



Nitte Meenakshi Institute of Technology
An Autonomous Institution under VTU, Belagavi
PB No. 6429, Yelahanka, Bangalore 560-064, Karnataka, India



Project Report

on

“Jarvis: A Voice-Enabled Chat Bot with Multiple Features”

A Dissertation submitted in partial fulfillment of the requirements for the award of degree of

MASTER OF COMPUTER APPLICATIONS

Of

Visvesvaraya Technological University (VTU)



By

Manu B B

1NT21MC052

Under the Guidance of

Mr. Rajeev Arora

Adjunct Professor

Dept of MCA

NMIT, Bengaluru

JULY 2023



Nitte Meenakshi Institute of Technology
An Autonomous Institution under VTU, Belagavi
PB No. 6429, Yelahanka, Bangalore 560-064, Karnataka, India



Project Report

on

“Jarvis: A Voice-Enabled Chat Bot with Multiple Features”

A Dissertation submitted in partial fulfillment of the requirements for the award of degree of

MASTER OF COMPUTER APPLICATIONS

Of

Visvesvaraya Technological University (VTU)



Submitted By

Manu B B

USN: 1NT21MC052

Under the guidance of

Internal Guide

Mr. Rajeev Arora
Adjunct Professor
Department of MCA
NMIT, Bengaluru

External Guide

Mr. Mohammed Azhar Hussain
Mentor & AI Engineer,
AQMENZ Automation Pvt. Ltd
Bengaluru 560032

JULY 2023



Nitte Meenakshi Institute of Technology
An Autonomous Institution under VTU, Belagavi
PB No. 6429, Yelahanka, Bangalore 560-064, Karnataka, India



Department of Master of Computer Applications

CERTIFICATE

*This is to certify that **Manu B B** bearing USN **INT21MC052** has completed her final semester project work entitled “**Jarvis: A Voice-Enabled Chat Bot with Multiple Features**” as a partial fulfilment for the award of Master of Computer Applications degree, during the academic year 2022-2023 under our supervision.*

Signature of Internal Guide

Mr. Rajeev Arora
Adjunct Professor
Dept of MCA NMIT, Bengaluru

Signature of External Guide

Mr. Mohammed Azhar Hussain
Mentor & AI Engineer,
AQMENZ Automation Pvt. Ltd

Signature of HOD

Dr. Sreekanth R.
Prof & Head, Dept. of MCA
NMIT, Bengaluru

Signature of Principal

Dr. H.C. Nagaraj
NMIT, Bengaluru

Submitted to the Department of Master of Computer Applications, Nitte Meenakshi Institute of Technology, for the Viva Voce held on _____

INTERNAL EXAMINER

EXTERNAL EXAMINER

COMPANY CERTIFICATE



Aqmenz Automation
Private Limited

CIN: U74999KA2018PTC117612



CERTIFICATE OF APPRECIATION

THIS CERTIFICATE IS PROUDLY PRESENTED TO

MANU B B

1NT21MC052

On behalf of **Aqmenz Automation Pvt. Ltd.**, we proudly provide the **Certificate of Appreciation**.

The Certificate recognises the outstanding & remarkable performance of the student in the completion of Project titled

JARVIS: A VOICE-ENABLED CHAT BOT WITH MULTIPLE FEATURES

CERTIFICATE ISSUE NO: IN / **AC0524** / FY2023

Mr Mohammed Azhar Hussain
Chief Technology Officer (CTO),
AQMENZ Automation Pvt. Ltd.,
Bengaluru - 560032

Scan to Verify



Mr Mohan V S
Chief Executive Officer (CEO),
AQMENZ Automation Pvt. Ltd.,
Bengaluru - 560032

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without mentioning the people who made it possible. With deep gratitude, I acknowledge all those whose guidance and encouragement served as a beacon of light and crowned my efforts with success. I thank each one of them for their valuable support.

I would like to thank **AQMENZ Automation Pvt. Ltd** for providing me an opportunity to work as an Intern and **Mr. Mohammed Azhar Hussain** for his timely support and guidance during my Internship period.

I express my thanks to, **Dr. H. C. Nagaraj, Principal, NMIT, Bengaluru**, for all the support and encouragement during my Post Graduate Studies.

I express my sincere thanks to **Dr. Sreekanth R., Professor & HOD-MCA, NMIT, Bengaluru**, whose constant encouragement, guidance and support that has helped me complete the project work successfully on time.

I express my sincere thanks to my Internal Guide, **Mr. Rajeev Arora, Adjunct Professor, Department of MCA, NMIT, Bengaluru**, whose constant encouragement, guidance and support that has helped me complete the project work successfully on time.

My special thanks to all the teaching faculty members for providing me invaluable support and guidance. Finally, I thank my Parents and Friends who have helped me in all possible ways for the success of this project work.

Name: **Manu B B**

USN: **1NT21MC052**

DECLARATION

I, **Manu B B**, student of IV Semester of MCA, **Nitte Meenakshi Institute of Technology**, Bengaluru, bearing USN **1NT21MC052**, hereby declare that the that the project entitled “**Jarvis: A Voice-Enabled Chat Bot with Multiple Features**” has been carried out by me under the supervision of External Guide, **Mr. Mohammed Azhar Hussain, Mentor & AI Engineer**, and Internal Guide, **Mr. Rajeev Arora, Adjunct Professor, NMIT** and submitted in partial fulfillment of the requirements for the award of the Degree of **Master of Computer Applications** by the **Visvesvaraya Technological University** during the academic year **2022 - 2023**. This report has not been submitted to some other Organization/University for any award of degree or certificate.

