

Assistive Voice Technology using A.I.

A PROJECT REPORT

Submitted By

AAKASH SHARMA
(RA1811033010073)

Under the guidance of

DR. KOTTILINGAM K
Associate Professor, Department of Computational Intelligence

In partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE ENGINEERING
with specialization in (SOFTWARE ENGINEERING)



DEPARTMENT OF COMPUTATIONAL INTELLIGENCE COLLEGE
OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

MAY 2022

Annexure II

Department of Computational Intelligence
SRM Institute of Science Technology
Own Work* Declaration Form

This sheet must be filled in (each box ticked to show that the condition has been met). It must be signed and dated along with your student registration number and included with all assignments you submit – work will not be marked unless this is done.

To be completed by the student for all assessments

Degree/ Course : Bachelor of Technology in Computer Science Eng.

Student Name : AAKASH SHARMA

Registration Number : RA1811033010073

Title of Work : AI Voice Assistant

I hereby certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism**, as listed in the University Website, Regulations, and the Education Committee guidelines.

I confirm that all the work contained in this assessment is my own except where indicated, and that I have met the following conditions:

- Clearly referenced / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc)
- Given the sources of all pictures, data etc. that are not my own
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course handbook / University website

I understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:
<p>I am aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except where indicated by referring, and that I have followed the good academic practices noted above.</p> <p>Aakash Sharma RA1811033010073</p>
<p>If you are working in a group, please write your registration numbers and sign with the date for every student in your group.</p>

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR–603 203**

BONAFIDE CERTIFICATE

Certified that 18CSP109L project report [18CSP110L semester internship report] titled “**Assistive Voice Technology using A.I.**” is the bonafide work of “**AAKASH SHARMA [RA1811033010073]**” who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. Kottilingam K

SUPERVISOR

Associate Professor
Computational Intelligence

SIGNATURE

Dr. R. Annie Uthra

**HEAD OF THE DEPARTMENT
COMPUTATIONAL INTELLIGENCE**

Signature of the Internal Examiner

Signature of the External Examiner

ACKNOWLEDGEMENT

We would like to express our gratitude to all those who gave us the possibility to complete this project and report. We would like to show our gratitude to **Dr. R. Annie Uthra**, Head of the department, Computational Intelligence, School of Computing, for giving us the opportunity to do the final year project / Internship. A special thanks to our final year project coordinators **Mrs. Jayasudha J.** whose help, stimulating suggestions and encouragement helped us in coordinating the project / internship and writing the report and my guide **Dr. Kottilingam K** who propelled the idea of this project and her full effort in guiding the team in achieving the goal as well as her encouragement to maintain our progress on track.

AAKASH SHARMA

ABSTRACT

Python is a relatively new programming language, creating a Voice Assistant script in Python is a cinch. You have complete control over how the assistant responds to your commands. With voice recognition, you can turn anything you're saying into text. Voice-activated assistants like Alexa, Siri, and others often use this technique. Speech to text conversion is made possible in Python using the SpeechRecognition API. Creating my own personal helper was a fascinating challenge. With the use of a single voice command, you can now send emails, search the internet, play music, and launch your favourite IDE without ever having to open a browser. The present state of technology means that it is capable of doing any work as successfully as we are, if not better. I discovered that the notion of AI in every sector reduces human work and saves time via the creation of this project. Functionalities of this project include:

This project's features include: 1.) Command prompt, your chosen IDE or notepad etc. may all be opened with this app. 2.) It has a built-in stereo. 3.) You may use it to do Wikipedia searches. 4.) It may be used to access popular websites like Google, YouTube, and others in a web browser. 5.) It can predict the weather. 6.) Basic discussion is possible. 7.) Emails may be sent with this device. Now, the fundamental question is, "How does an AI work?" It's as if the virtual assistant I've built isn't an A.I. at all, but rather the result of a bunch of statements.

However, the primary goal of artificial intelligence (AI) computers is to execute human activities as efficiently as or more effectively than humans. It is a reality that my virtual assistant is not a very excellent example of A.I., but it is an A.I.

ANNEXURE IV

TABLE OF CONTENTS

ABSTRACT	i
1 INTRODUCTION	7
1.1 WORK CONNECTED TO THIS SUBJECT	9
1.2 PROPOSED PLAN OF WORK	10
1.3 AI VOICE ASSISTANCE	11
2 METHODOLOGY	11
2.1 PRESENT SYSTEM	13
2.2 PROPOSED SYSTEM	13
3 SYSTEM ARCHITECTURE AND DESIGN	15
3.1 ARCHITECTING A COMPUTER SYSTEM	18
4 METHODOLOGY	19
4.1 SPECIFICATIONS OF THE PROGRAM	19
5 IMPLEMENTATION WORK DETAIL	24
5.1 APPLICATION IN THE REAL WORLD	24
5.2 DATA IMPLEMENTATION AND PROGRAM EXECUTION	25
6 SOURCE CODE AND COMMANDS	26
7 INPUT/OUTPUT SCREENSHOT	31
8 CONCLUSION	37
REFERENCES	38

LIST OF FIGURES

1.1	Text to speech technology	4
1.2	Process of working	5
2.1	Backend working data flow	8
2.2	Input output text to speech	10
3.1	Signals data flows	11
3.2	Tutor data flow	20
3.3	Use case diagram	21
4.1	Implementation of libraries	22
4.2	Command prompt window	23
4.3	Voice assistance word	24
4.4	Output of command	25
5.1	All the libraries	28
5.2	Code Implementation	29
6.1	Input/Output of the Code	33

LIST OF TABLES

1.1	Voice and Brain technology	12
1.2	Computer Vision table	13
3.2	Data flow Table	20
4.1	Library Tables	28

ABBREVIATIONS

- **IDE :** Integrated Development Environment.
- **ANN :** Artificial Neural Network.
- **CNN :** Convolutional Neural Network.
- **AI :** Artificial Intelligence.
- **IoT :** Internet of Thing.
- **ASR :** Automatic Speech Recognition.
- **HAM :** Human Against Machine.
- **API :** Application programming interface.
- **MNIST :** Modified National Institute of Standards and Technology.
- **UI :** User Interface.
- **VSCode :** Visual Studio Code.
- **IVR :** Integrated Voice Recognition.
- **TTS :** Teach-to-Speech.
- **NLU :** Natural Language Understanding.
- **NER :** Named Entity Recognition.
- **VA :** Voice Assistant.
- **OS :** Operating System.

Chapter 1

INTRODUCTION

When AI is combined with other machines, it demonstrates that it is capable of thinking like a human. The goal is to create a computer system that necessitates the use of a human user interface. Because Python is a relatively new programming language, creating a Voice Assistant script in Python is a cinch. You have complete control over how the assistant responds to your commands. Alexa, Siri, and other voice-activated assistants use speech recognition technology. An API called Speech Recognition exists in Python, and it enables us to turn spoken words into written ones. Creating my own personal helper was a fascinating challenge. With the use of a single voice command, you can now send emails, search the internet, play music, and launch your favourite IDE without ever having to open a browser. The present state of technology means that it is capable of doing any work as successfully as we are, if not better. I discovered that the notion of AI in every sector reduces human work and saves time via the creation of this project

There are a number of features that make this app useful, such as the ability to send emails, the ability to open command prompts (such as your preferred IDE or notepad), and the ability to play music, as well as the ability to run Wikipedia searches for you. Basic discussion is possible. There has been research on the similarities and differences between various voice assistant devices and services. When AI is combined with other machines, it demonstrates that it is capable of thinking like a human. The goal is to create a computer system that necessitates the use of a human user interface. Because Python is a relatively new programming language, creating a Voice Assistant script in Python is a cinch. For IoT devices that don't have a touchscreen, a voice-activated interface is needed (Metz, 2014). You have complete control over how the assistant responds to your commands. Alexa, Siri, and other voice-activated assistants use speech recognition technology. An API called Speech Recognition exists in Python, and it enables us to turn spoken words into written ones. Creating my own personal helper was a fascinating challenge. With the use of a single voice command, you can now send emails, search the internet, play music, and launch your favourite IDE without ever having to open a browser. The present state of technology means that it is capable of doing any work as successfully as we are. Canalys (2018) estimates that there will be 225 million smart speakers in use by 2020 and 320 million by 2022. if not better. I discovered that the notion of AI in every sector reduces human work and saves time via the creation of this project

In the twenty-first century, whether it's your house or your automobile, everything is moving toward automation. Over the last several years, technology has seen an incredible shift or growth. In today's environment, you can communicate with your computer. As a human, how do you engage with a computer? Obviously, you'll have to provide some input, but what if you don't type anything at all, but instead use your own voice? With more than 70 percent of all intelligent voice assistant-enabled devices running the Alexa platform, it is the dominating market leader (Griswold, 2018). Is it possible to have the computer communicate with you in a similar way as a personal assistant? Isn't it possible that the computer is not only

providing you with the best results, but also suggesting a better alternative? Using voice instructions to control a machine is a revolutionary new method of human-system interaction. We need to utilise a voice to text API in order to interpret the input[6]. Companies like Google and Amazon are attempting to make this universally available. Aren't you awestruck that you can create reminders simply by stating "Remind me to..." Set a timer for.. to wake me up, or set an alarm for.. to wake me up. A system that can be installed anywhere in the neighbourhood and can be asked to assist you accomplish anything for you only by speaking with the device has been developed to recognise the relevance of this issue. Adding to this, you may link two of these gadgets together through Wi-Fi in the future to enable communication between them. You may use this gadget on a daily basis, and it can help you perform more effectively by continually reminding you of tasks and providing you with updates and notifications. What's the point of it? Your voice, rather than an enter key, is becoming the ideal input method

Because the voice assistant is powered by Artificial Intelligence, the results it provides are very accurate and efficient. Using an assistant reduces the amount of human work and time required to accomplish a job; they eliminate the need for typing entirely and act as an additional person to whom we may converse and delegate tasks. In terms of effectiveness and (López et al., 2017; Kpuska and Bohouta, 2018). Science and the educational sector are also looking at whether these new gadgets can aid education, as they do with every new groundbreaking technology. In the past, personal computers and tablet computers have had comparable issues (Algoufi, 2016; Gikas and Grant, 2013; Herrington and Herrington, 2007).

I used Visual Studio Code to construct this project, and all of the py files were produced in VSCode. The following modules and libraries were also utilised in my project: PyAudio, pyttsx3, Wikipedia, Smtplib, pyAudio, OS, Webbrowser, and so on. Because it adds visual appeal and a unique design element to the interaction with JARVIS, I developed a live GUI interface

In today's world, virtual assistants are really helpful. It facilitates human existence in ways similar to Using just vocal commands, run a computer or laptop. Using a virtual assistant saves time. We are able to devote more time to other projects thanks to the help of a virtual assistant. Assistive technology A virtual assistant is often a cloud-based application that works with devices connected to the internet. is the ability to contract for just the services they need. As a means of developing a virtual assistant Python will take over your PC. Task-oriented virtual assistants are the most common kind of virtual assistant. The use of a remote assistance understanding of and capacity to follow instructions. In a three-week study, Beirl et al. (2019) examined how Alexa was used in the household. Studying how families use Alexa's new talents in music, storytelling, and gaming was the goal of the research..A virtual assistant is a computer programme that is able to recognise and respond to user requests. Clients' instructions are followed verbally and in writing. To put it simply, they're ability to understand and react to human speech via the use of artificial voice syntheses. A variety of voices are available. assistants on the market, such as Apple TV's Siri and Pixel phones' Google Assistant An Alexa-powered smart speaker built on a Raspberry Pi and Microsoft Windows. There are ten Cortanas in the world. Our own virtual assistant was produced in the same way as all other virtual assistants. windows. This project would benefit greatly from the application of artificial intelligence technologies. Python may also be used as a the language,

since python has a large number of well-known libraries. A microphone is required to run this programme.

In this research, Savago et al. look at the usage of voice assistants by seniors (age 65 and above) (2019). In order to better understand the usage of digital technology by older persons, the authors stress the need of doing more study. Additionally, Kowalski et al. (2019) studied older individuals' usage of voice-activated devices. The research included seven elderly persons. Voice-activated input and output devices. Several diverse technologies, such as speech recognition, voice analysis, and language processing, are used in this procedure. Natural language processing is used by virtual assistants to translate text and voice input from users into actionable instructions. Audio signals are translated to digital signals when a user instructs their personal virtual assistant to do a job.

