interesting thing which is wolfram which is very powerful for these functionalities which shows the exact information by providing the acute information.
5. There are also few more features rolling out soon, you know something is always getting updated from our Beloved LEXI

Additional Packages:

- Weather
- Notes
- Recapture 0.
- Json
- Ctypes
- Requests
- Urllib

As we know Python is a suitable language for scriptwriters and developers. Let's write a script forVoice 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.

8.1. Modules Needed

- **Subprocess:** This module is used for getting system subprocess details which are used in variouscommands i.e Shutdown, Sleep, etc. This module comes built-in with Python.

.

- **Pyttsx3:** This module is used for the conversion of text to speech in a program it works offline.To install this module type the below command in the terminal.

- **Tkinter:** This module is used for building GUI and comes inbuilt with Python. This module Comes built in with Python.

- **Wikipedia:** As we all know Wikipedia is a great source of knowledge just like GeeksforGeeks we have used the Wikipedia module to get information from Wikipedia or to perform a Wikipediasearch. To install this module type the below command in the terminal.

- **Speech Recognition:-** Since we're building an Application of voice assistant, one of the most important things in this is that your assistant recognizes your voice (means what youwantto say/ ask). To install this module type the below command in the terminal.

- **Web browser:-** To perform Web Search. This module comes built-in with Python.

- **Ecapture:-** To capture images from your Camera. To install this module type the below command in the terminal.

- **Pyjokes:-** Pyjokes is used for collection Python Jokes over the Internet. To install this moduletype the below command in the terminal.

- **Datetime:-** Date and Time is used to showing Date and Time. This module comes built-intwith Python.

- **Requests:** Requests is used for making GET and POST requests. To install this module typethe below command in the terminal.

- **Urllib:** Urllib package is the URL handling module for python. It is used to fetch URLs

(Uniform Resource Locators). It uses the *urlopen* function and is able to fetch URLs using a variety of different protocols. Urllib is a package that collects several modules for working with URLs, such as:

- urllib.request for opening and reading.
- urllib.parse for parsing URLs
- urllib.error for the exceptions raised
- urllib.robotparser for parsing robot.txt files

Json:

It's pretty easy to load a JSON object in Python. Python has a built-in package called json, which can beused to work with JSON data. It's done by using the JSON module, which provides us with a lot of methods which among loads() and load() methods are gonna help us to read the JSON file.

Deserialization of JSON

The Deserialization of JSON means the conversion of JSON objects into their respective Python objects.The load()/loads() method is used for it. If you have used JSON data from another program or obtained as a string format of JSON, then it can easily be deserialized with load()/loads(), which is usually used to load from string.

Request :

Requests allows you to send HTTP/1.1 requests extremely easily. There's no need to manually add query strings to your URLs, or to form-encode your PUT & POST data — but nowadays, just use theJSON method!

Supported Features & Best–Practices

Requests is ready for the demands of building robust and reliable HTTP–speaking applications, for theneeds of today.

- Keep-Alive & Connection Pooling
- International Domains and URLs
- Sessions with Cookie Persistence
- Browser-style TLS/SSL Verification

- Basic & Digest Authentication
- Familiar dict –like Cookies
- Automatic Content Decompression and Decoding
- Multi-part File Uploads
- SOCKS Proxy Support
- Connection Timeouts
- Streaming Downloads
- Automatic honoring of .netc
- Chucked HTTP requests

9. CODE

Required Packages

```
from ast import While
import traceback
import speech_recognition as sr
import pyttsx3
import pywhatkit
import datetime
import wikipedia
import pyjokes
import subprocess
import webbrowser
import wolframalpha
import json
import requests
import winshell
import ctypes
from urllib import request
import urlopen

listener = sr.Recognizer()
engine = pyttsx3.init()
voices = engine.getProperty('voices')

engine.setProperty('voice', voices[1].id)
engine.say('Hello I am Lexi!, your Virtual personal Assistant')
engine.runAndWait()

def talk(text):
    engine.say(text)
    engine.runAndWait()

def take_command():
    command = "Krishna"
    try:
        with sr.Microphone() as source:
            print('Listening')
            voice = listener.listen(source)
            command = listener.recognize_google(voice)
            print(command)
    except:
        pass
    return command
```

Lexi Operations

```
        def run_lexi():
    command = take_command()
    try :
```