Place: Bangalore

Date:

Signature of the Student

Manu B B

1NT21MC052

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.

TABLE OF CONTENTS

Sl. No.	Contents	Page No.
	Acknowledgement	i
	Declaration	ii
	Abstract	iii
	Table of Contents	iv
	List of Figures	v
1	Introduction	1
	1.1 Statement of the Problem	1
	1.2 Objectives	3
	1.3 Scope	3
	1.4 Methodology	5
2	Literature Review	8
	3.1 Existing system	9
	3.2Proposed System	9
3	System Design	11
	3.1 speech to text interface	11
	3.2 data traffic	13
	3.3 Architecting a computer system	14
4	Software Requirements Specification	15
	4.1 Specification of program	15
	4.2Guide to python python files	16
5	Implementation	17
	7.1Sample Code	19
	7.2Screen Shots	23
6	Result And Outcome	27
	6.1 Outcome and summary	28
	6.2 Limitations	29
7	Conclusion	30
8	Appendix A Bibliography	31
	Appendix B Plagiarism Check	32

LIST OF FIGURES

Figure No.	Description	Page No.
1.4.1	Data flow Diagram	05
1.4.2	input/output text to speech	06
1.4.3	process of working of ASR	07
3.1	signals data flows	12
3.2	Tutor data flow	13
3.3	use case diagram	14
6.1.1	voice assistance work	28

Chapter 1

INTRODUCTION

1.1 PROBLEM STATEMENT:

The project "Jarvis Chatbot with Voice Chat Feature" is an artificial intelligence- based project aimed at creating a virtual assistant that can perform multiple tasks on voice command. The project is named after the famous virtual assistant in the Iron Man movie series, Jarvis, which was capable of performing various tasks for the superhero. The Jarvis Chatbot is developed using Python and integrates various APIs for performing different functionalities. In today's fast-paced world, people are always looking for ways to automate their daily tasks and make their lives easier. The need for virtual assistants has grown with the rise in the use of technology. The main challenge in creating a virtual assistant is to make it efficient in handling various tasks while being user- friendly. The problem statement for this project is to develop a virtual assistant that can perform multiple tasks on voice command, making it easy for users to interact with the technologies

INTRODUCTION TO THE PROJECT

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, Canals (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

I used Visual Studio Code to construct this project, and all of the py files were produced in VSCode, and also we are used **ASR(Automatic Speech Recognition) algorithm** for converting human speech into text. And also we are **not using NLP(natural language processing)** because this is the **feature based bot not translation based bot**. 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, which is a feature based bot. 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.

1.2 Objectives

The main objectives of this project are:

1. To develop an efficient and user-friendly virtual assistant that can perform multiple tasks on voice command.
2. To integrate various APIs and libraries to enable the virtual assistant to perform different functionalities such as youtube search, googlesearch, set alarms, download videos, and do basic math and science.
3. To develop an automatic message sending feature on WhatsApp that can send messages on a timed or delayed basis.

1.3 Scope:

The scope of the project is vast, as it includes multiple functionalities that a user can perform using voice commands. The virtual assistant will be capable of performing tasks such as searching on Google and YouTube, setting alarms, downloading videos, and doing basic math and science. The project's scope also includes the automatic sending of messages on WhatsApp, which can be used for sending timed or delayed messages.

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.

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.

,

1.4 METHODOLOGY

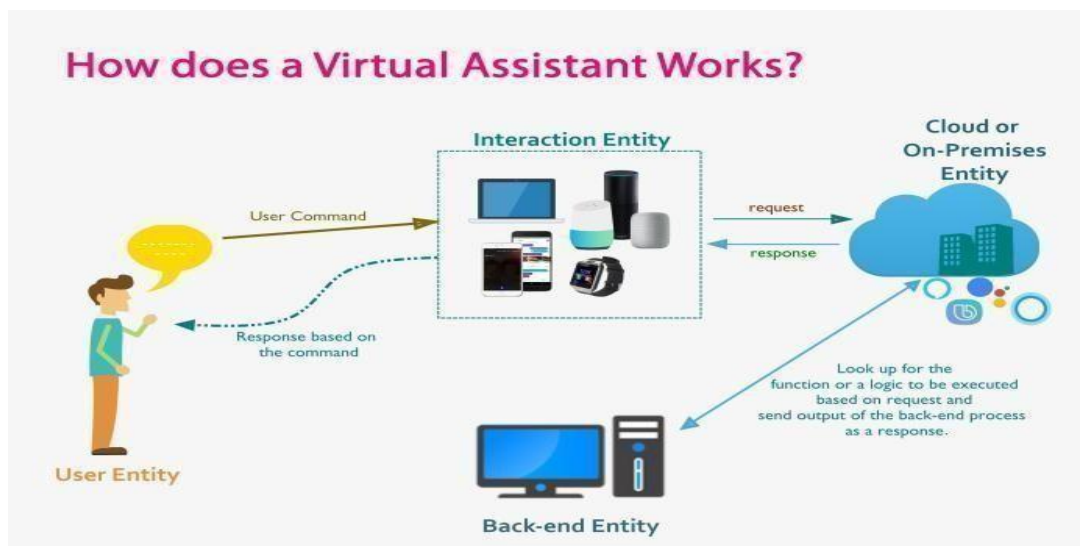


Figure 1.4.1 Shows backend working data flow

User input may be matched with executable instructions using ASR. 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. Virtual Assistant 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.