1.1 WORK CONNECTED TO THIS SUBJECT

Because of the unique techniques and approaches each company-developer uses to create the intelligent assistant, the end result is affected. Another can accomplish things more properly and without the need for extra explanations or corrections, while others can perform a more limited range of duties but do so most accurately and to the user's satisfaction. No matter how hard you try, you'll never be able to find a universal helper who can do everything. An assistant's personality is totally determined by how much time and care the creator has devoted to each feature. Due to the fact that all systems are based on machine learning and employ vast quantities of data acquired from different sources and then trained on them, the source of this data plays a vital part in their production. The kind of assistance that emerges depends on the quantity of data gathered from various sources. Despite the wide variety of learning methodologies, algorithms, and techniques, the basic building blocks of these systems remain essentially the same across the industry. Figure 1 depicts the emerging technologies.

using a human's natural language to construct intelligent systems for human-computer interaction. Voice activation, automated speech recognition, Teach-To-Speech, voice biometrics, dialogue manager, natural language comprehension, and named entity identification are among the most common technologies used in the industry.

A system that can accurately interpret and answer questions from users is our ultimate aim for Voice Assistant jobs. Despite the fact that WER is the industry standard for assessing ASR errors, creating a high-performing VA system demands an accurate comprehension of the user's intent. Using domain and intent recognition, we examine VA task performance here. After an ASR processes the speech to produce text, the text is passed into a domain classifier (e.g., phone call, online search) in order to determine the user's purpose and the kind of application (phone call, web search, etc.). Finally, intent is utilised to determine the user's desired outcome. If a user's purpose is incorrect, the VA is likely to reply incorrectly or not at all, resulting in a bad user experience.

1.2 PROPOSED PLAN OF WORK

Analyzing the user's microphone instructions was the first step in the process. Everything from retrieving data to managing a computer's internal files falls under this category. Reading and testing the cases from the literature cited above, this is an empirical qualitative investigation. Programming is done according to books and internet resources, with the intention of discovering best practises and a deeper grasp of Voice Assistant.

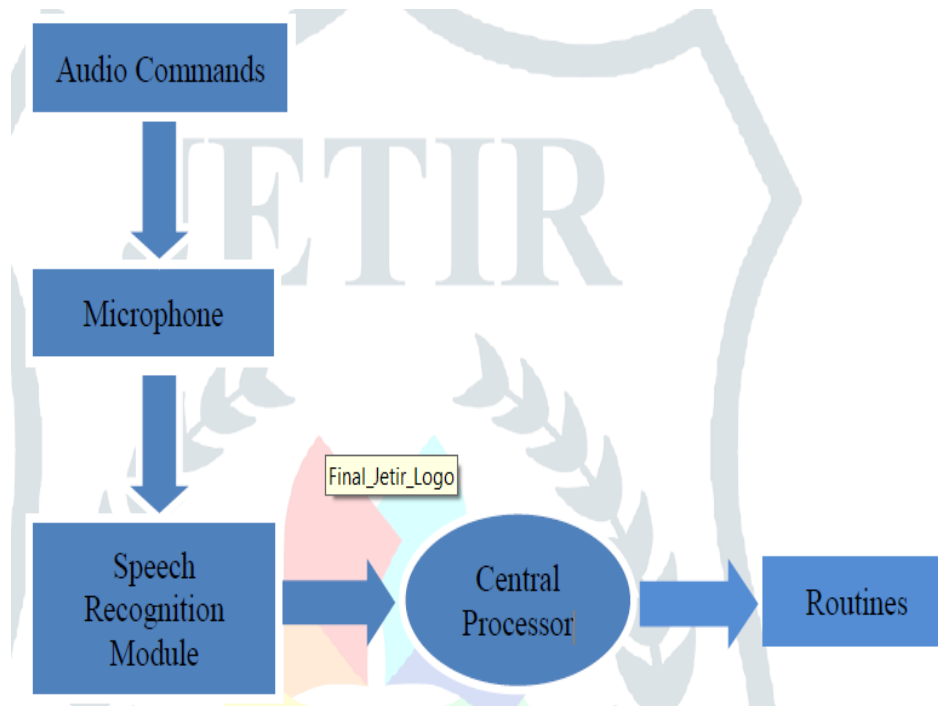


Figure. 1.1. Shows the process of working

A limited subset of often asked questions (e.g., "what's the weather like today?") should be prioritised by an ASR system while performing voice assistant activities. This is often done by raising the weight of the language model (or reducing the AM by proxy) such that the probability of recognising a frequent command or phrase is high. If the LM weight is too high, the model will only produce sentences it has already seen. Dysfluencies such as sound repeats or prolongations tend to generate mistakes in the AM component of the system, so we may exploit this trade-off. There may be fewer repetitions with a lower level of resistance (LM) than with a higher level of resistance (LM). While the default ASR decoder correctly identified "what is my brother's add add add address" when applied to a dysfluent speech sample, increasing the LM weight improved its accuracy to "what is my brother's address." Repetitive "EY d" sound repeats, which resulted in the half word "add," were eliminated.

1.2 AI VOICE ASSISTANCE

AI role as a personal assistant, the end-user is assisted with everyday tasks such as general human conversation, searching queries in various search engines like Google and Bing, retrieving videos, live weather conditions, word meanings, searching for medicine details, health recommendations based on symptoms, and reminding the user of the scheduled events and tasks. Machine learning is used to determine the best course of action based on the user's comments and requests.

Presently, Jarvis is being developed as an automation tool and virtual assistant. Among the Various roles played by Jarvis are:

1. Search Engine with voice interactions
2. Medical diagnosis with Medicine aid.
3. Reminder and To-Do application.
4. Vocabulary App to show meanings and correct spelling errors.
5. Weather Forecasting Application.

Everything remains the same, even for a developer working on Linux who relies on running queries. By allowing online searches for our voice assistant, we've fulfilled a critical need for internet users. Node JS and the Selenium framework have been used in this example to both extract and show the results from the web. Jarvis shows search results from a variety of search engines, including Google, Bing, and Yahoo, by scraping the entered searches.

As a primary source of entertainment, videos have remained a top priority for virtual assistants. These videos have a dual purpose: entertainment and education, since the majority of educational and scientific endeavours now take place on YouTube. This facilitates a more hands-on, outside-the-classroom learning experience.

The core Golang service manages a subprocess module that Jarvis uses to implement the functionality. The Selenium WebDriver and the YouTube search query are scraped by this service in a Node JS subprocess.

It is easier to send emails from Jarvis than it would be if you had to open the email account in question. Jarvis eliminates the necessity for switching to another tab to do a common daily job. Emails may be sent to the recipient of the user's choice. Once he selects Send mail, a form will appear. Click the Send Email button after filling out the form.

Chapter 2

METHODOLOGY

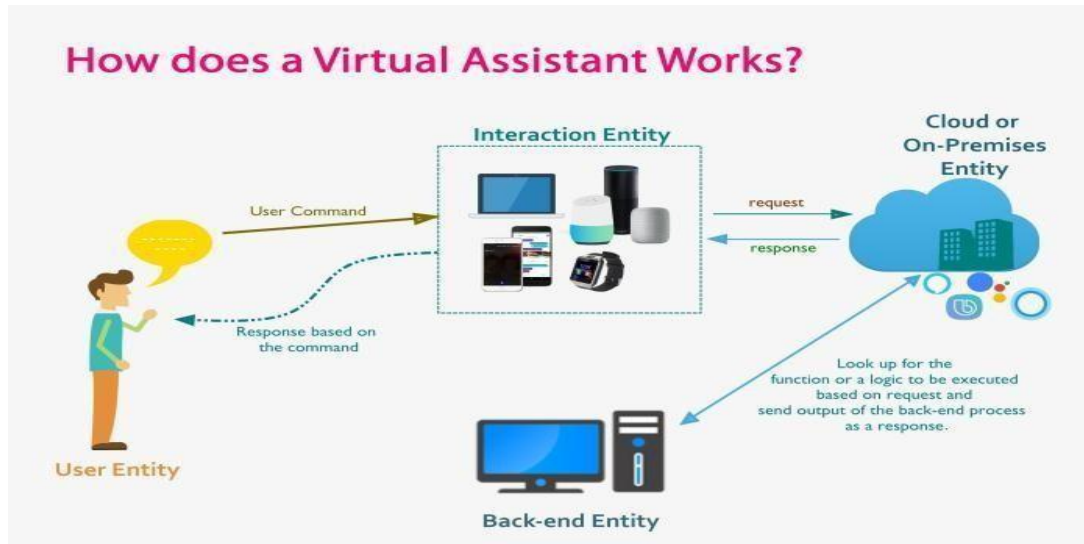


Figure 2.1. Shows backend working data flow

User input may be matched with executable instructions using Natural Language Processing (NLP). An audio signal is translated into executable commands or digital data that may be used by software to do a specific action when a user asks a query. VirtualAssistant is used to operate machines based on your own instructions, and then this data is compared with software data to obtain an appropriate solution. We utilise python installers like- to create virtual assistants

Horn proposes a classroom environment (2018). Each classroom should have enough microphones to detect each student's voice and offer individualised replies to each student's headphones via voice assistants, according to the author. Each classroom might have a smart speaker where students can ask questions. Alternatively Teachers should have access to voice assistant data in real time so they may step in as necessary. Teachers are not replaced by the gadgets, but rather their job is amplified by their use of them.

Neiffer investigates the impact of intentional education using the intelligent voice assistant Siri on student participation in science classes in upper elementary and middle school grades (2018). Student involvement is connected with student graduation rates. High student involvement leads to greater teacher's satisfaction and pleasure. Research shows that there is too much complexity in the relationship between technology and education to draw any firm conclusions. Furthermore, there is no clear correlation between the use of Siri in 5th and middle school science classrooms and an increase in students' interest in learning science. A unique Alexa Skill on Scotland was made by Davie and Hilber (2018), who utilised it with students prior to a trip to the country. Students utilised the Amazon Echo gadget and found the talent to be interesting.

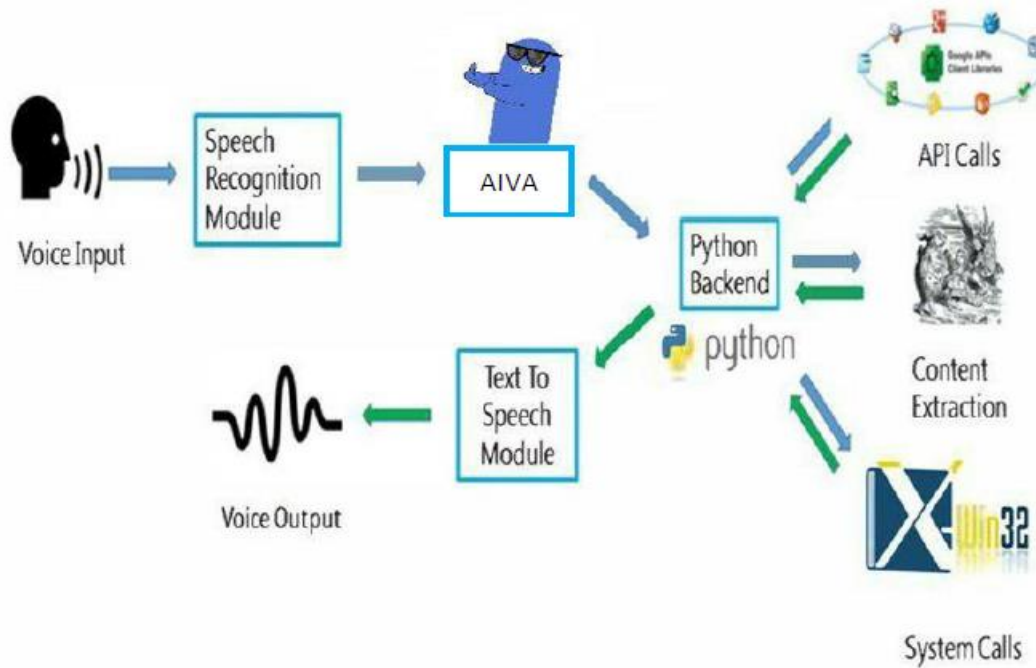


Figure 2.2. input/output text to speech

The proposed multi-domain ASR framework consists of three main modules: a basic ASR module to conduct first-pass decoding and generate top N hypotheses of a speech query, a text classification module to determine which domain the speech query belongs to, and a reranking module to rescore n-best lists of the first-pass decoding output using domainspecific language models. Figure 1 shows the diagram of the proposed multi-domain ASR framework

Speech recognition:

To translate spoken input into text, the system makes use of Google's online voice recognition technology. A particular corpus of voice data is saved on a computer network server at the information centre and then delivered to Google Cloud for speech recognition, allowing users to talk and get the text as a result of their voice input. The voice assistant application receives and sends the exact same text.

Backend in Python:

Python is used as the backend for the whole software. Context extraction, API calls, and system calls are all types of calls that the Python Backend can distinguish between with the use of a speech recognition module. The output is then provided back to the respondent.

Calls to the application programming interface:

The API's job is to act as a bridge between two programmes so that they may communicate with one another. This means that APIs act as a messenger between the service provider and the user, delivering their requests and subsequently returning their responses.