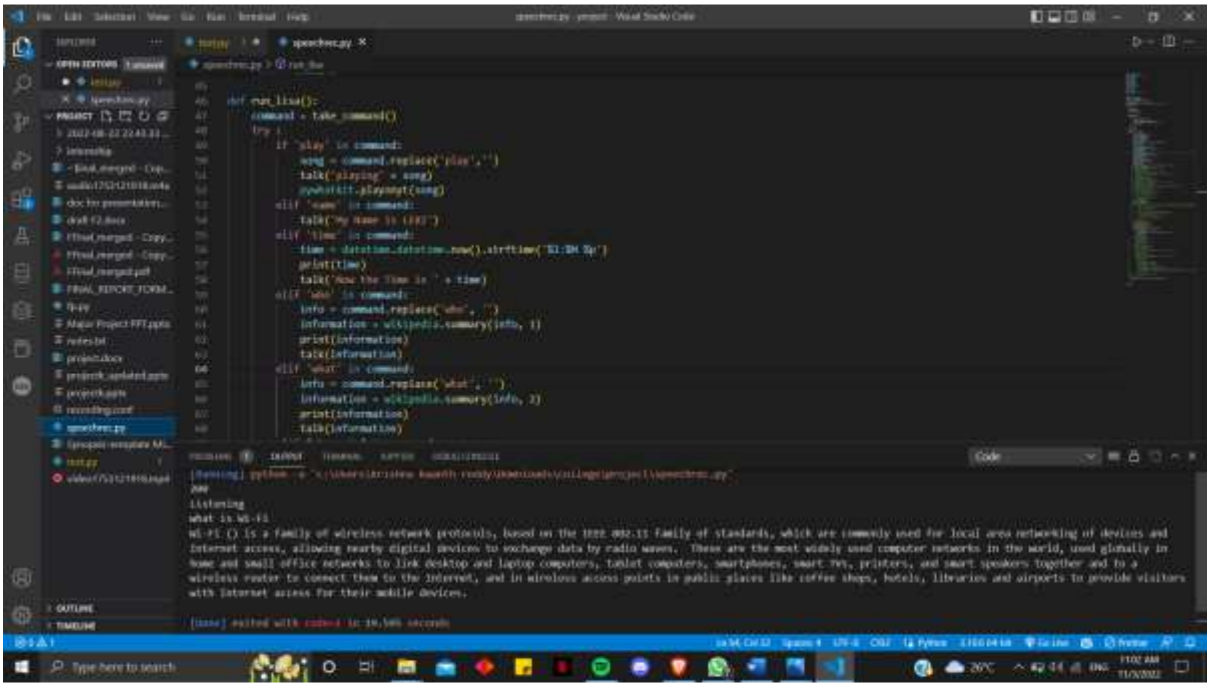
```
if 'play' in command:
    song = command.replace('play',")
    talk('playing' + song)
    pywhatkit.playonyt(song)
elif 'name' in command:
    talk('My Name is LEXI')
elif 'time' in command:
    time = datetime.datetime.now().strftime('%I:%M %p')
    print(time)
    talk('Now the Time is ' + time)
elif 'who' in command:
    info = command.replace('who', ")
    information = wikipedia.summary(info, 1)
    print(information)
    talk(information)
elif 'what' in command:
    info = command.replace('what', ")
    information = wikipedia.summary(info, 2)
    print(information)
    talk(information)
elif "where is" in command:
    query = command.replace("where is", "")
    location = query
    talk("User asked to Locate")
    talk(location)
    webbrowser.open("https://www.google.nl/maps/place/" + location + "")
elif "restart" in command:
    subprocess.call(["shutdown", "/r"])
# elif "hibernate" in command or "sleep" in command:
#     talk("Hibernating")
#     subprocess.call("shutdown / h")
# elif "Shutdown" in command or "sign out" in command:
#     talk("Make sure all the application are closed before sign-out")
#     # time.sleep(5)
#     subprocess.call(["shutdown", "/l"])
# elif 'lock window' in command:
#     talk("locking the device")
#     ctypes.windll.user32.LockWorkStation()
# elif 'shutdown system' in command:
#     talk("Hold On a Sec ! Your system is on its way to shut down")
#     subprocess.call('shutdown / p /f')
# elif 'empty recycle bin' in command:
#     winshell.recycle_bin().empty(confirm=False, show_progress=False, sound=True)
#     talk("Recycle Bin Recycled")
# elif "what is" in command or "who is" in command:
#     # Use the same API key
#     # that we have generated earlier
#     client = wolframalpha.Client("API_ID")
#     res = client.query(command)
    # try:
    #     print(next(res.results).text)
    #     talk(next(res.results).text)
```

```
            # except StopIteration:
            #     print("No results")
        elif 'joke' in command:
            talk(pyjokes.get_joke())
        # elif 'what are the latest news' in command or 'news' in command or 'what is the news' in command:
        #     try:
        #         jsonObj = urlopen('''https://newsapi.org / v1 / articles?source = the-times-of-india&sortBy = top&apiKey
        #         =\\times of India Api key\\''')
        #         data = json.load(jsonObj)
        #         i = 1
        #         talk('here are some top news from the times of india')
        #         print('''=============== TIMES OF INDIA ============''' + '\n')
        #         for item in data['articles']:
        #             print(str(i) + '. ' + item['title'] + '\n')
        #             print(item['description'] + '\n')
        #         talk(str(i) + '. ' + item['title'] + '\n')
        #         i += 1
        #     except Exception as e:
        #         print(str(e))
        elif "write a note" in command:
            talk("What should i write, sir")
            note = take_command()
            file = open('notes.txt', 'w')
            talk("Sir, Should i include date and time")
            snfm = take_command()
            if 'yes' in snfm or 'sure' in snfm:
                strTime = datetime.datetime.now().strftime("% H:% M:% S")
                file.write(strTime)
                file.write(" :- ")
                file.write(note)
            else:
                file.write(note)
    #     elif "show me the note" in command:
    #         talk("Showing Notes")
    #         file = open("jarvis.txt", "r")
    #         print(file.read())
    #         talk(file.read(6))
    #     elif "weather" in command:
    #         # Google Open weather website
    #         # to get API of Open weather
    #         api_key = "Api key"
    #         base_url = "http://api.openweathermap.org / data / 2.5 / weather?"
    #         talk(" City name ")
    #         print("City name : ")
    #         city_name = take_command()
    #         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_pressure = y["pressure"]
    #            current_humidiy = y["humidity"]
    #            z = x["weather"]
    #            weather_description = z[0]["description"]
    #            print(" Temperature (in kelvin unit) = " + str(
    #            current_temperature) + "\n atmospheric pressure (in hPa unit) =" + str(
    #            current_pressure) + "\n humidity (in percentage) = " + str(current_humidiy) + "\n
description = " + str(
    #            weather_description))
    #    else:
    #        # talk('Please say the command again.')
    #        print(command)
    except:
        print(f"Something went wrong: {traceback.format_exc()}")

    # run_lexi()

run_lexi()
```

10.TESTING

10.1. OUTPUT

10.1.1.What is Wi-Fi?