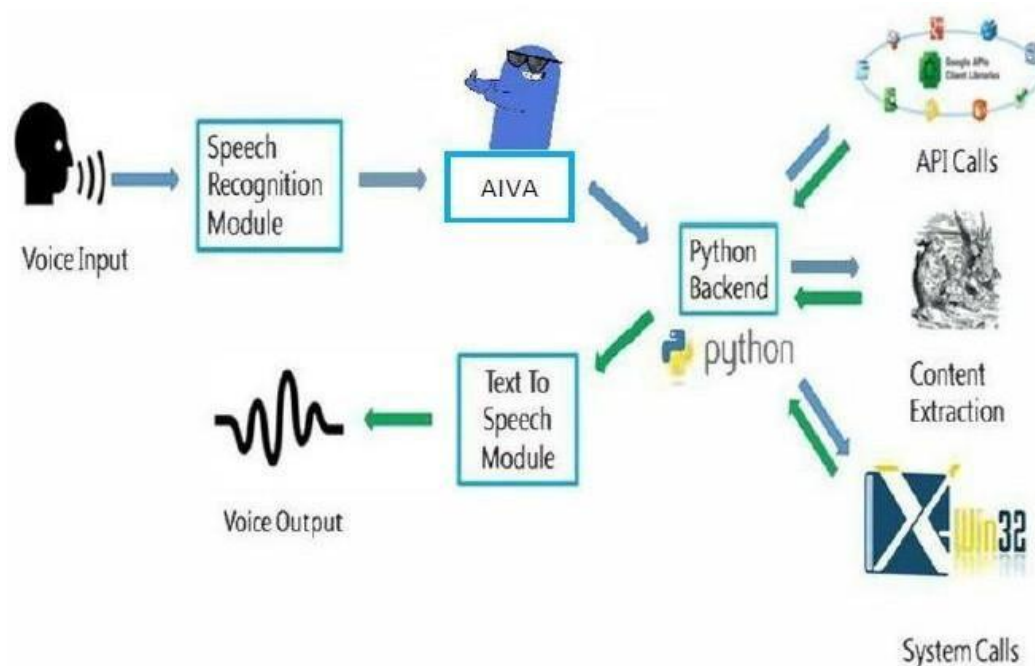


Figure 1.4.2 input/output text to speech

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 1.1 shows the diagram of the proposed multi-domain ASR framework

Developing Speech recognition system:

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.

Developing Backend using 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.

Developing the Automatic Speech Recognition (ASR) :

ASR is a technology that uses machine learning (ML) and artificial intelligence (AI) to convert human speech into text. It's a common technology that many of us encounter every day – think Siri, Okay Google or any speech dictation software.

Working: Most ASR voice technology begins with an acoustic model to represent the relationship between audio signals and the basic building blocks of words. Just like a digital thermometer converts an analog temperature reading into numeric data, an acoustic model transforms sound waves into bits that a computer can use. From there, language and pronunciation models take that data, apply computational linguistics, and consider each sound in sequence and in context to form words and sentences, **fig 1.3 shows the working of ASR**

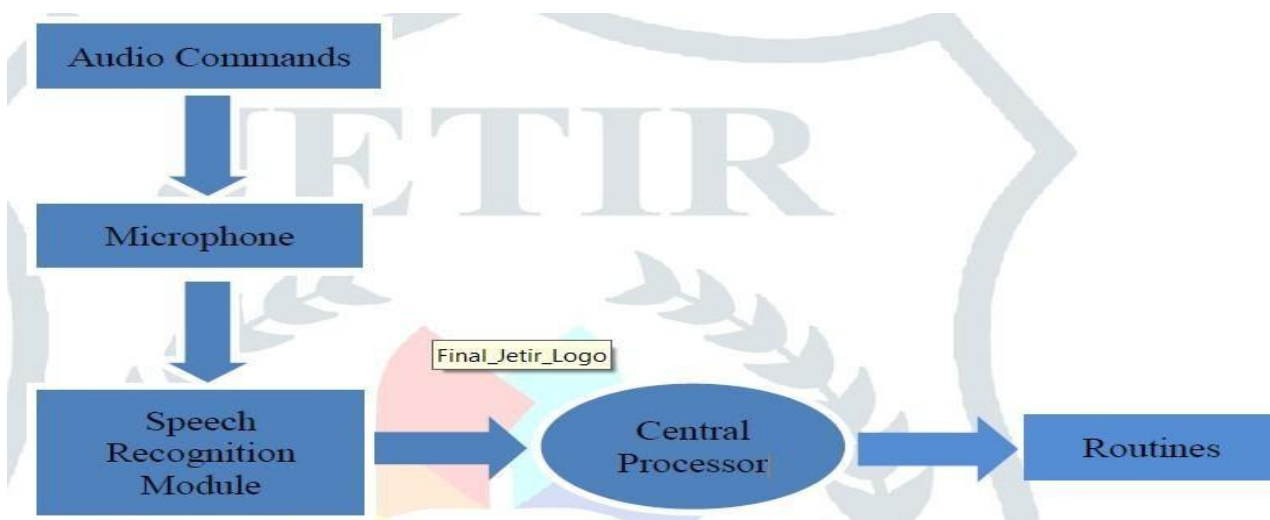


Figure. 1.4.3 Shows the process of working OF ASR

CHAPTER-2

LITERATURE SURVEY

1950's and 60's: The first speech recognition systems focused on numbers rather than words. In 1952, Bell Labs developed the "Audrey" system, which could recognize a single voice saying a number aloud. 10 Years Later, IBM Introduces a 'Shoebox' That Understands and Responds Up to 16 words in English. Other countries around the world have developed hardware that can recognize sounds and voices. And by the late '60s, technology could support words containing his four vowels and his nine consonants.