Content Extraction:

Machine-readable documents that are unstructured or semi-structured may be automatically analysed using Context Extraction. Natural language processing (NLP) is used in this activity to process documents written in human language. Content extraction might include tasks like image/video/audio annotation and content extraction

System Calls

For example, accessing the hard disc drive, creating new processes, and communicating with process scheduling are all examples of System Calls. An key part of the OS-process interaction is provided by this component.

Google-Text-to-Speech

For the most part, Text-To-Speech is used to turn user-provided Text into Speech. Sound may be generated from the phonemic representation of the text by a TTS Engine once it has been translated into waveform form. Third-party publishers have contributed a variety of languages to the TTS's growing feature set.

2.1 PRESENT SYSTEM

Many current voice assistants, such as Alexa, Siri, Google Assistant, and Cortana, utilise the language processing and speech recognition concepts that we are all acquainted with. They pay attention to the user's instructions and carry out the requested task quickly and effectively.

Using Artificial Intelligence, these voice assistants are able to provide results that are very accurate and efficient. Using these assistants, we may do more with less human effort and time consumption, since they do not need any typing at all and act as if they were an actual person to whom we were conversing and giving instructions. There is no comparison between these helpers and a person, yet we can state that they are more effective and efficient at doing any duty. Because of this, the method utilised to create these assistants minimises the amount of time required.

These assistants, however, need an account (such as a Google or Microsoft account) and an internet connection in order to be used, since these assistants will only function while connected to the internet. They are compatible with a wide range of gadgets, including mobile phones, computers, and speakers, among others.

2.2 PROPOSED SYSTEM

Creating my own personal helper was a fascinating challenge. With the use of a single voice command, you can now send emails, search the internet, play music, and launch your favourite IDE without ever having to open a browser. While most standard voice assistants rely on an internet connection to get instructions, Jarvis is unique in that it is desktop-specific and does not need a user account in order to use it.

VSCode is the IDE used in this project. Using VSCode, I was able to construct the

python files and install all of the essential dependencies. It was necessary to utilise the following modules and libraries for this project, including pyttsx3, SpeechRecognition and Datetime. Using the JARVIS, I've constructed a live GUI that allows me to interact with it in a more visually appealing way.

Tutor's growth means that he or she can complete any work as effectively as we can, or even better. I discovered that the notion of AI in every sector reduces human work and saves time via the creation of this project. Among the features of this project are the ability to send emails and read PDF files; the ability to launch command prompt, your preferred IDE, notepad, and other applications; the ability to play music; the ability to make Wikipedia searches; and the ability to set up desktop reminders of your choosing. Basic discussion is possible

The following functionalities will be included in the system as proposed:

- 1.)In order to respond to a call with the specified functionality, it always retains a list of its name.
- 2.)In addition, it retains the sequence of inquiries asked of it in relation to its setting, which it uses in the future. As a result, every time the identical situation is brought up, you'll be in a position to bring up pertinent points of discussion.
- 3) Using voice instructions to do arithmetic computations and returning the results by voice.
- 4.)In this fourth step, the computer searches the Internet depending on the user's voice input and returns a voice response with more interactive questions.
- 5) The data on its cloud server will maintain auto synchronisation up to date.
- 6.)Update the data in the cloud with the help of a Firebase server.
- 7.)User may connect smart devices and conduct actions such as turning on and off lights with the assistance of the IoT architecture.
- 8.)Push notifications, such as email or text messages, may be used to alert the owner of a smartphone.
- 9.)Some more options include playing music, setting an alarm, and monitoring local weather conditions. The use of reminders, spell-checks, etc

Chapter 3

System Design

Flow Charts and System Diagrams

Speech-to-Text Interface

The goal of voice recognition is to offer a way to convert spoken words into written ones. This objective may be achieved in a variety of ways. Building models for each word that has to be identified is the simplest method. Speech signal mainly transmits the words or message being said. The underlying meaning of the utterance is the focus of speech recognition. Extracting and modelling the speech-dependent properties that may successfully differentiate one word from another is the key to success in speech recognition. The system consists of a set of components.

Due to the fact that all systems are based on machine learning and employ vast quantities of data acquired from different sources and then trained on them, the source of this data plays a vital part in their production. The kind of assistance that emerges depends on the quantity of data gathered from various sources. Despite the wide variety of learning methodologies, algorithms, and techniques, the basic building blocks of these systems remain essentially the same across the industry.

Assistive technology A virtual assistant is often a cloud-based application that works with devices connected to the internet. is the ability to contract for just the services they need. As a means of developing a virtual assistant Python will take over your PC. Task-oriented virtual assistants are the most common kind of virtual assistant. The use of a remote assistance understanding of and capacity to follow instructions

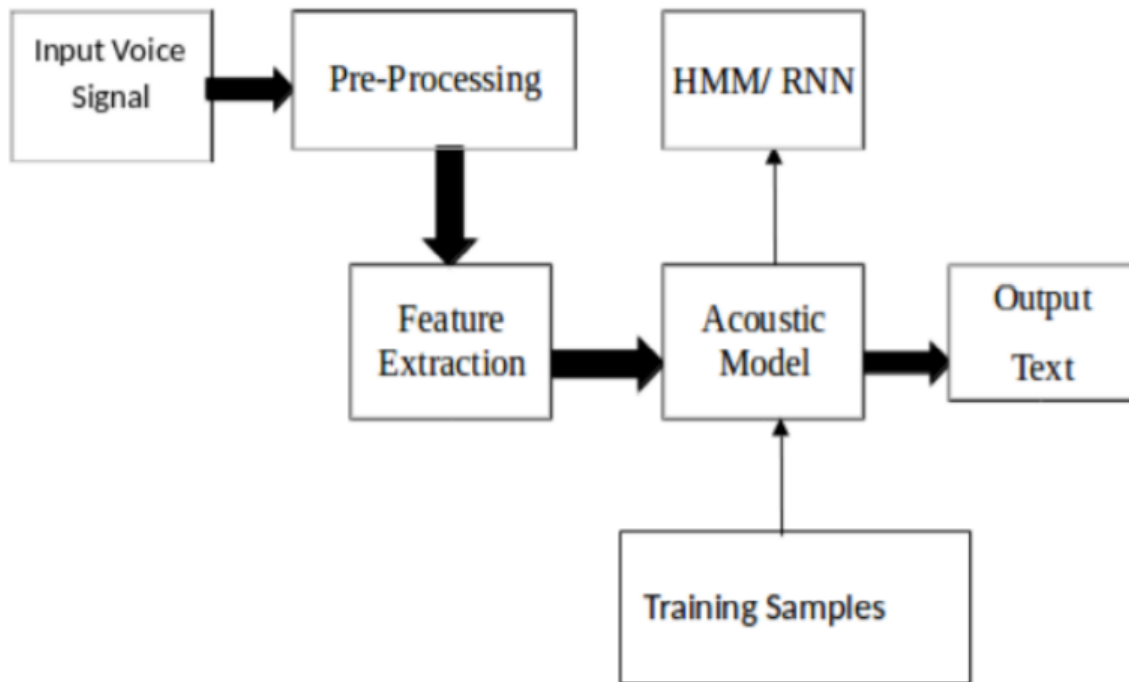


Figure 3.1. Shows signals data flows

DATA TRAFFIC.

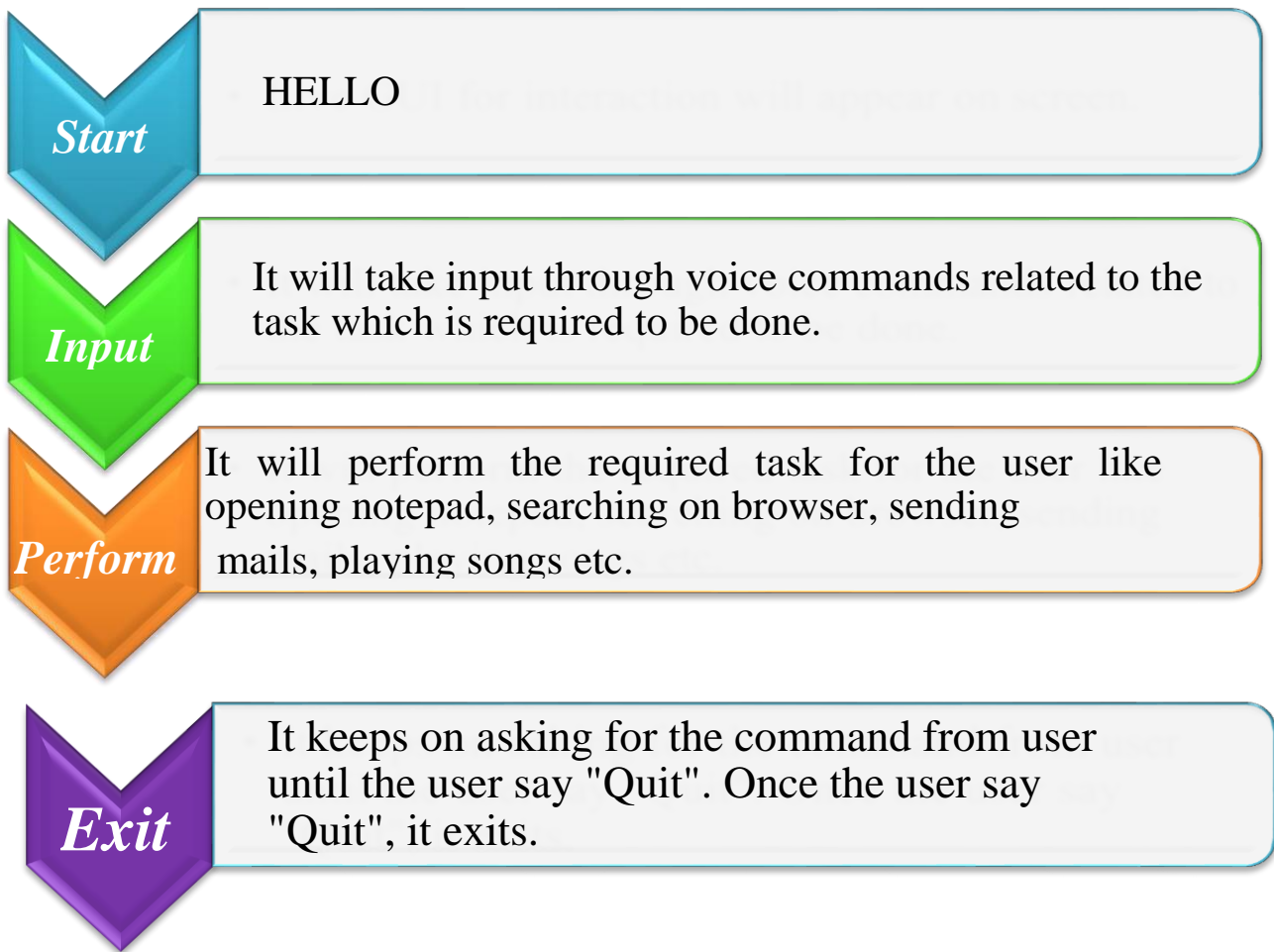


Figure 3.2 shows Tutor data flow

The system is built on the idea of Artificial Intelligence and the relevant Python packages. pyttsx3 can read PDFs using Python's various libraries and packages, such as python. Chapter 3 of this study goes into depth about these packages.

Everything in this project is based on human input, thus the assistant will do anything the user commands it to do. Everything a user wishes to be done, in human language, may be entered as a list of tasks. English.

Student involvement is connected with student graduation rates. High student involvement leads to greater teacher's satisfaction and pleasure. Research shows that there is too much complexity in the relationship between technology and education to draw any firm conclusions.