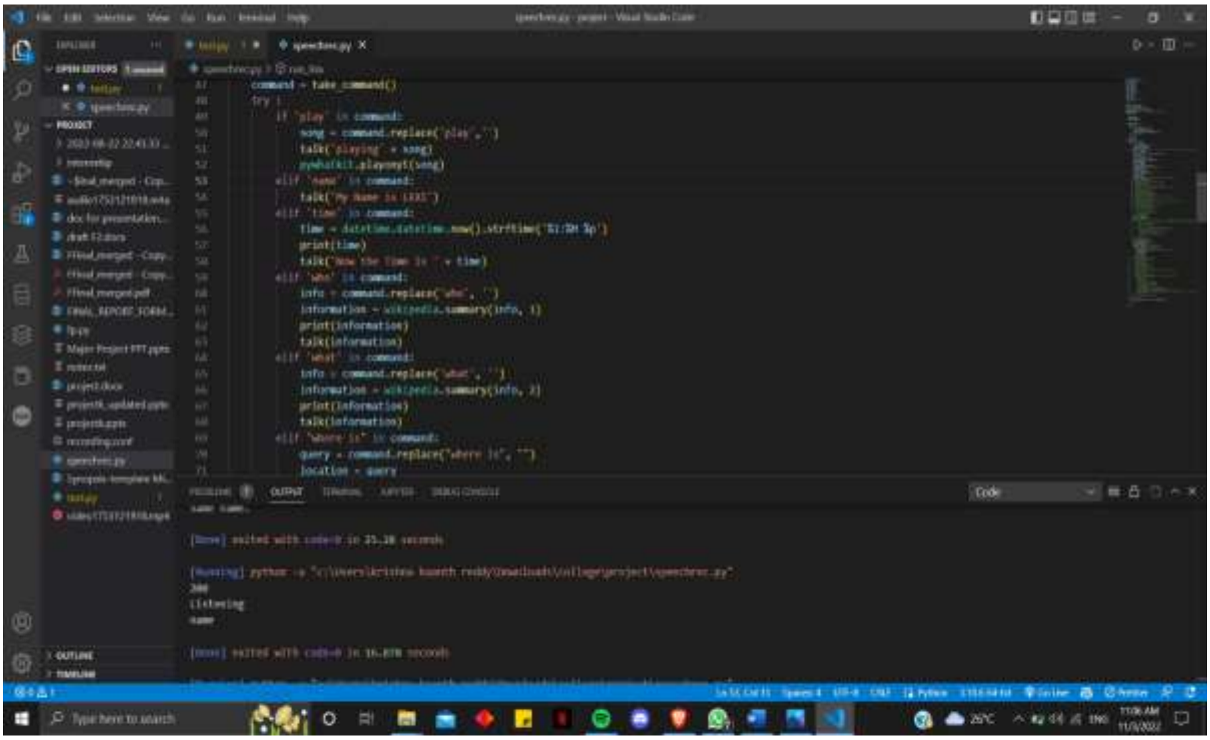


```python
def run_lisa():
    command = take_command()
    try :
        if 'play' in command:
            song = command.replace('play','')
            talk('playing' + song)
            pywhatkit.playonyt(song)
        elif 'name' in command:
            talk('My Name is LISA')
        elif 'time' in command:
            time = datetime.datetime.now().strftime('%I:%M %p')
            print(time)
            talk('Now the Time is ' + time)
        elif 'who' in command:
            info = command.replace('who', '')
            information = wikipedia.summary(info, 1)
            print(information)
            talk(information)
        elif 'what' in command:
            info = command.replace('what', '')
            information = wikipedia.summary(info, 2)
            print(information)
            talk(information)
```

```
[Running] python -u "c:\Users\krishna kaanth reddy\Downloads\college\project\speechrec.py"
200
Listening
what is Wi-Fi
Wi-Fi () is a family of wireless network protocols, based on the IEEE 802.11 family of standards, which are commonly used for local area networking of devices and
Internet access, allowing nearby digital devices to exchange data by radio waves.  These are the most widely used computer networks in the world, used globally in
home and small office networks to link desktop and laptop computers, tablet computers, smartphones, smart TVs, printers, and smart speakers together and to a
wireless router to connect them to the Internet, and in wireless access points in public places like coffee shops, hotels, libraries and airports to provide visitors
with Internet access for their mobile devices.

[Done] exited with code=0 in 19.5[illegible] seconds
```