1970s : Speech recognition has made several important advances in the last decade. This is largely thanks to the US Department of Defense and his DARPA. Their Speech Understanding Research (SUR) program was one of the largest of its kind in the history of speech recognition. Carnegie Mellon's "Harpy" speech system evolved from this program and was able to understand his over 1,000 words, roughly equivalent to the vocabulary of his 3-year-old. Also important in the 1970s, Bell Labs introduced system that could interpret multiple voices could that could interpret multiple voices.

1980s: In the 1980s, speech recognition vocabularies grew from hundreds to thousands of words. One breakthrough came from a statistical technique called Hidden Markov Models (HMMs). Rather than simply using words to look for patterns of sounds, HMMs estimated the probability that an unknown sound is actually a word.

1990s: In the 1990s, speech recognition was primarily driven by personal computers. Faster processors have made software like Dragon Dictate more widely available. BellSouth introduced Voice Portal (VAL), an interactive dial-in speech recognition system. This system spawned countless phone tree systems that still exist today.

2000s: By 2001, voice recognition technology had reached his nearly 80% penetration rate. Not much progress was made for most of the decade until Google introduced his Google Voice Search. Since voice recognition was an app, it was in the hands of millions of people. This was also important because computing power could be moved to the data center. Additionally, Google has gathered data from billions of searches that can help predict what people are actually saying. At the time, Google's English search system contained his 230 billion words derived from user searches.

2010s: In 2011, Apple launched Siri, which is similar to Google's voice search. This early decade saw an explosion of other speech recognition apps. And we've seen consumers find it easier and easier to converse with machines thanks to Amazon's Alexa and Google Home.

Today, some of the biggest tech companies vie for the title of "linguistic accuracy." In 2016, IBM achieved a word error rate of 6.9 percent. In 2017, Microsoft usurped IBM, claiming a 5.9 percent stake. Shortly thereafter, IBM improved this rate to his 5.5%. However, Google's value is the lowest for him at 4.9%.

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

3.1 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. It 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.