3.1 ARCHITECTING A COMPUTER SYSTEM

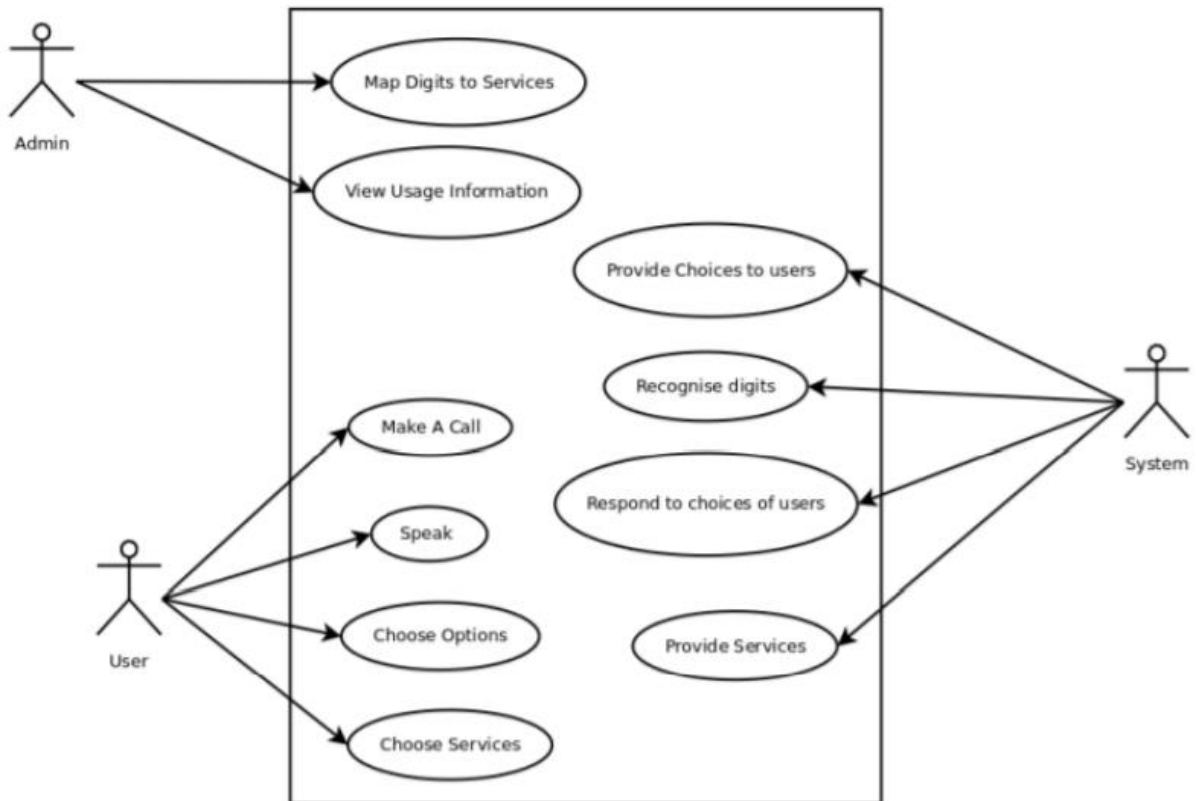


Figure .3.3 shows use case diagram

Chapter 4

SPECIFICATIONS OF THE PROGRAM

VSCoDe is the IDE of choice for this project. Every piece of Python code was written in VSCoDe and the essential packages could be installed with relative ease using this IDE. It was necessary to utilise pyttsx3, SpeechRecognition, Datetime, Wikipedia, Smtplib and pyAudio in order to complete this project. Using the JARVIS, I've constructed a live GUI that allows me to interact with it in a more visually appealing way.

VSCODE

It's an IDE, for short. Scientific tools like matplotlib, numpy and scipy, web frameworks like Django and web2py, refactoring in Python, integrated python debugger, code completion, code and project navigation etc. are all supported by this integrated development environment. When used in conjunction with Anaconda, it offers Data Science as a side benefit

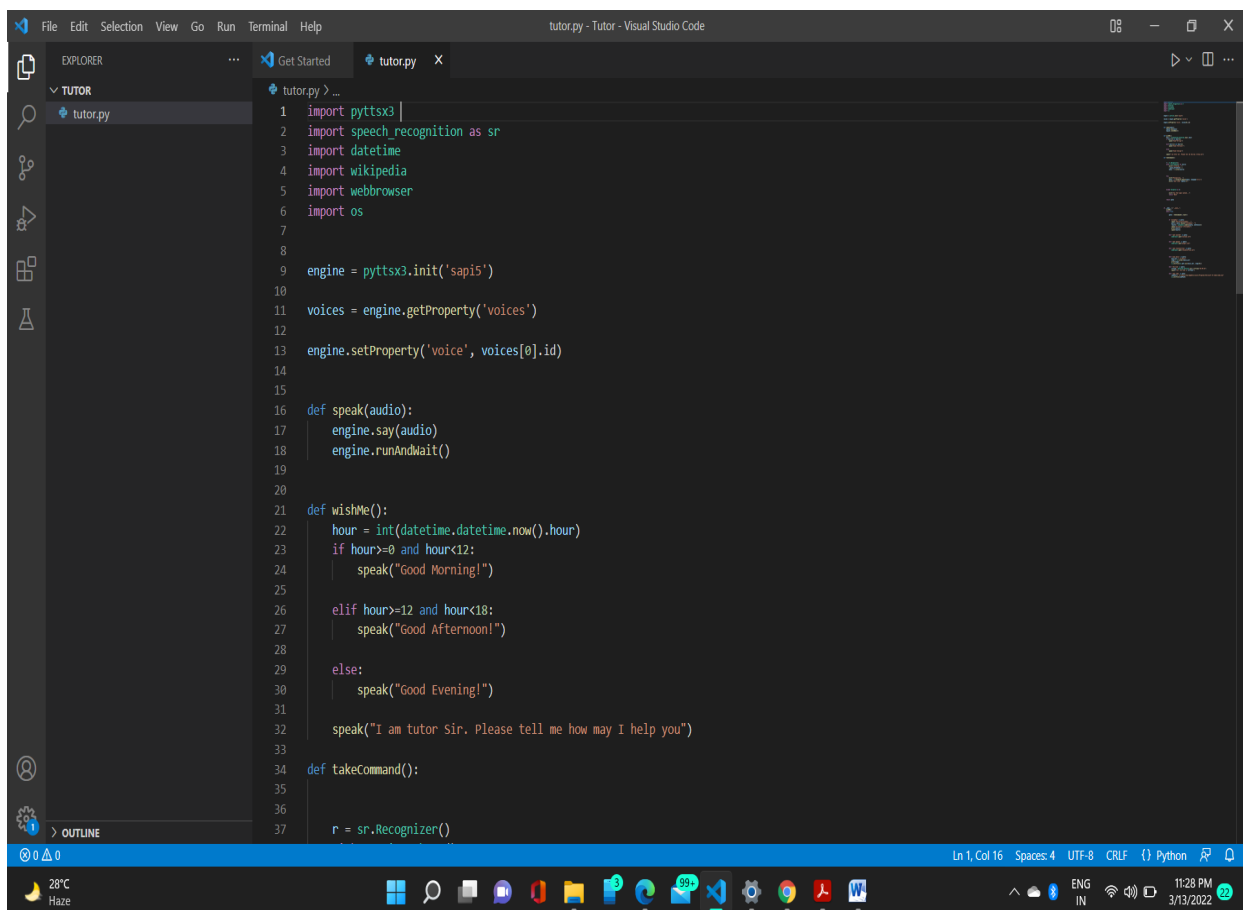
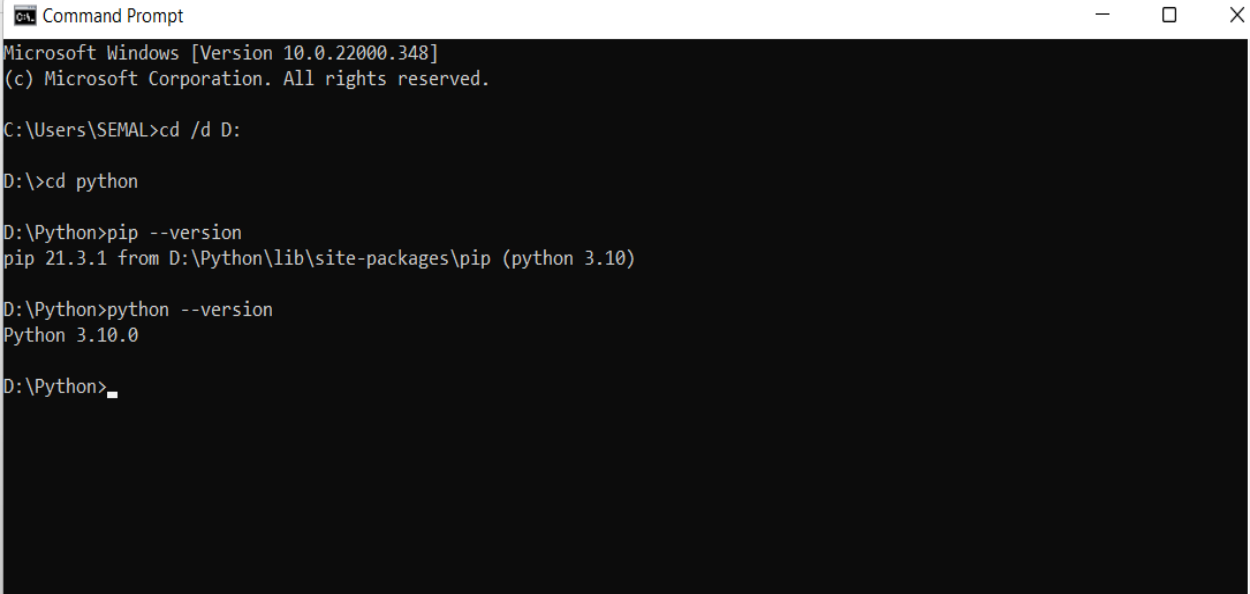


Figure 4.1. Shows the Implementation of library

GUIDE TO PYTHON FILES

This Python library was used in JARVIS:

- 1.)Pytttsx3: A python library that translates text into voice.
- 2.)Python package SpeechRecognition translates voice into text.
- 3.)Datetime: We can get the current date and time using this library.
- 4.)A python module for searching Wikipedia is provided.
- 5.)Sending and routing emails between mail servers is made simple using Smtplib, a lightweight email transfer protocol.
- 6.)OS: It's a set of Python libraries that rely on the System to provide data.
- 7.)Webbrowser: It gives users an interface for viewing documents hosted on the
- 8.)Pyjokes- Pyjokes is a collection of jokes that may be found online. Pyjokes is included in our project since it provides a source of amusement. It's a fascinating topic. In our project, Pyjokes is the one-sentence joke that keeps it interesting.



```
Command Prompt
Microsoft Windows [Version 10.0.22000.348]
(c) Microsoft Corporation. All rights reserved.

C:\Users\SEMAL>cd /d D:

D:\>cd python

D:\Python>pip --version
pip 21.3.1 from D:\Python\lib\site-packages\pip (python 3.10)

D:\Python>python --version
Python 3.10.0

D:\Python>_
```

Figure 4.2 shows the command prompt window

Result

Using a virtual assistant saves time. When a customer gives a job to a virtual assistant, the programme takes over and completes it. Using Natural Language Processing (NLP), virtual assistants may match a user's voice or text input to a command. You may operate your laptop or PC on your own with the aid of a virtual assistant. It's a time-saver since it's so simple. You can always count on your virtual assistant to be there when you need them, and they'll be able to respond swiftly to any changes in your schedule or priorities.

We've covered Python-based Personal Virtual Assistants for Windows in this article. Humans' lives are made simpler by virtual assistants. Using a virtual assistant gives you the freedom to contract for just the services you need.. Python is used to create virtual assistants for all Windows versions, much as Alexa, Cortana, Siri, and Google Assistant. Artificial Intelligence is used in this project, and virtual personal assistants are an excellent method to keep track of your calendar. Because of their portability, loyalty, and availability at any moment, virtual personal assistants are more dependable than human personal assistants. Our virtual assistant will get to know you better and be able to provide ideas and follow orders. This device will most likely be with us for the rest of our lives.

As a result, voice recognition systems have made their way into a wide range of industries. The use of speech signals as input to a system is one of the many advantages of IVR (Interactive Voice Response) systems. This is why we proposed the creation of an Interactive Voice Response (IVR) system that includes automatic speech recognition (ASR). It was the primary goal of the project to design a system that could recognise speech signals in the Nepali language for input to the IVR system..

Many difficulties and challenges arose throughout the development period, which pushed us to design a system that could recognise words in the Nepali language based on their numbers. To do this, we conducted extensive study into several speech-recognition systems and applied the data to build the system.

Outcome

During this research, we have built a voice-activated assistant that can do any action in response to orders supplied by the user without mistake. More functions have been introduced, such as listening exclusively to the users' speech and not being triggered by background noise. This project is easier to grasp and more adaptable because of its modular design. The program's functionality won't be affected if we add new features. The VS Code Integrated Development Environment was used to implement the Python code, which included installing all of the necessary packages (IDE). This project made use of Python 3.x and data from

Other environmental sounds were also used to create other noise

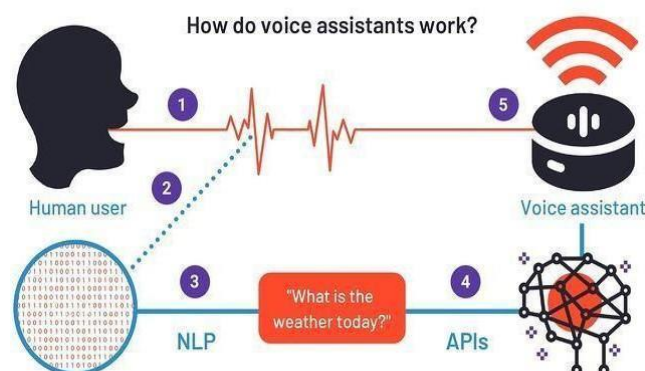


Figure 4.3 shows voice assistance work

Summary

In this research, we created a voice assistant that can carry out any job provided by the users without mistake in exchange for their requests. We've also incorporated features like listening exclusively to the user's speech and not being triggered by background noise. This project is easier to grasp and more adaptable because of its modular design. Additional features may be added without affecting the program's usability. The VS Code Integrated Development Environment was used to implement the Python code, which included installing all of the necessary packages (IDE). This software used Python 3.x and data from the environment Methodology for various sounds..