Fig. 10.1.1

10.1.2 Asking who are you?

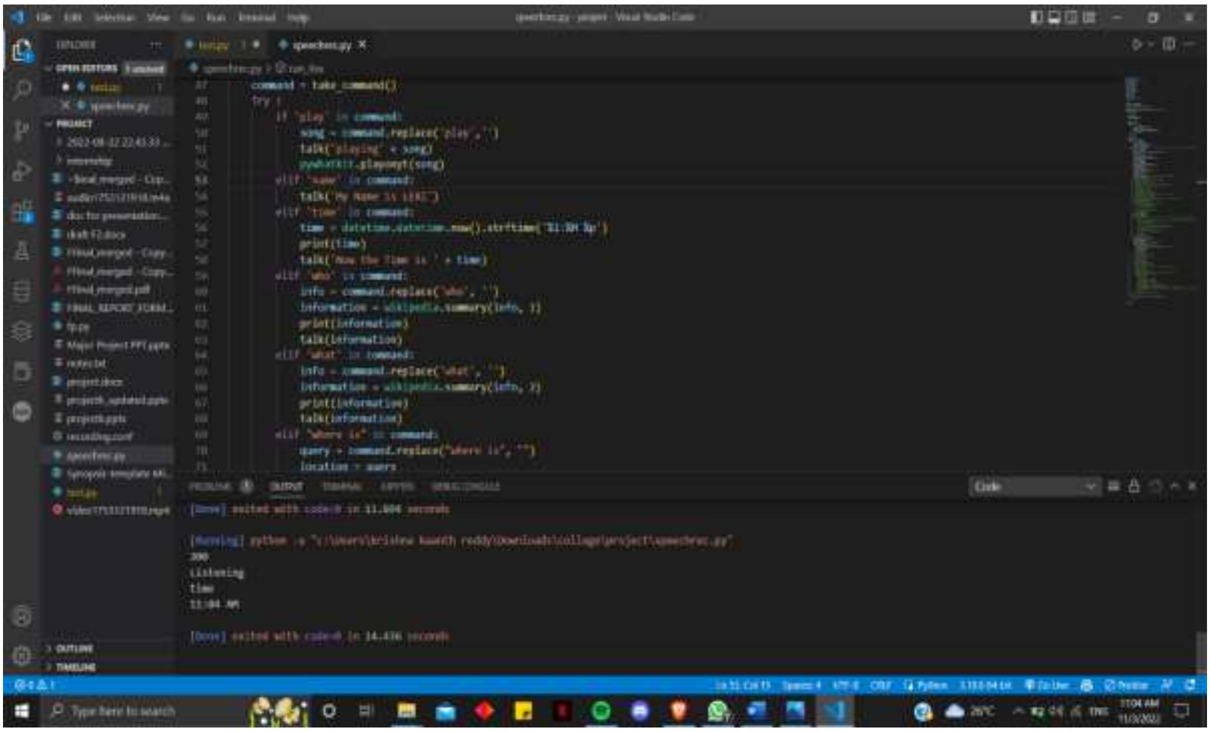


```python
        command = take_command()
        try :
            if 'play' in command:
                song = command.replace('play','')
                talk('playing' + song)
                pywhatkit.playonyt(song)
            elif 'name' in command:
                talk('My Name is LISA')
            elif 'time' in command:
                time = datetime.datetime.now().strftime('%I:%M %p')
                print(time)
                talk('Now the Time is ' + time)
            elif 'who' in command:
                info = command.replace('who', '')
                information = wikipedia.summary(info, 1)
                print(information)
                talk(information)
            elif 'what' in command:
                info = command.replace('what', '')
                information = wikipedia.summary(info, 2)
                print(information)
                talk(information)
            elif 'where is' in command:
                query = command.replace("where is", "")
                location = query
```

```
name name...

[Done] exited with code=0 in 25.28 seconds

[Running] python -u "c:\Users\krishna kaanth reddy\Downloads\college\project\speechrec.py"
200
Listening
name

[Done] exited with code=0 in 16.878 seconds
```

Fig. 10.1.2

10.1.3. Time from the Lexi?

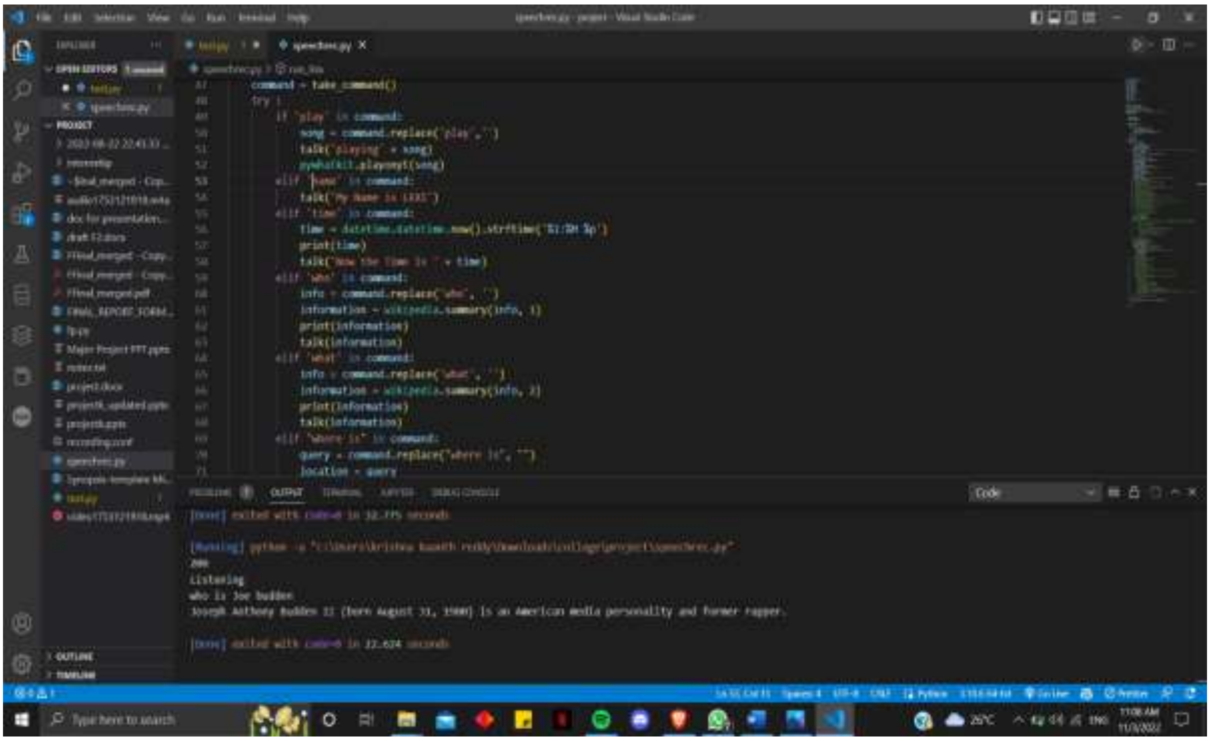


```
command = take_command()
try :
    if 'play' in command:
        song = command.replace('play','')
        talk('playing' + song)
        pywhatkit.playonyt(song)
    elif 'name' in command:
        talk('My Name is LEXI')
    elif 'time' in command:
        time = datetime.datetime.now().strftime('%I:%M %p')
        print(time)
        talk('Now the Time is ' + time)
    elif 'who' in command:
        info = command.replace('who', '')
        information = wikipedia.summary(info, 1)
        print(information)
        talk(information)
    elif 'what' in command:
        info = command.replace('what', '')
        information = wikipedia.summary(info, 2)
        print(information)
        talk(information)
    elif "where is" in command:
        query = command.replace("where is", "")
        location = query
```

```
[Done] exited with code=0 in 11.884 seconds

[Running] python -u "c:\Users\krishna kaanth reddy\Downloads\college\project\speechrec.py"
200
Listening
time
11:04 AM

[Done] exited with code=0 in 14.436 seconds
```