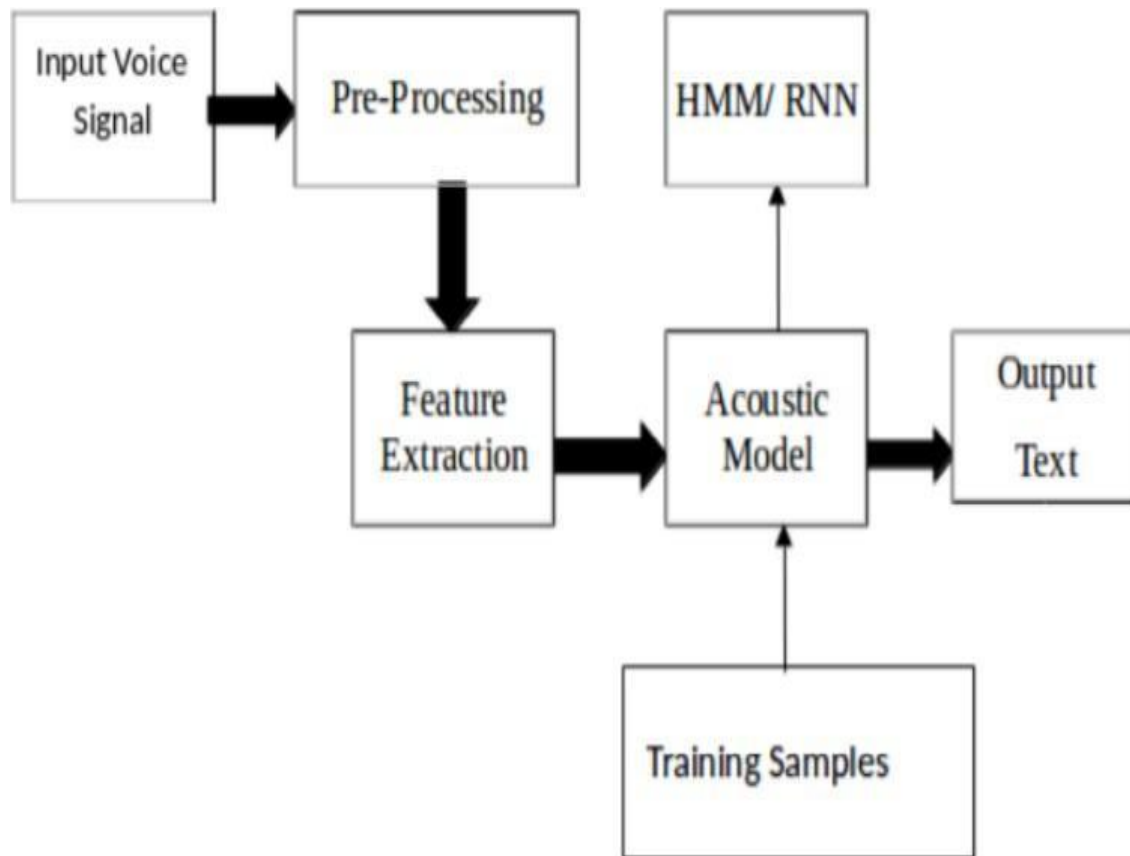


Figure 3.1 Shows signals data flows

3.2 DATA TRAFFIC.

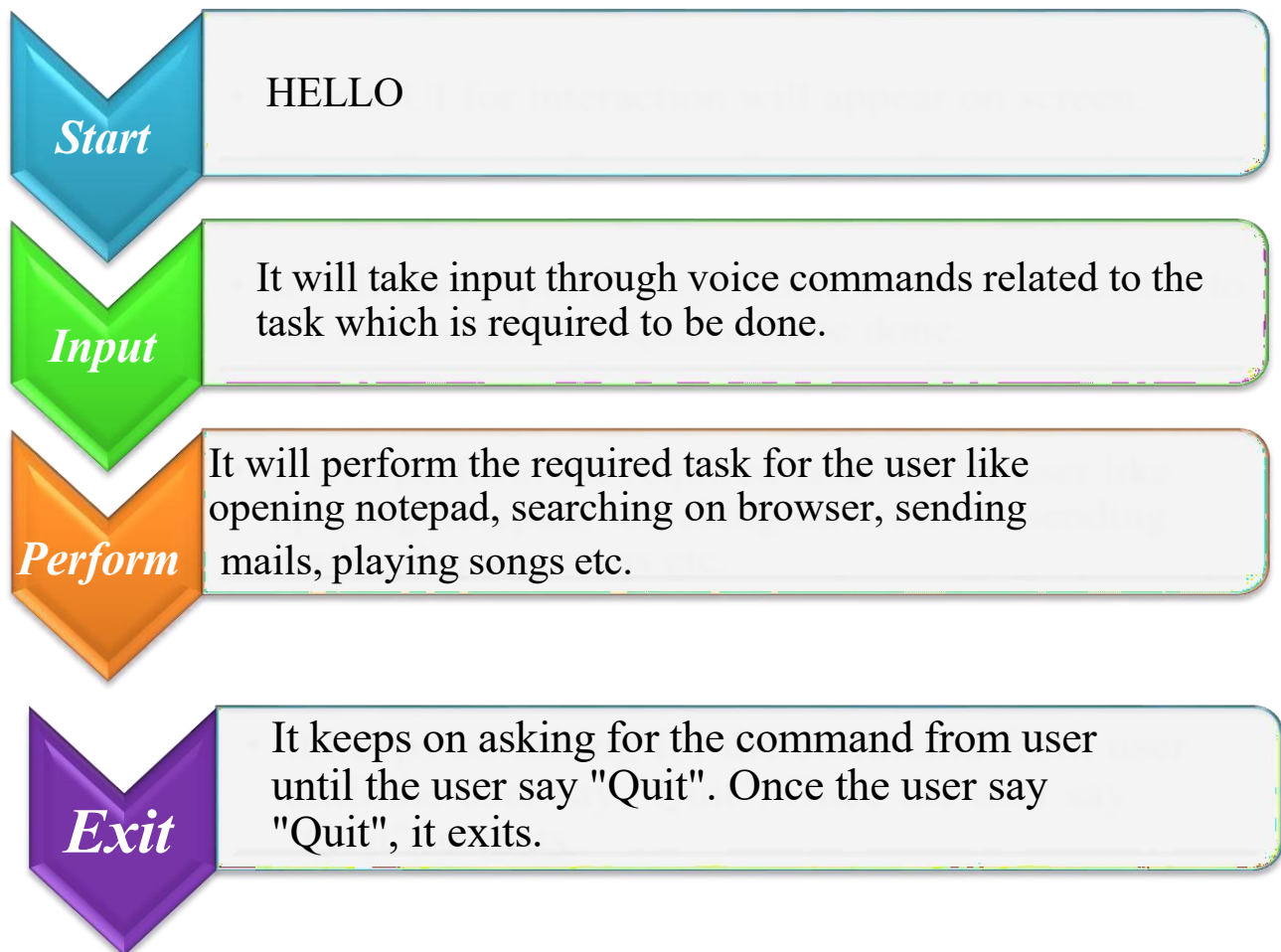


Figure 3.2 shows Tutor data flow

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 pythont. 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.