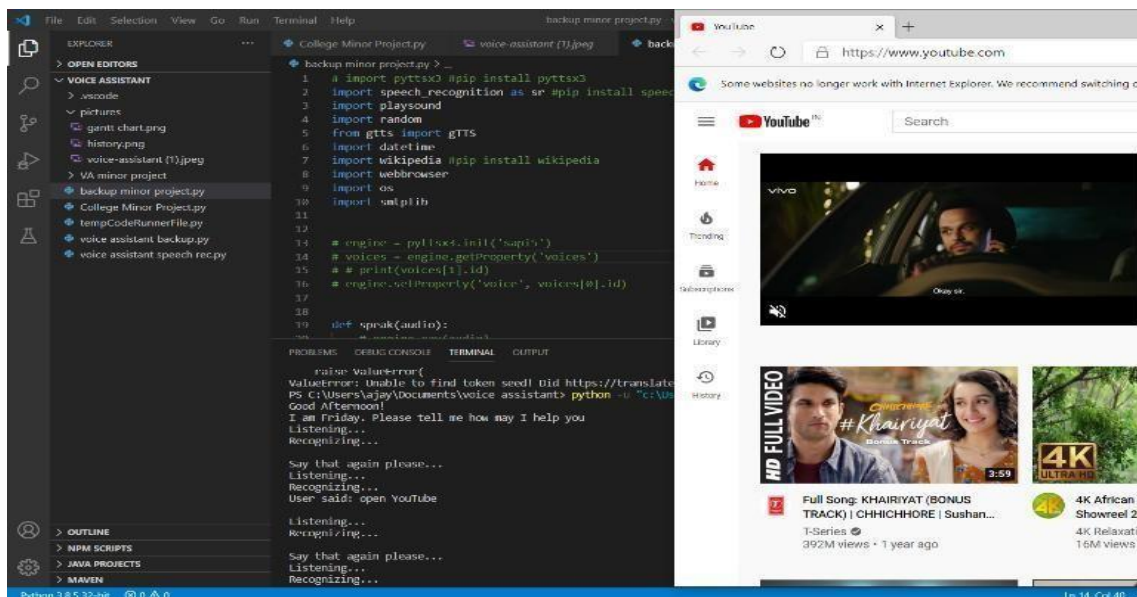


Figure 4.4 output of command

Chapter 5

IMPLEMENTATION WORK DETAILS

In order to automate many of the routine desktop operations, such as playing music or launching your preferred IDE, TUTOR, a desktop assistant, utilises a voice assistant. While most standard voice assistants rely on an internet connection to get instructions, Jarvis is unique in that it is desktop-specific and does not need a user account in order to use it.

5.1 APPLICATION IN THE REAL WORLD

This desktop voice assistant, known as TUTOR, allows us to do a variety of things just by speaking our commands into the gadget.

Using a conversational approach, it streamlines the process of accomplishing any operation by using Python's most important modules and libraries automatically. The conversational interaction between providing input and receiving the required output in the form of a job completed makes it seem like a human helper when a user instructs it to do a task.

As a result of its receptive character, the desktop assistant responds to the user in a fashion that is intelligible to human beings, which is why it responds in English. As a result, the user reacts intelligently and with knowledge.

The most common use of it is its capacity to do many tasks at once. It may keep asking for instructions one after the other until the user says "QUIT" to stop it.

After receiving the user's instructions, it just does the work without requiring a "trigger phase" to begin the process. Many difficulties and challenges arose throughout the development period, which pushed us to design a system that could recognise words in the Nepali language based on their numbers. To do this, we conducted extensive study into several speech-recognition systems and applied the data to build the system.

5.2 DATA IMPLEMENTATION AND PROGRAM EXECUTION

Installing all of the required packages and libraries is a good place to start. Installing the libraries is as simple as running "pip install" and then importing the results. The following components are included in the set:FUNCTIONS

- 1.) Use takeCommand() to get a command from the user's microphone and return it as a string with the function's output.
- 2.) Good Morning, Good Afternoon, and Good Evening are some of the greetings that wishMe() may send to the user based on the current time.
- 3.) SendEmail(), pdf reader(), news(), and numerous conditions in if conditions like "open google," "open notepad," "search on Wikipedia," "play music," and so on and so forth are all defined in taskExecution().

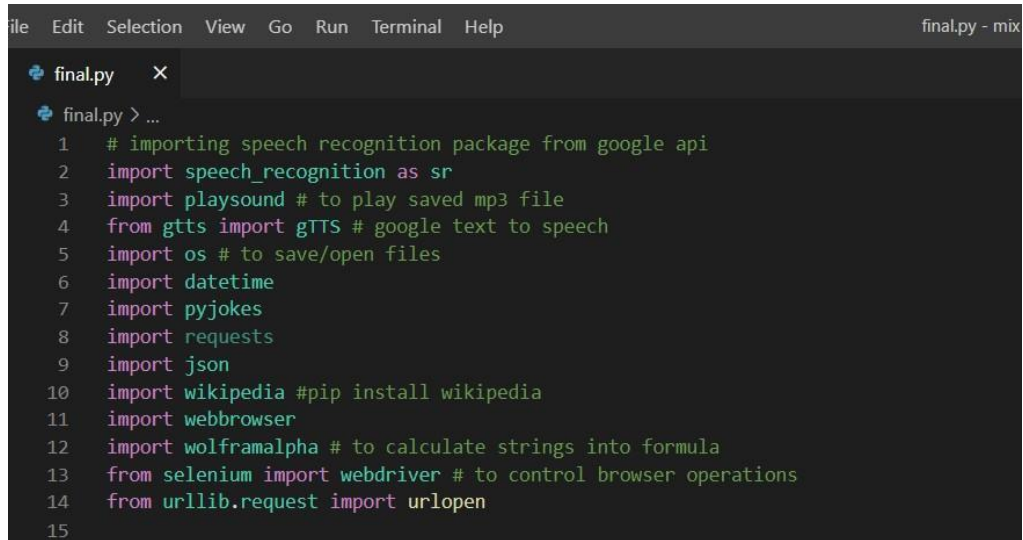
Without a doubt, the effectiveness and efficiency of Tutor as a voice assistant make it a valuable tool for busy users. These limitations and opportunities for improvement were discovered while working on this project, which are outlined in the following sections. Artificial Intelligence and Natural Language Processing will be used to create a voice-

activated personal assistant that can operate IoT devices and even search the web for answers to specific questions. There are various subsystems that may be automated to reduce the amount of time and effort required to communicate with the main system. The system's goal is to make human existence as pleasant as possible. In further detail, this system is meant to communicate intelligently with other subsystems and operate these devices, including Internet of Things (IoT) devices or receiving news from the Internet, delivering other information, obtaining customised data previously kept on the system, and so on. The Android app should allow the user to add data, such as calendar entries, alarms, or reminders, to the app. All of these platforms will be made more accessible with the help of the software, which will go through the following stages: voice data collecting, analysis, text conversion, data storage, and speech generation from text output processed via these stages. The data collected at each stage may be utilised to identify trends and provide recommendations to the user. Artificial intelligence devices that can learn and comprehend their users may utilise this as a significant foundation. It has been determined that the suggested system would not only make it easier for us to interface with other systems and modules, but it also helps us stay organised. With a little help from the device we can help build a new generation of voice-controlled devices and bring about a long-term change in the automation industry. A prototype for a wide range of future applications can be found in this paper.

As a result, voice recognition systems have made their way into a wide range of industries. The use of speech signals as input to a system is one of the many advantages of IVR (Interactive Voice Response) systems. This is why we proposed the creation of an Interactive Voice Response (IVR) system that includes automatic speech recognition (ASR). It was the primary goal of the project to design a system that could recognise speech signals in the Nepali language.

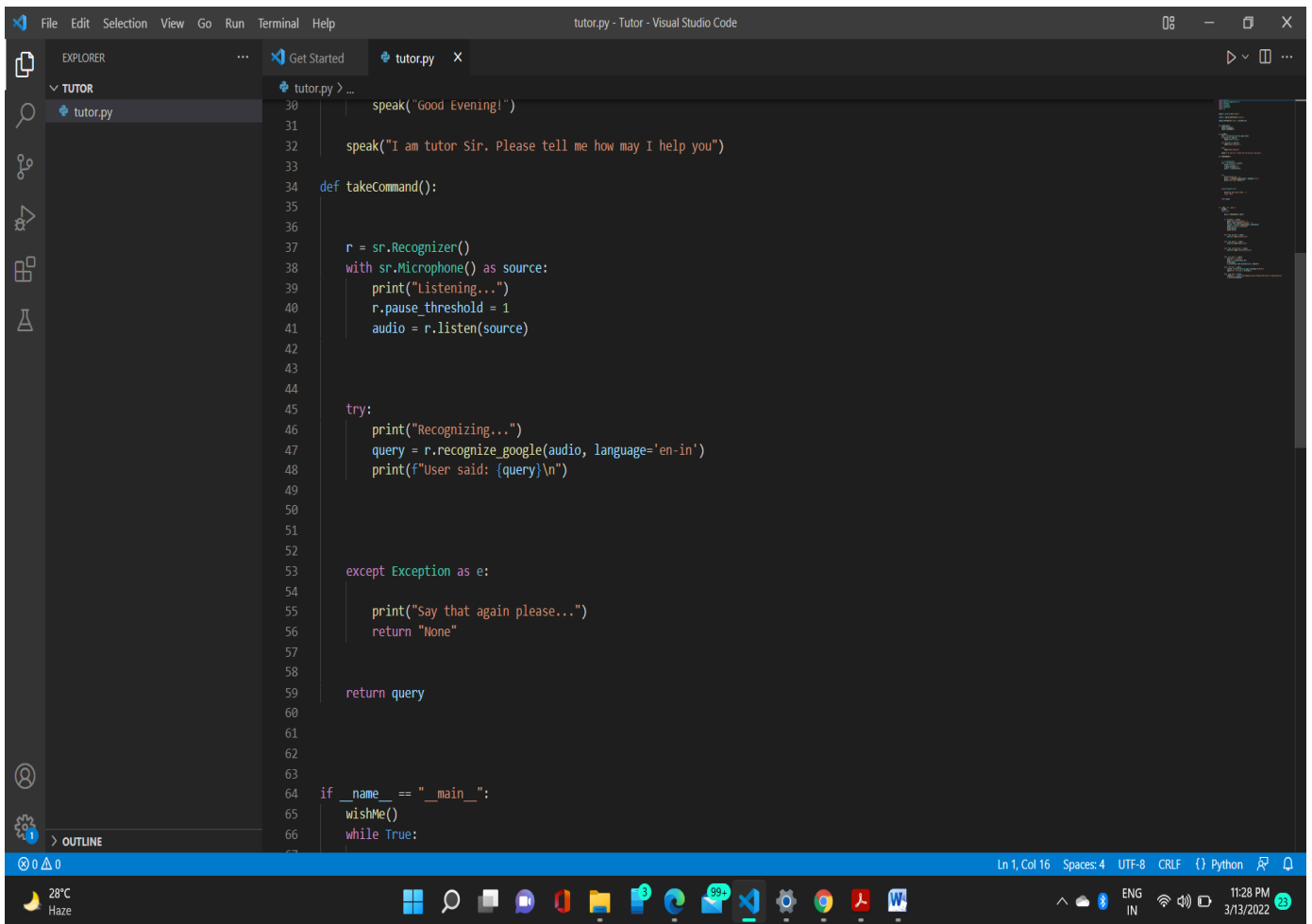
Chapter 6

SOURCE CODE AND COMMANDS

A screenshot of a code editor window titled 'final.py - mix'. The editor shows a Python file named 'final.py' with 15 lines of code. The code consists of a series of import statements for various Python libraries. The libraries imported are: speech_recognition (as sr), playsound, gtts (GTTTS), os, datetime, pyjokes, requests, json, wikipedia (with a comment to install it via pip), webbrowser, wolframalpha, selenium (with webdriver), and urllib.request (with urlopen). The code is syntax-highlighted with green for comments, blue for keywords like 'import' and 'from', and black for the library names.

```
1 # importing speech recognition package from google api
2 import speech_recognition as sr
3 import playsound # to play saved mp3 file
4 from gtts import gTTS # google text to speech
5 import os # to save/open files
6 import datetime
7 import pyjokes
8 import requests
9 import json
10 import wikipedia #pip install wikipedia
11 import webbrowser
12 import wolframalpha # to calculate strings into formula
13 from selenium import webdriver # to control browser operations
14 from urllib.request import urlopen
15
```