Fig. 10.1.3

10.1.4. Who is joe budden ?

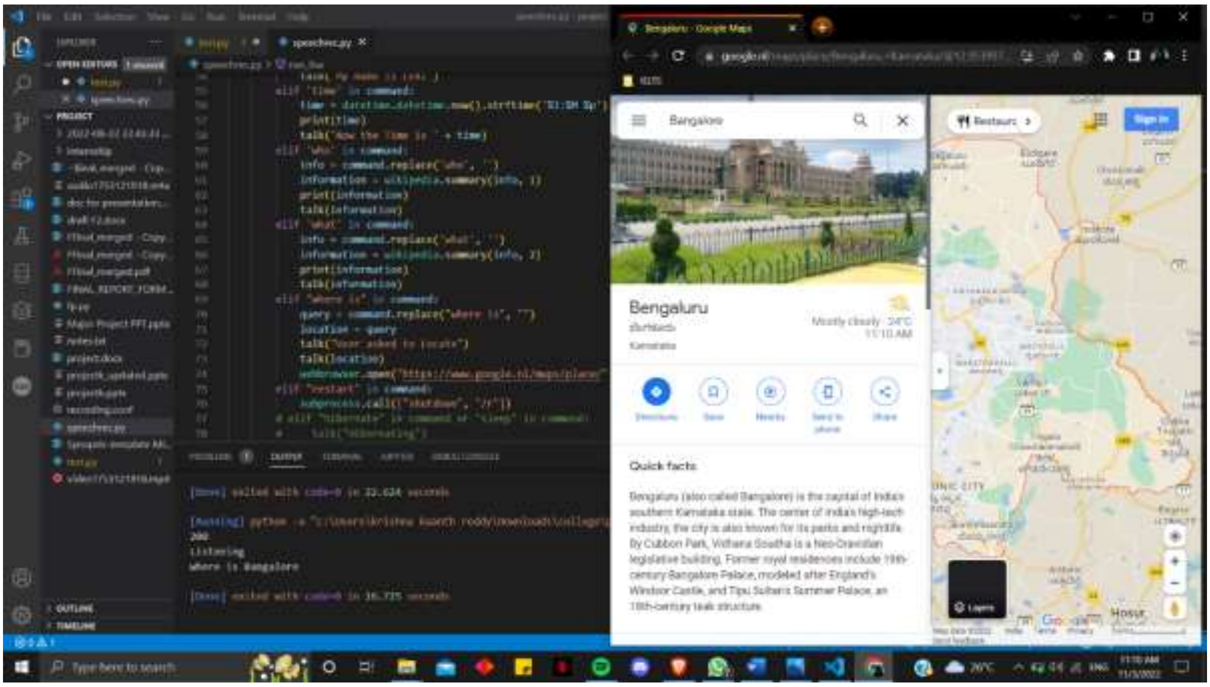


```
command = take_command()
try :
    if 'play' in command:
        song = command.replace('play','')
        talk('playing' + song)
        pywhatkit.playonyt(song)
    elif 'name' in command:
        talk('My Name is LEXI')
    elif 'time' in command:
        time = datetime.datetime.now().strftime('%I:%M %p')
        print(time)
        talk('Now the Time is ' + time)
    elif 'who' in command:
        info = command.replace('who', '')
        information = wikipedia.summary(info, 1)
        print(information)
        talk(information)
    elif 'what' in command:
        info = command.replace('what', '')
        information = wikipedia.summary(info, 2)
        print(information)
        talk(information)
    elif "where is" in command:
        query = command.replace("where is", "")
        location = query
```

```
[Done] exited with code=0 in 32.775 seconds

[Running] python -u "c:\Users\krishna kaanth reddy\Downloads\college\project\speechrec.py"
200
Listening
who is Joe budden
Joseph Anthony Budden II (born August 31, 1980) is an American media personality and former rapper.

[Done] exited with code=0 in 22.624 seconds
```