3.3 ARCHITECTING A COMPUTER SYSTEM

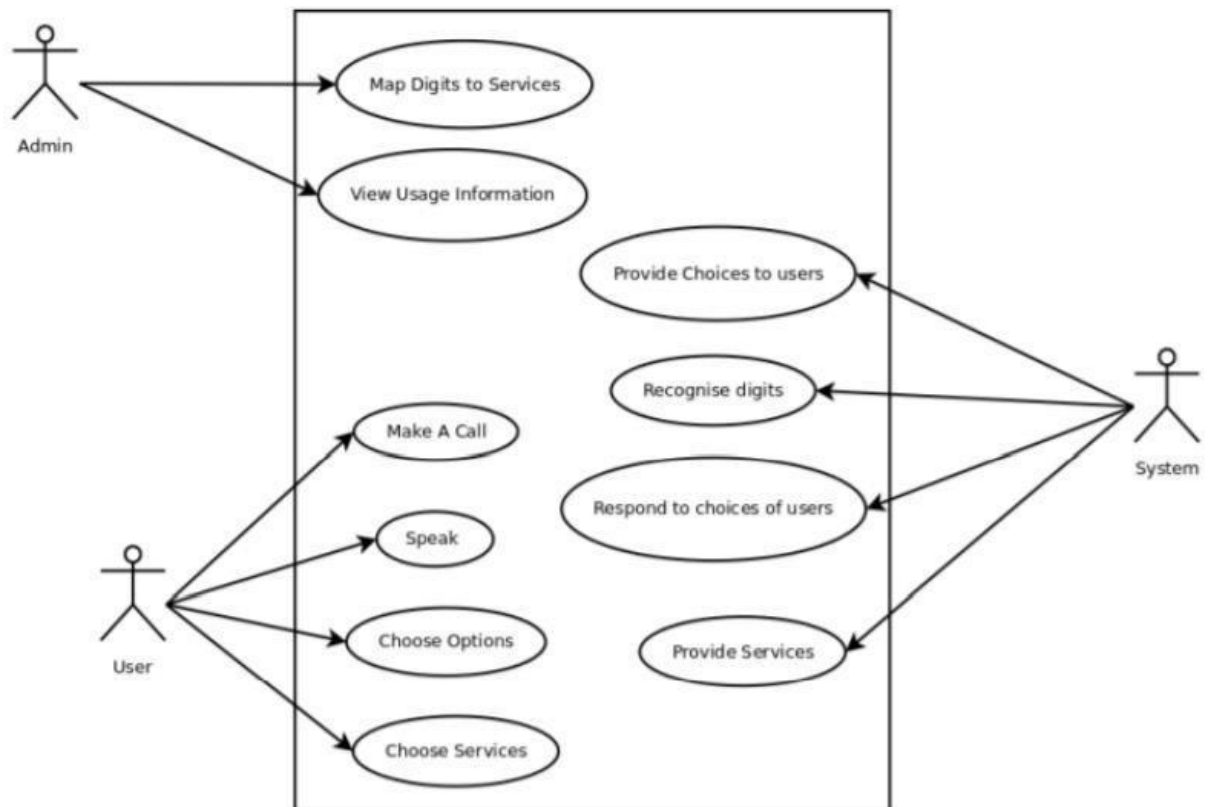


Figure 3.3 shows use case diagram

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 pytsx3, 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 frameworkslike 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

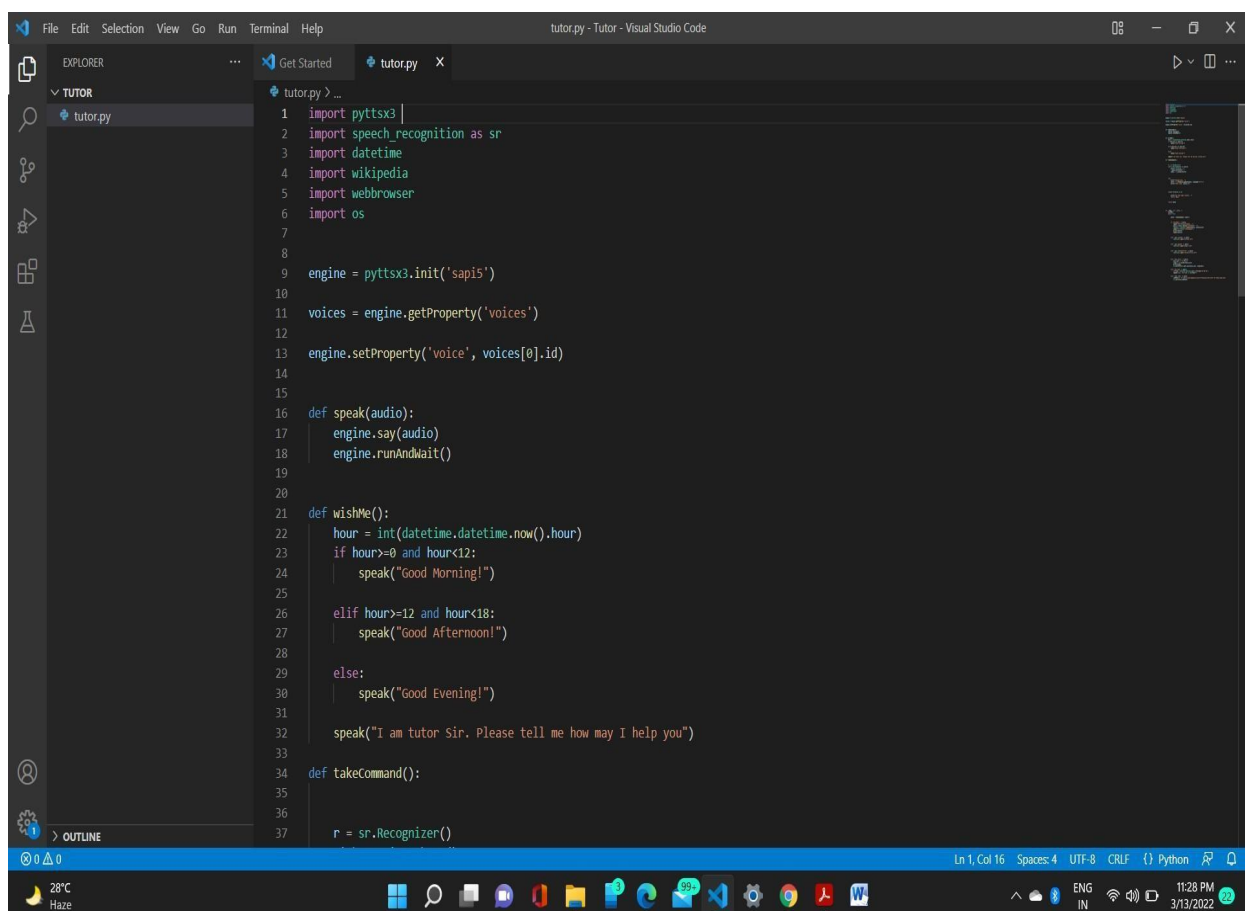
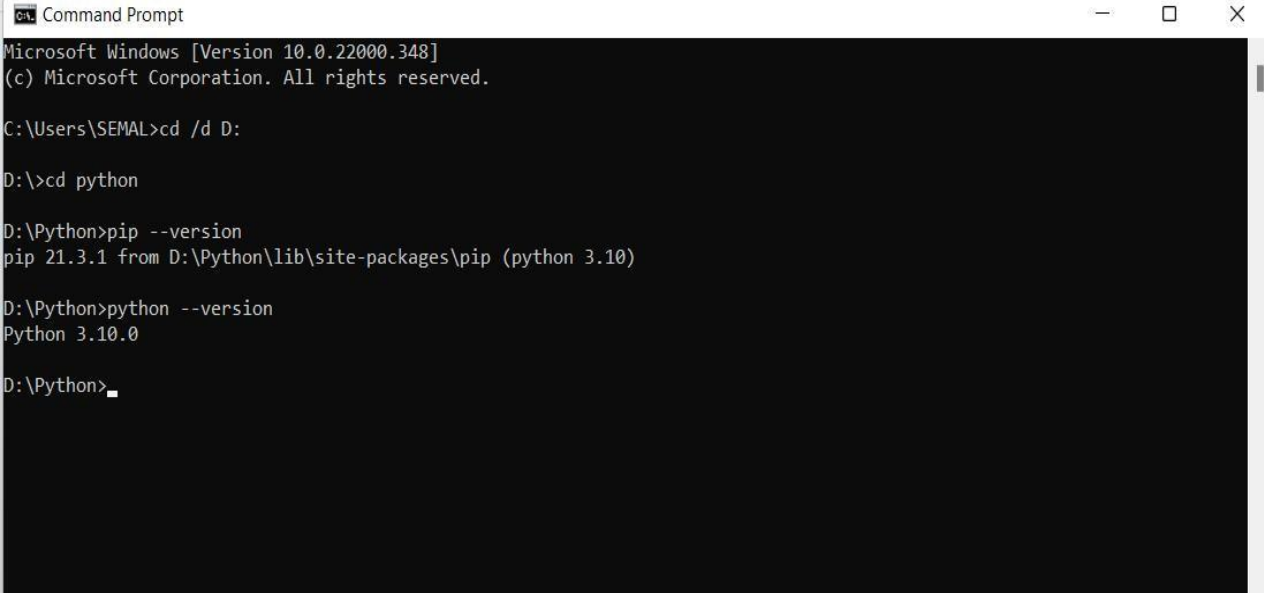