Figure 6.1. Shows all the libraries



```
File Edit Selection View Go Run Terminal Help
tutor.py - Tutor - Visual Studio Code

EXPLORER
TUTOR
  tutor.py

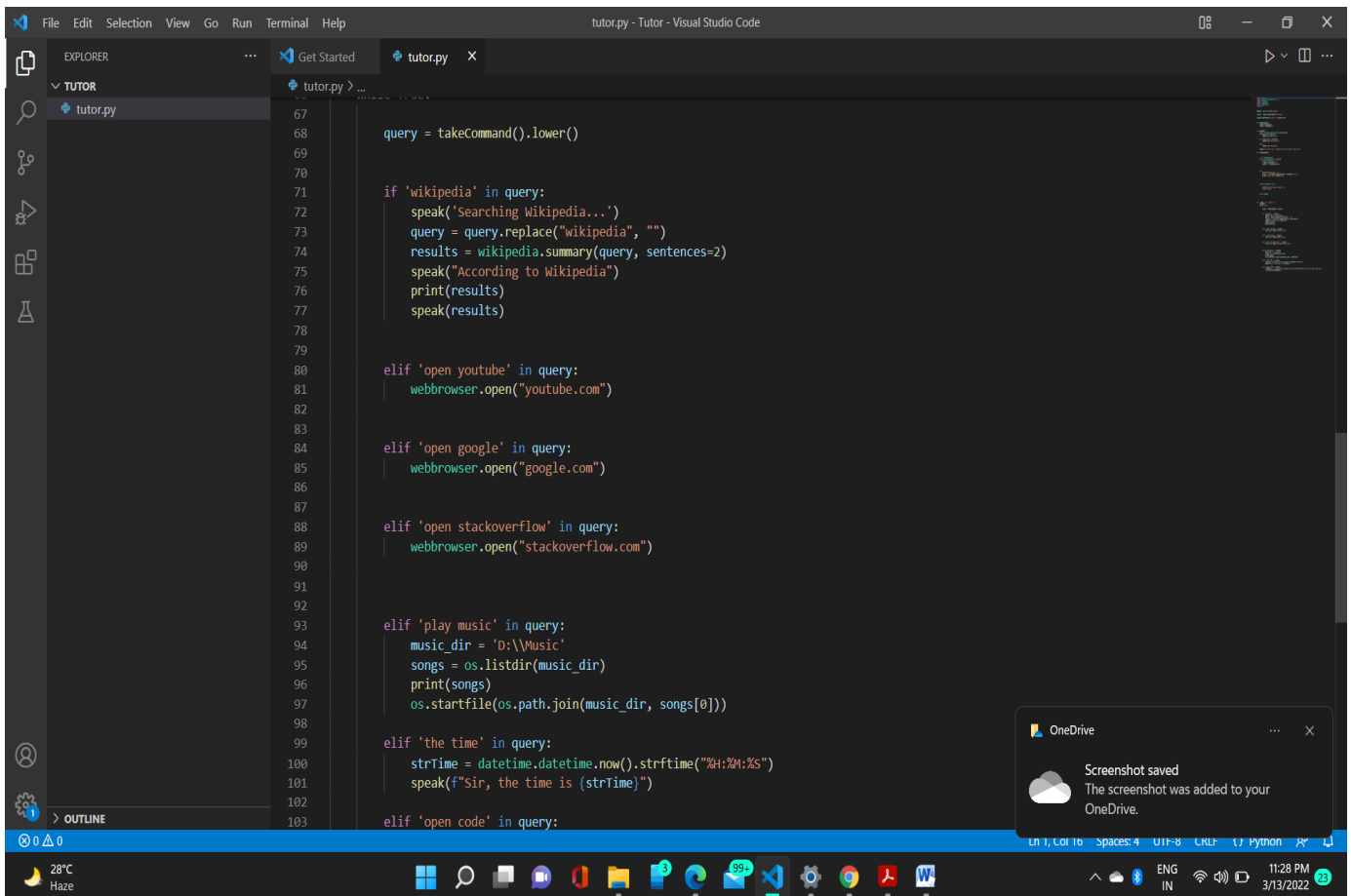
Search
Run and Debug

tutor.py > ...
30 speak("Good Evening!")
31
32 speak("I am tutor Sir. Please tell me how may I help you")
33
34 def takeCommand():
35
36
37     r = sr.Recognizer()
38     with sr.Microphone() as source:
39         print("Listening...")
40         r.pause_threshold = 1
41         audio = r.listen(source)
42
43
44
45     try:
46         print("Recognizing...")
47         query = r.recognize_google(audio, language='en-in')
48         print(f"User said: {query}\n")
49
50
51
52
53     except Exception as e:
54
55         print("Say that again please...")
56         return "None"
57
58
59     return query
60
61
62
63
64 if __name__ == "__main__":
65     wishMe()
66     while True:
```

Ln 1, Col 16 Spaces: 4 UTF-8 CRLF Python

28°C Haze 11:28 PM 3/13/2022

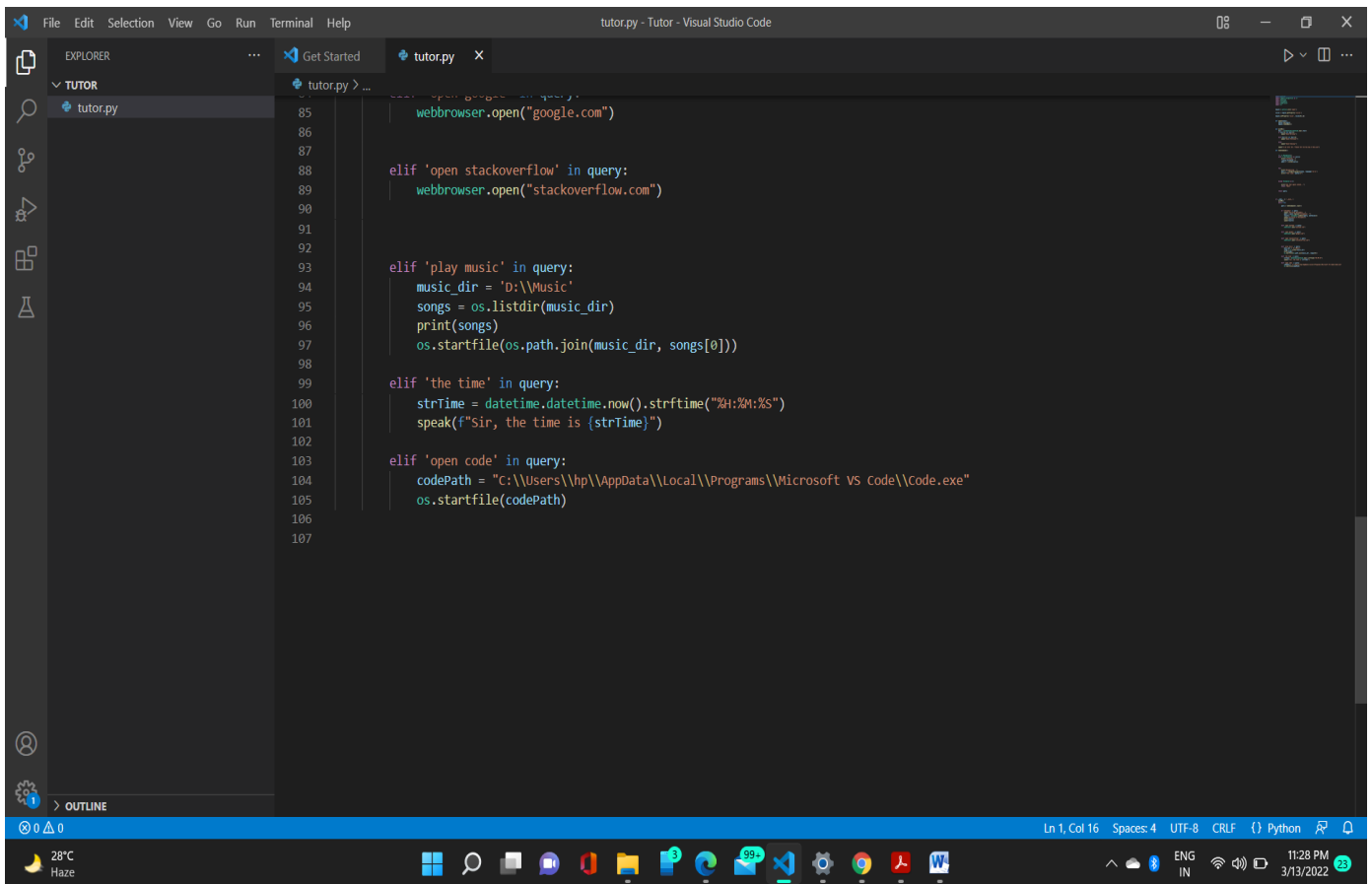
Figure 6.2. Shows the screenshot of the code



```
67 query = takeCommand().lower()
68
69
70
71 if 'wikipedia' in query:
72     speak('Searching Wikipedia...')
73     query = query.replace("wikipedia", "")
74     results = wikipedia.summary(query, sentences=2)
75     speak("According to Wikipedia")
76     print(results)
77     speak(results)
78
79
80 elif 'open youtube' in query:
81     webbrowser.open("youtube.com")
82
83
84 elif 'open google' in query:
85     webbrowser.open("google.com")
86
87
88 elif 'open stackoverflow' in query:
89     webbrowser.open("stackoverflow.com")
90
91
92
93 elif 'play music' in query:
94     music_dir = 'D:\\Music'
95     songs = os.listdir(music_dir)
96     print(songs)
97     os.startfile(os.path.join(music_dir, songs[0]))
98
99
100 elif 'the time' in query:
101     strTime = datetime.datetime.now().strftime("%H:%M:%S")
102     speak(f"Sir, the time is {strTime}")
103
104 elif 'open code' in query:
```

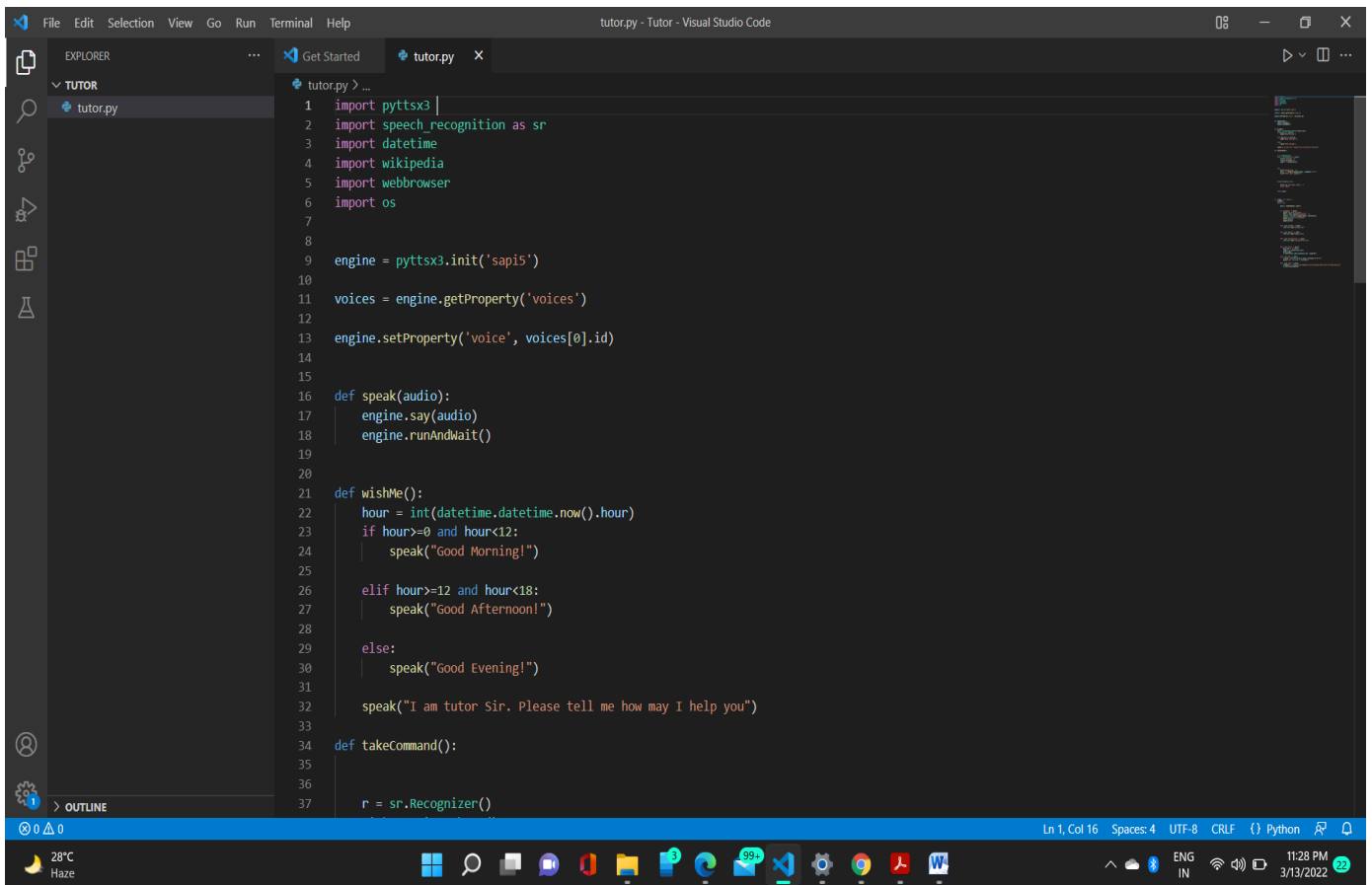
OneDrive
Screenshot saved
The screenshot was added to your OneDrive.