Fig. 10.1.4

10.1.5. Where is bangalore:



```
        talk('My name is ...')
    elif 'time' in command:
        time = datetime.datetime.now().strftime('%I:%M %p')
        print(time)
        talk('Now the time is ' + time)
    elif 'who' in command:
        info = command.replace('who', '')
        information = wikipedia.summary(info, 1)
        print(information)
        talk(information)
    elif 'what' in command:
        info = command.replace('what', '')
        information = wikipedia.summary(info, 2)
        print(information)
        talk(information)
    elif "where is" in command:
        query = command.replace("where is", "")
        location = query
        talk("User asked to locate")
        talk(location)
        webbrowser.open("https://www.google.nl/maps/place/"
    elif "restart" in command:
        subprocess.call(["shutdown", "/r"])
    # elif "hibernate" in command or "sleep" in command:
    #     talk("Hibernating")
```

```
[Done] exited with code=0 in 22.624 seconds

[Running] python -u "c:\Users\krishna kaanth reddy\Downloads\college
200
Listening
where is Bangalore

[Done] exited with code=0 in 36.735 seconds
```

Bengaluru

Karnataka

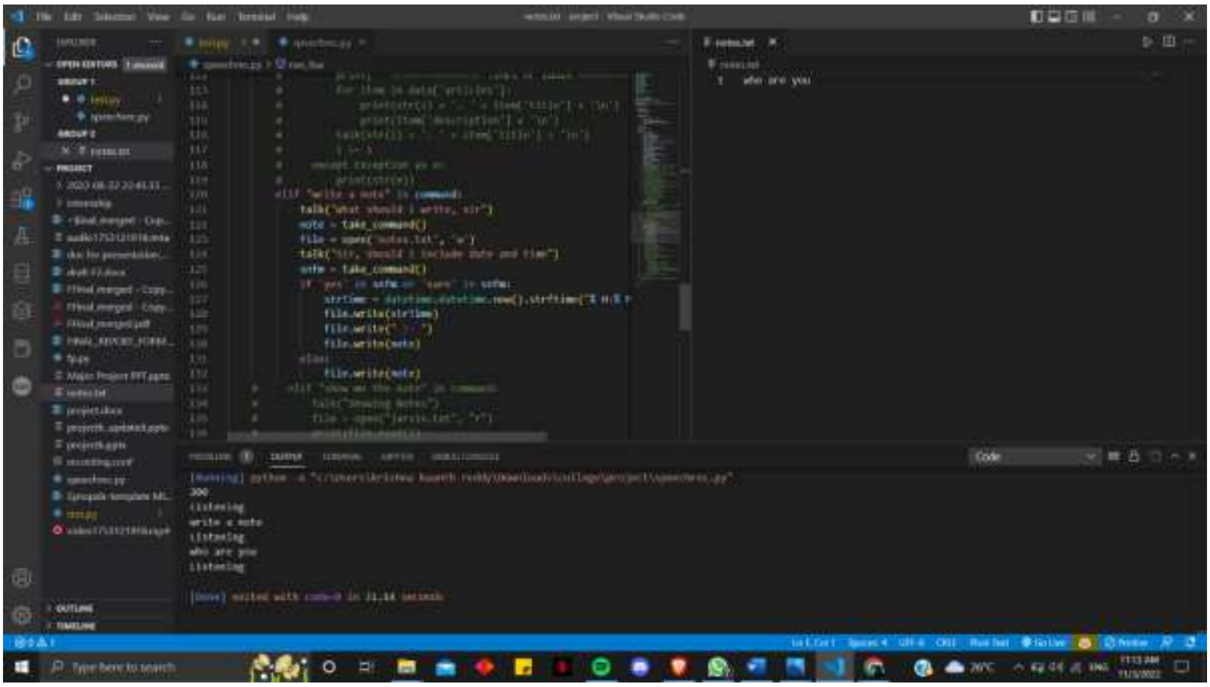
Mostly cloudy · 24°C 11:10 AM

Quick facts

Bengaluru (also called Bangalore) is the capital of India's southern Karnataka state. The center of India's high-tech industry, the city is also known for its parks and nightlife. By Cubbon Park, Vidhana Soudha is a Neo-Dravidian legislative building. Former royal residences include 19th-century Bangalore Palace, modeled after England's Windsor Castle, and Tipu Sultan's Summer Palace, an 18th-century teak structure.