Figure 6.3. Shows the screenshot of the code



```
85     webbrowser.open("google.com")
86
87
88 elif 'open stackoverflow' in query:
89     webbrowser.open("stackoverflow.com")
90
91
92
93 elif 'play music' in query:
94     music_dir = 'D:\\Music'
95     songs = os.listdir(music_dir)
96     print(songs)
97     os.startfile(os.path.join(music_dir, songs[0]))
98
99
100 elif 'the time' in query:
101     strTime = datetime.datetime.now().strftime("%H:%M:%S")
102     speak(f"Sir, the time is {strTime}")
103
104 elif 'open code' in query:
105     codePath = "C:\\Users\\hp\\AppData\\Local\\Programs\\Microsoft VS Code\\Code.exe"
106     os.startfile(codePath)
107
```

Figure 6.4. Shows the screenshot of the code



```
1 import pyttsx3
2 import speech_recognition as sr
3 import datetime
4 import wikipedia
5 import webbrowser
6 import os
7
8
9 engine = pyttsx3.init('sapi5')
10
11 voices = engine.getProperty('voices')
12 engine.setProperty('voice', voices[0].id)
13
14
15
16 def speak(audio):
17     engine.say(audio)
18     engine.runAndWait()
19
20
21 def wishMe():
22     hour = int(datetime.datetime.now().hour)
23     if hour>=0 and hour<12:
24         speak("Good Morning!")
25
26     elif hour>=12 and hour<18:
27         speak("Good Afternoon!")
28
29     else:
30         speak("Good Evening!")
31
32     speak("I am tutor Sir. Please tell me how may I help you")
33
34 def takeCommand():
35
36     r = sr.Recognizer()
```

Figure 6.5. Shows the screenshot of the code

Chapter 7

INPUT/OUTPUT SCREENSHOT

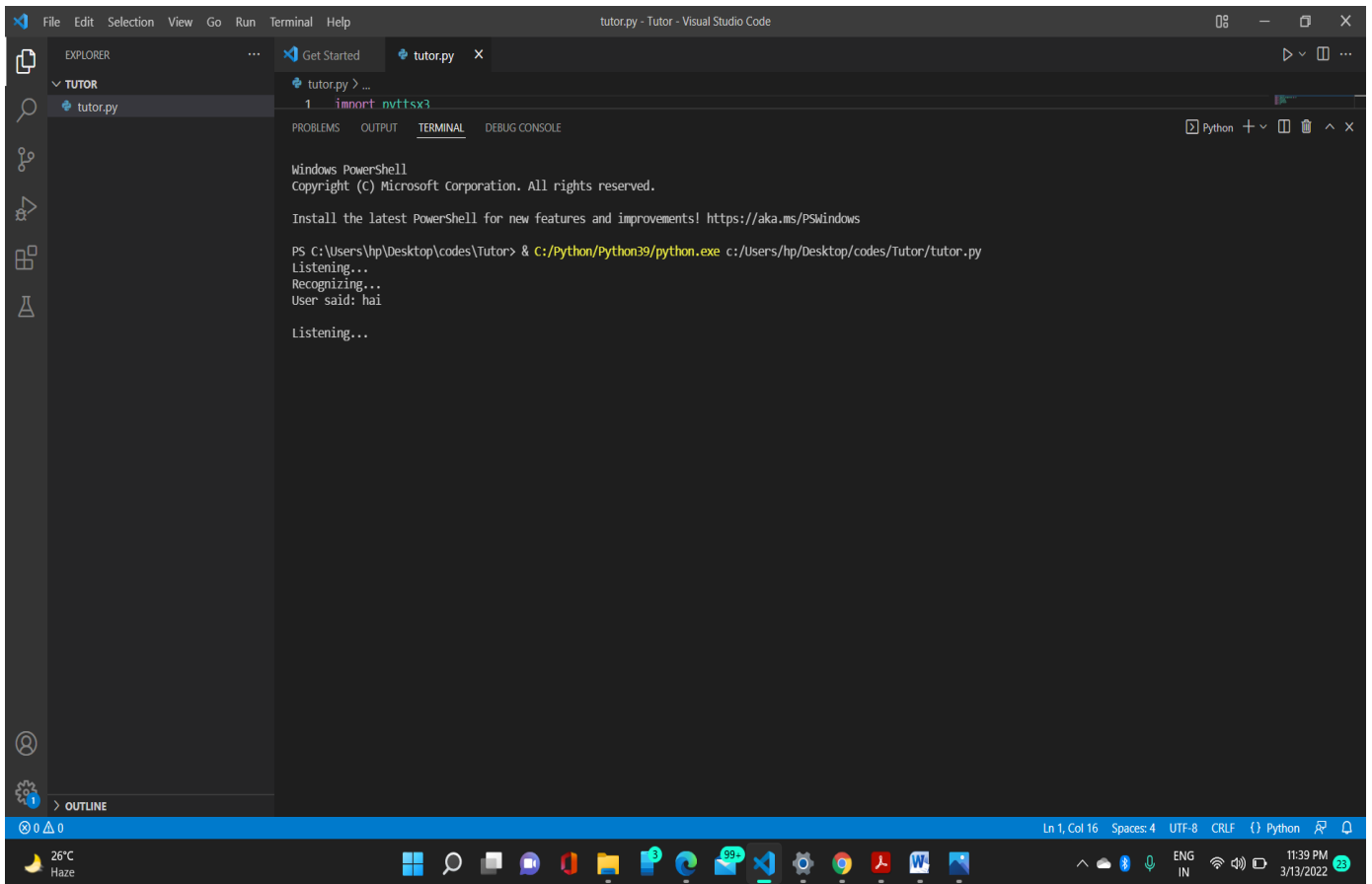


Figure 7.1. Shows the screenshot of the output

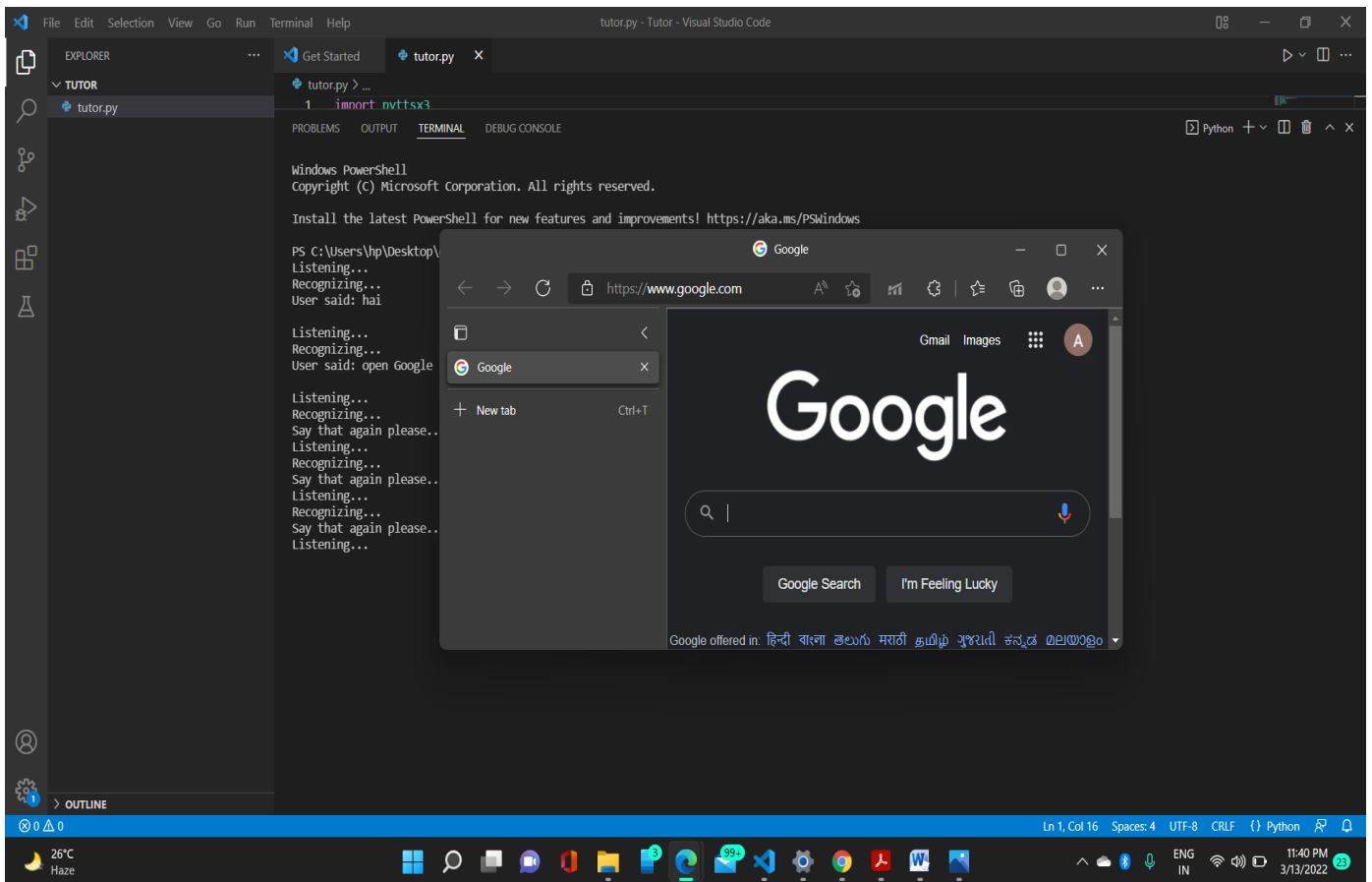


Figure 7.2. Shows the output of the code “hello google”

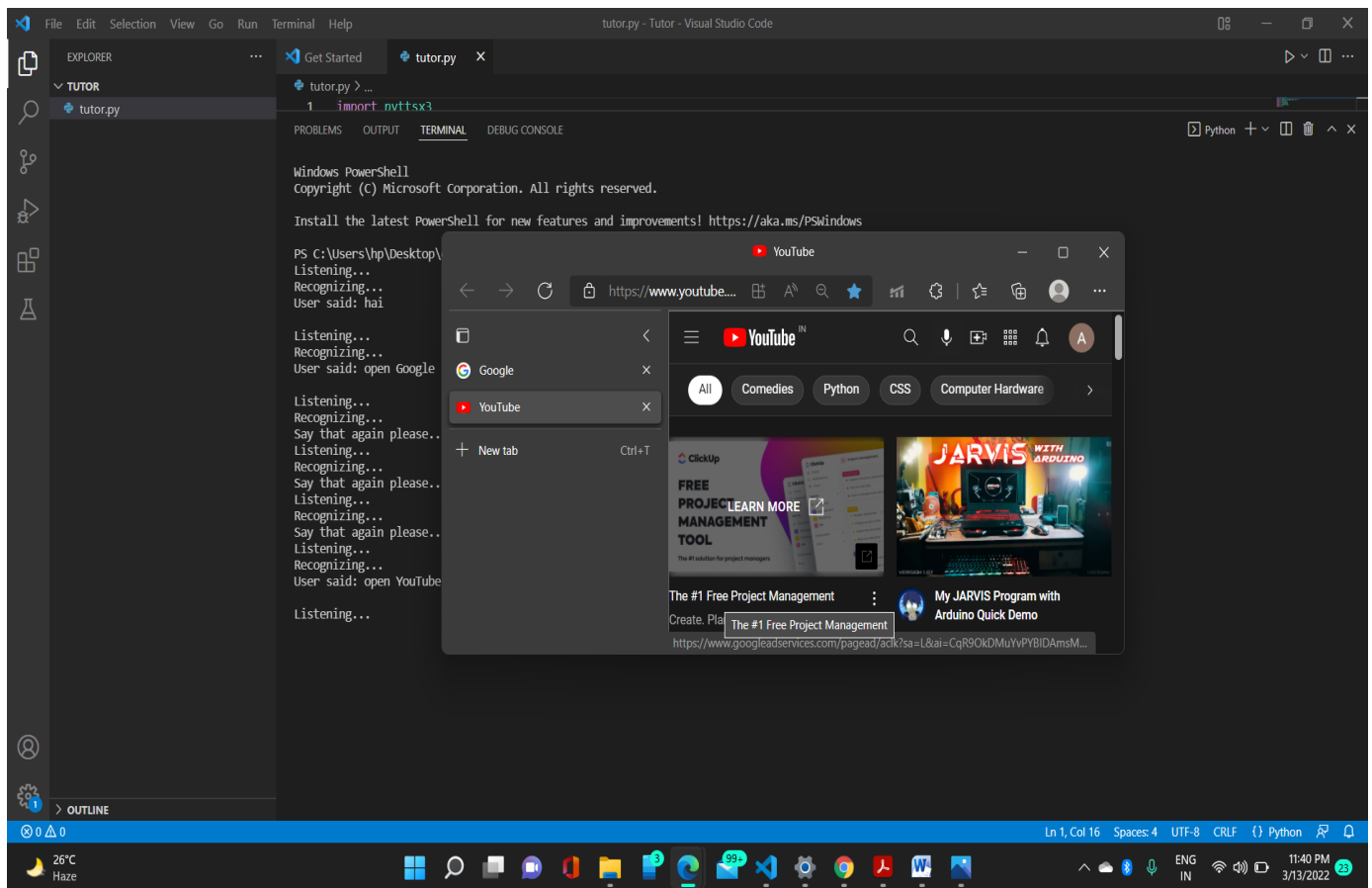


Figure 7.3. Shows the output of the code “open youtube”

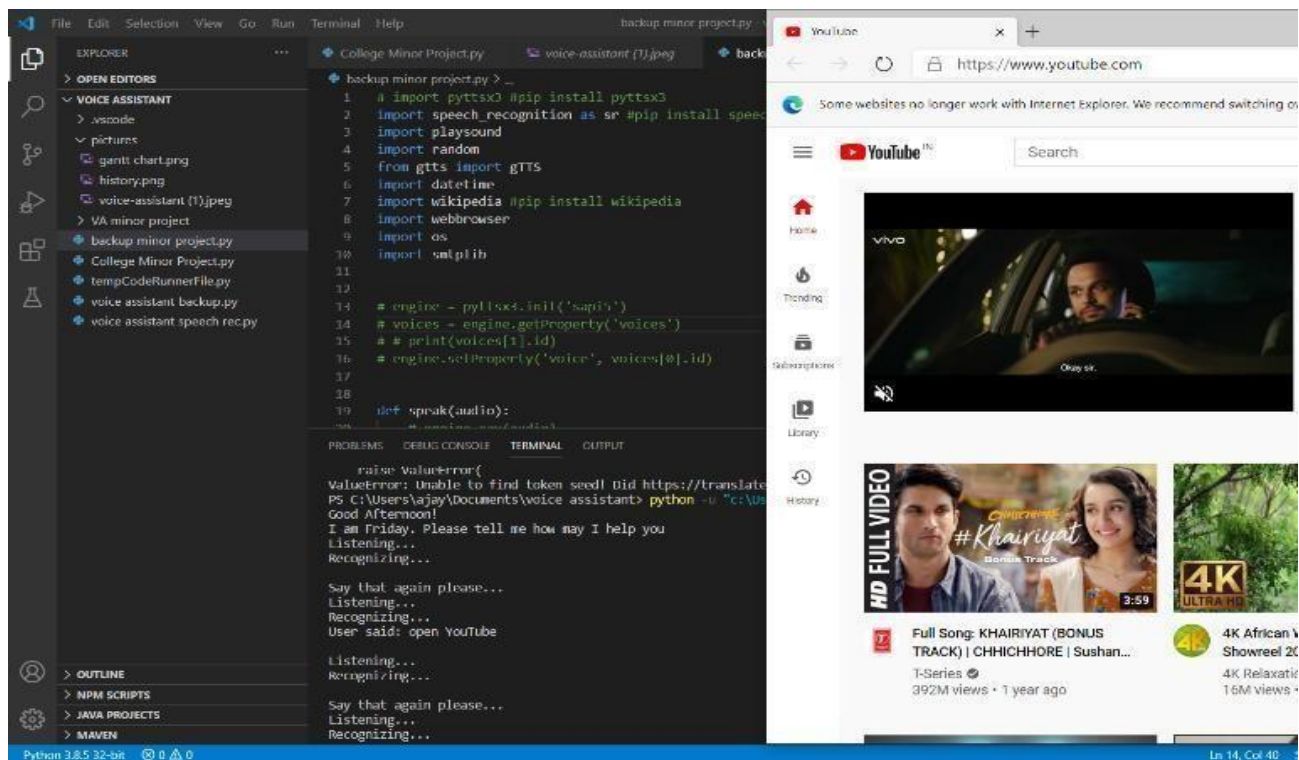


Figure 7.4. Output of code

LIMITATIONS

- The lack of voice command encryption raises concerns about the project's overall security.
- Voices in the background might cause issues.
- Accents can cause misinterpretation, which can lead to inaccurate results.
- Unlike Google Assistant, which can be accessed by saying, "Ok Google! ", TUTOR cannot be accessed externally at any time."

SCOPE FOR FUTURE WORK

- Make TUTOR learn on its own and acquire a new skill.
- Additionally, TUTOR android apps may also be made.
- Increase the number of voice terminals available.
- For further protection, voice instructions may be encrypted.

the severely disabled or those who have suffered minor repetitive stress injuries, i.e., those who may need the assistance of others to manage their surroundings. The use of an IVR system is increasing on a daily basis. Such technologies make it easier for the user to communicate with the computer system, which in turn facilitates the completion of a variety of activities.

The IVR system acts as an intermediary between humans and computers. Due to the time and research constraints, the existing IVR system is only suitable to desktop computers and will not be implemented in real phone devices. This is a disadvantage since the IVR system with Automatic Voice Recognition (AVR) may be used in a wide range of applications. Although the project is still in its infancy, there is plenty of room for improvement in the years to come.

The following are some of the places that may be relevant:

1. Organizational inquiry desk: The system may be utilised in different organisations for simple access to information about the organisation using the voice command.

2. Detection of isolated words is all that the suggested system does, but it might be expanded to include audio to text conversion with additional improvements in algorithms.

3. It is possible to employ voice recognition of Nepali phrases in order to accomplish the work of freshly generated apps and therefore create more user pleasant applications..

4 In embedded systems, voice commands may be used to handle multiple activities using speech recognition technology. This promotes automation of labour and can thus be very advantageous in industrial process automation. process automation

5. Application for People with Disabilities: People with disabilities may also benefit from voice recognition software. It is particularly beneficial for those who are unable to use their hands.

CONCLUSION

We've covered Python-based Personal Virtual Assistants for Windows in this article. Humans' lives are made simpler by virtual assistants. Using a virtual assistant gives you the freedom to contract for just the services you need.. Python is used to create virtual assistants for all Windows versions, much as Alexa, Cortana, Siri, and Google Assistant. Artificial Intelligence is used in this project, and virtual personal assistants are an excellent method to keep track of your calendar. Because of their portability, loyalty, and availability at any moment, virtual personal assistants are more dependable than human personal assistants. Our virtual assistant will get to know you better and be able to provide ideas and follow orders. This device will most likely be with us for the rest of our lives

It is possible to enhance education by using immersive technology.Voice assistants may help students study in new and innovative ways. This article contains studies on the use of AI voice assistants in education. There hasn't been a lot of study done on voice assistants yet, but that's about to change. New discoveries could be made in the future as a result of this study's results. Next years will be all about voice-activated devices like smart speakers and virtual assistants. Exactly how they will be most successful in the classroom is still a mystery. As a result, not all voice assistants are bilingual, and this might be problematic. Additionally, voice assistants lack sufficient security safeguards and protection filters that students may use in the classroom. The use of these devices in the classroom can only be successful if instructors are given the proper training and incentives to do so. Despite the fact that most students and teachers have reported positive results, the data are sparse, fragmentary, and unstructured. More research is required to better understand the use of these devices in the classroom, according to our findings so far.

REFERENCES

- [1] Beirl, D., Yuill, N., & Rogers, Y. (2019). Using Voice Assistant Skills in Family Life. In Lund, K., Niccolai, G. P., Lavoué, E., Gweon, C. H., & Baker, M. (Eds.), *A Wide Lens: Combining Embodied, Enactive, Extended, and Embedded Learning in Collaborative Settings*, 13th International Conference on Computer Supported Collaborative Learning (CSCL) 2019, Volume 1 (pp. 96–103). Lyon, France: International Society of the Learning Sciences. DOI: 10.22318/csc12019.96
- [2] <https://www.activestate.com/blog/how-to-build-a-digital-virtual-assistant-in-python/>
- [3] *Canalys. (2018). Media alert: Smart Speaker Installed Base to Hit 100 Million by End of 2018.* https://www.canalys.com/static/press_release/2018/090718_Media_alert_Smart_speaker_installed_base_to_hit_100_million_by_end_of_2018.pdf
- [4] <https://medium.com/codex/making-your-own-ai-virtual-assistant-with-python-5c2046dadfa7>
- [5] <https://nevonprojects.com/voice-based-intelligent-virtual-assistance-for-windows/>
- [6] *Python Programming-* Kirtan Gurbani
- [7] *Learning Python* - Mark Lutz
- [8] *Designing Personal Assistant Software for Task Management using semantic Web Technologies and Knowledge Databases*
- [9] *Python code for Artificial Intelligence: Foundations of Computational Agents*, David L. Poole and Alan K. Mackworth
- [10] *Chatbot Learning: Everything you need to know about machine learning chatbots* (2020). <https://www.whoson.com/chatbots-ai/chatbot-learning-everything-need-know-machine-learning-chatbots/>
- [11] *How to use an API with Python (Beginner's Guide)*. Retrieved from the link <https://rapidapi.com/blog/how-to-use-an-api-with-python/>
- [12] *Introduction to Machine Learning using Python* (January 2019). Retrieved from <https://www.geeksforgeeks.org/introduction-machine-learning-using-python>
- [13] <https://data-flair.training/blogs/artificial-intelligence-project-ideas/>.
- [14] Beirl, D., Yuill, N., & Rogers, Y. (2019). Using Voice Assistant Skills in Family Life.
- [15] Davie, N., Hilber, T. (2018). *Opportunities and Challenges of Using Amazon Echo in Education*. International Association for Development of the Information Society.
- [16] Davie, N., Hilber, T. (2018). Opportunities and Challenges of Using Amazon Echo in Education. International Association for Development of the Information Society.
- [17] Horn, M. B. (2018). *Hey Alexa, Can You Help Kids Learn More?*. Education Next, 18(2).
- [18] Metz, R., (2014). Voice Recognition for the Internet of Things. <https://www.technologyreview.com/s/531936/voice-recognition-for-the-internet-of-things>

- [19] Griswold, A. (2018). Even Amazon is surprised by how much people love Alexa. (February 2018)<https://qz.com/1197615/even-amazon-is-surprised-by-how-much-people-love-alexa/>
- [20] López, G., Quesada, L., Guerrero, L. A. (2017). Alexa vs. Siri vs. Cortana vs. Google Assistant: a comparison of speech-based natural user interfaces. In *International Conference on Applied Human Factors and Ergonomics* (pp. 241–250). Springer, Cham.
- [21] Algoufi, R. (2016). Using Tablet on Education. *World Journal of Education*, 6(3), 113–119.
- [22] Kowalski, J., Jaskulska, A., Skorupska, K., Abramczuk, K., Biele, C., Kopeć, W., Marasek, K. (2019). Older Adults and Voice Interaction: A Pilot Study with Google Home. *arXiv preprint arXiv:1903.07195*
- [23] Herrington, A., Herrington, J. (2007). Authentic mobile learning in higher education. In *Proceedings of the AARE 2007 International Educational Research Conference*. Fremantle, Western Australia, 1–9. DOI: 10.1109/ICNICONSMCL.2006.103