Fig. 10.1.5

10.1.6. Write a note :



```
    #         print('-------------- TOP NEWS ---------------')
    #         for item in data['articles']:
    #             print(str(i) + '. ' + item['title'] + '\n')
    #             print(item['description'] + '\n')
    #         talk(str(i) + '. ' + item['title'] + '\n')
    #         i += 1
    #     except Exception as e:
    #         print(str(e))
    elif "write a note" in command:
        talk("What should i write, sir")
        note = take_command()
        file = open('notes.txt', 'w')
        talk("Sir, should i include date and time")
        snfm = take_command()
        if 'yes' in snfm or 'sure' in snfm:
            strTime = datetime.datetime.now().strftime("% H:%M
            file.write(strTime)
            file.write(" :- ")
            file.write(note)
        else:
            file.write(note)
    #   elif "show me the note" in command:
    #       talk("Showing Notes")
    #       file = open("jarvis.txt", "r")
    #       print(file.read())
```

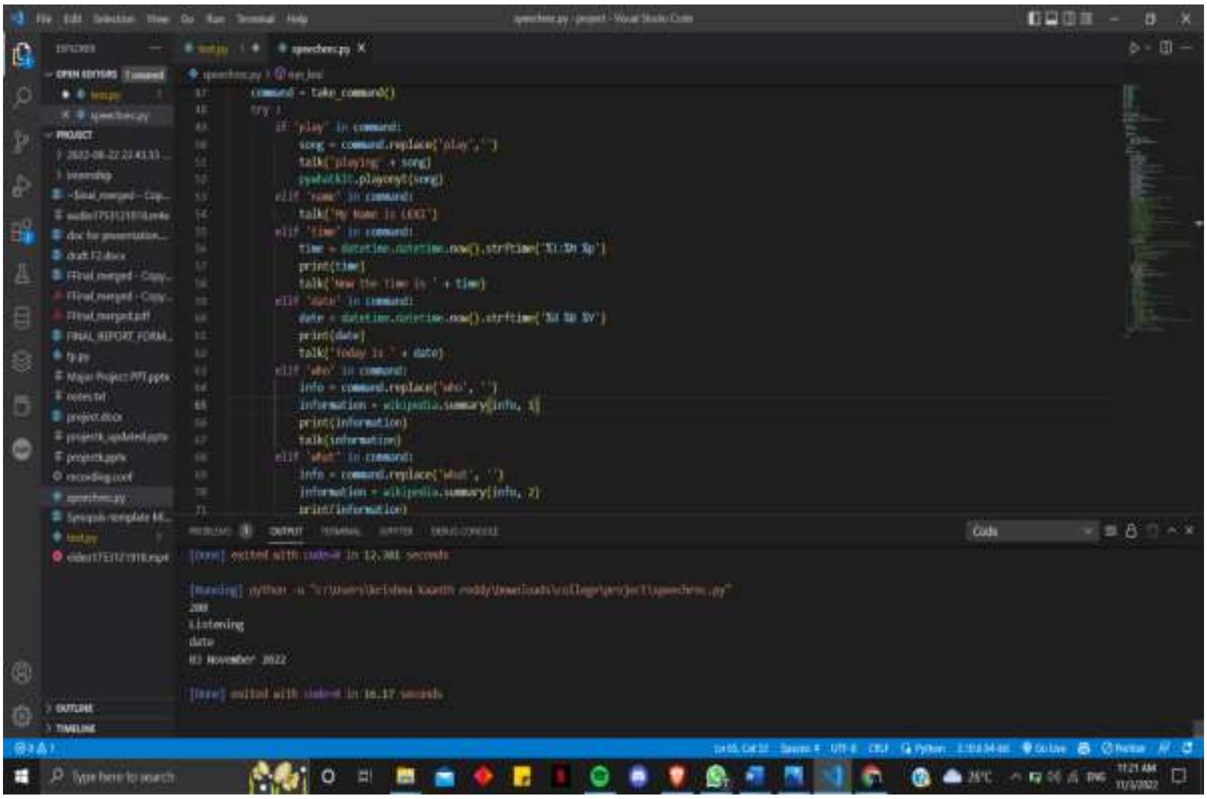
notes.txt

```
1   who are you
```

```
[Running] python -u "c:\Users\krishna kaanth reddy\Downloads\college\project\speechrec.py"
200
Listening
write a note
Listening
who are you
Listening

[Done] exited with code=0 in 31.14 seconds
```

Fig. 10.1.6

10.1.7. Date from lexi:

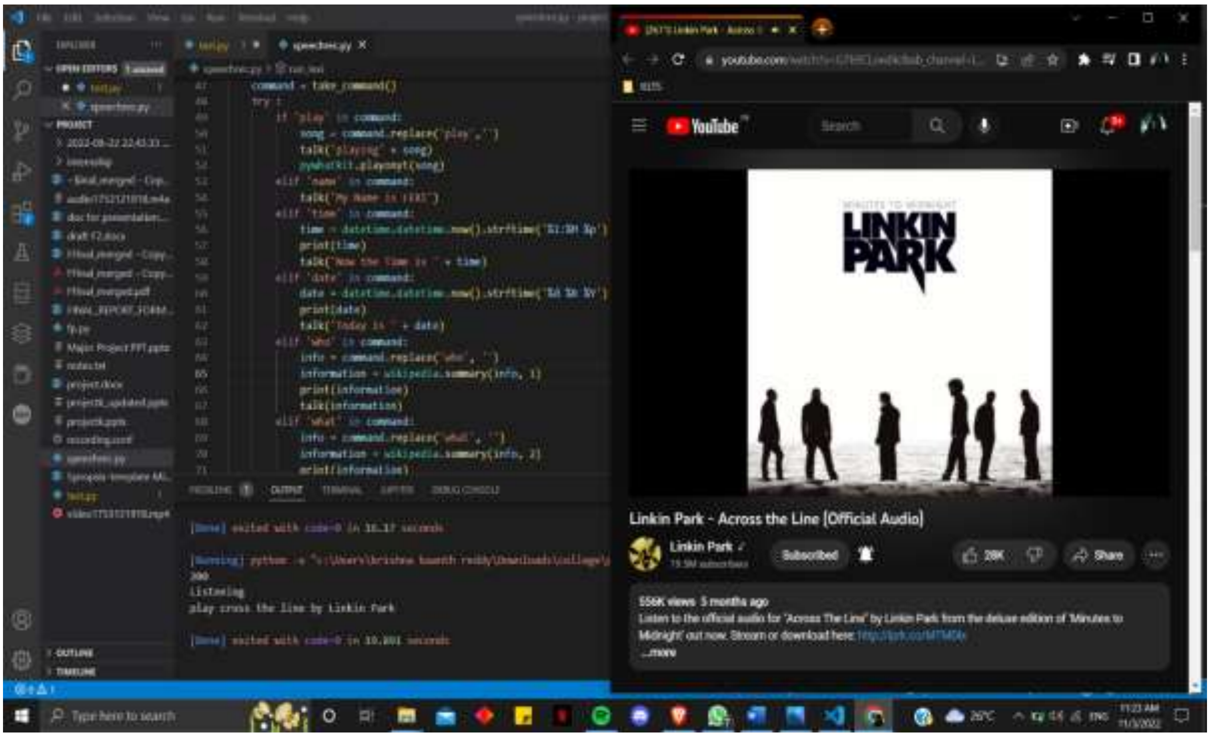


```
        command = take_command()
        try :
            if 'play' in command:
                song = command.replace('play','')
                talk('playing' + song)
                pywhatkit.playonyt(song)
            elif 'name' in command:
                talk('My Name is LEXI')
            elif 'time' in command:
                time = datetime.datetime.now().strftime('%I:%M %p')
                print(time)
                talk('Now the Time is ' + time)
            elif 'date' in command:
                date = datetime.datetime.now().strftime('%d %B %Y')
                print(date)
                talk('Today is ' + date)
            elif 'who' in command:
                info = command.replace('who', '')
                information = wikipedia.summary(info, 1)
                print(information)
                talk(information)
            elif 'what' in command:
                info = command.replace('what', '')
                information = wikipedia.summary(info, 2)
                print(information)
```

```
[Done] exited with code=0 in 12.381 seconds

[Running] python -u "c:\Users\krishna kaanth reddy\Downloads\college\project\speechrec.py"
200
Listening
date
03 November 2022

[Done] exited with code=0 in 16.17 seconds
```

Fig. 10.1.7

10.1.8. Play a song:



```
[Done] exited with code=0 in 16.17 seconds

[Running] python -u "c:\Users\krishna kaanth reddy\Downloads\college\
200
Listening
play cross the line by Linkin Park

[Done] exited with code=0 in 10.801 seconds
```

Linkin Park - Across the Line (Official Audio)

Linkin Park

556K views 5 months ago

Listen to the official audio for "Across The Line" by Linkin Park from the deluxe edition of 'Minutes to Midnight' out now. Stream or download here: http://lnk.to/MTMDlx

...more

Fig. 10.1.8

11. CONCLUSION

Through this voice assistant, we have automated various services using a single line command. It easesmost of the tasks of the user like searching the web, retrieving weather forecast details, vocabulary helpand medical related queries. Speech recognition in future will evolve the way people do business aroundthe web and will ultimately integrate world class e-business. Speech recognition & voice XML clearly represent the next generation of the web.

The future plans include integrating our software with mobile to provide a synchronized experience between the two connected devices. Further, in the long run, it is planned to featureauto deployment supporting elastic beanstalk, backup files, and all operations which a generalServer Administrator does. The functionality would be seamless enough to replace the ServerAdministrator with our software

REFERENCES

[1] X. Lei, G. Tu, A. X. Liu, C. Li, and T. Xie, “The insecurity of home digital voice assistants - amazonalexa as a case study,” CoRR, vol. abs/1712.03327, 2017.

[2] J. Gratch, N. Wang, J. Gerten, E. Fast, and R. Duffy, “Creating rapport with virtual agents,” in Intelligent Virtual Agents (C. Pelachaud, J.-C. Martin, E. Andre, G. Chollet, K. Karpouzis, and D. Pel ´e, eds.), (Berlin, ´ Heidelberg), pp. 125–138, Springer Berlin Heidelberg, 2007.

[3] J. C. Lester, S. A. Converse, S. E. Kahler, S. T. Barlow, B. A. Stone, and R. S. Bhogal, “The personaeffect: Affective impact of animated pedagogical agents,” in Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, CHI ’97, (New York, NY, USA), pp. 359–366, ACM, 1997.

[4] Embodied Conversational Agents. Cambridge, MA, USA: MIT Press, 2000.

[5] E. Andre and C. Pelachaud, ´ Interacting with Embodied Conversational Agents, pp. 123–149. Boston, MA: Springer US, 2010.

[6] B. Weiss, I. Wechsung, C. Kuhnel, and S. M ¨ oller, “Evaluating embodied ¨ conversational agents in multimodal interfaces,” Computational Cognitive Science, vol. 1, p. 6, Aug 2015.

[7] Y. Matsuyama, A. Bhardwaj, R. Zhao, O. Romeo, S. Akoju, and J. Cassell, “Socially-aware animatedintelligent personal assistant agent,” in Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 224–227, Association for Computational Linguistics, 2016.

[8] M. Schroeder, E. Bevacqua, R. Cowie, F. Eyben, H. Gunes, D. Heylen, M. ter Maat, G. McKeown,S. Pammi, M. Pantic, C. Pelachaud, B. Schuller, E. de Sevin, M. Valstar, and M. Wllmer, “Building autonomous sensitive artificial listeners,” IEEE transactions on affective computing, vol. 3, pp.