Figure 4.1. Shows the Implementation of library

GUIDE TO PYTHON FILES

This Python library was used in JARVIS:

- 1.) **Pytttsx3**: A python library that translates text into voice.
- 2.) **speechRecognition** Python package SpeechRecognition translates voice into text.
- 3.) **Datetime**: We can get the current date and time using this library
- 4.) **Wikipedia** A python module for searching Wikipedia is provided.
- 5.) **Smtplib** 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.



```
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

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.

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.

DATA IMPLEMENTATION AND PROGRAM EXECUTION

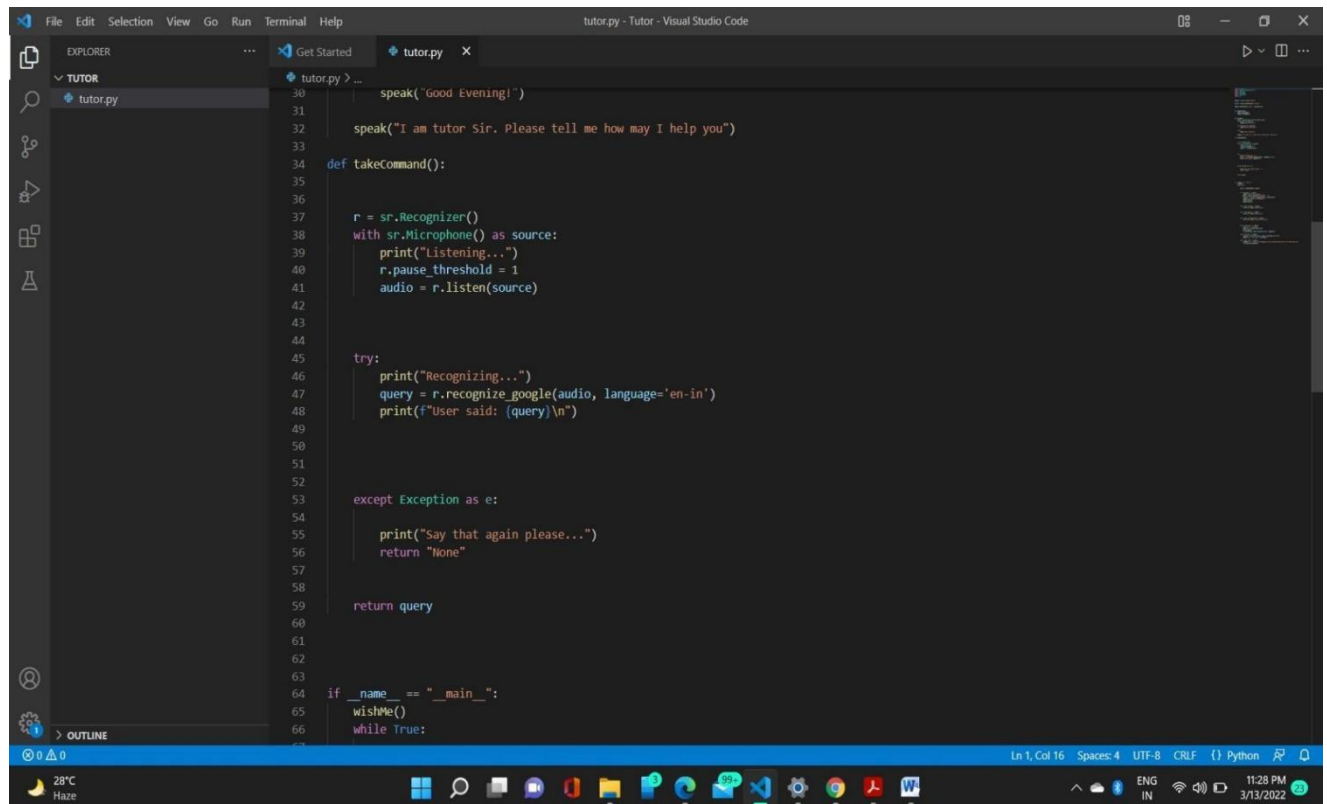
Installing all of the required packages and libraries is a good place to start. Installing the libraries is as simple as running "pip install" and then importing the results. The following components are included in the set:FUNCTIONS

- 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.

5.1 Sample Code



The screenshot displays the Visual Studio Code interface with a Python file named 'tutor.py' open. The code implements a chatbot named 'TUTOR' that can speak and listen. It uses the 'speak' function to output text and the 'sr.Recognizer' and 'sr.Microphone' classes for voice interaction. The 'takeCommand' function handles the logic for listening and recognizing speech. The code is as follows:

```
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     return query
59
60
61
62
63
64 if __name__ == "__main__":
65     wishMe()
66     while True:
```

Figure 5.1.1 Shows the screenshot of the code

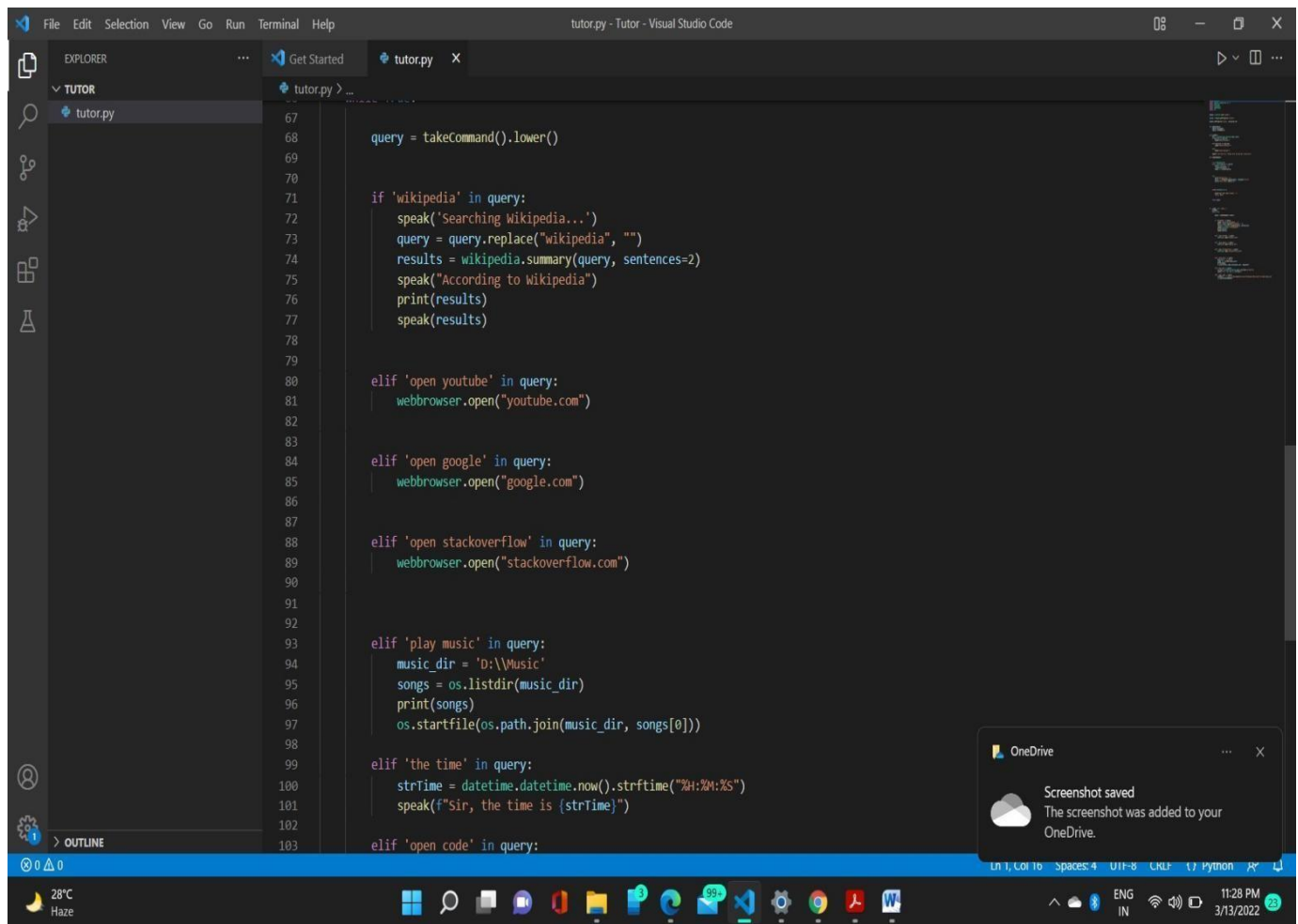
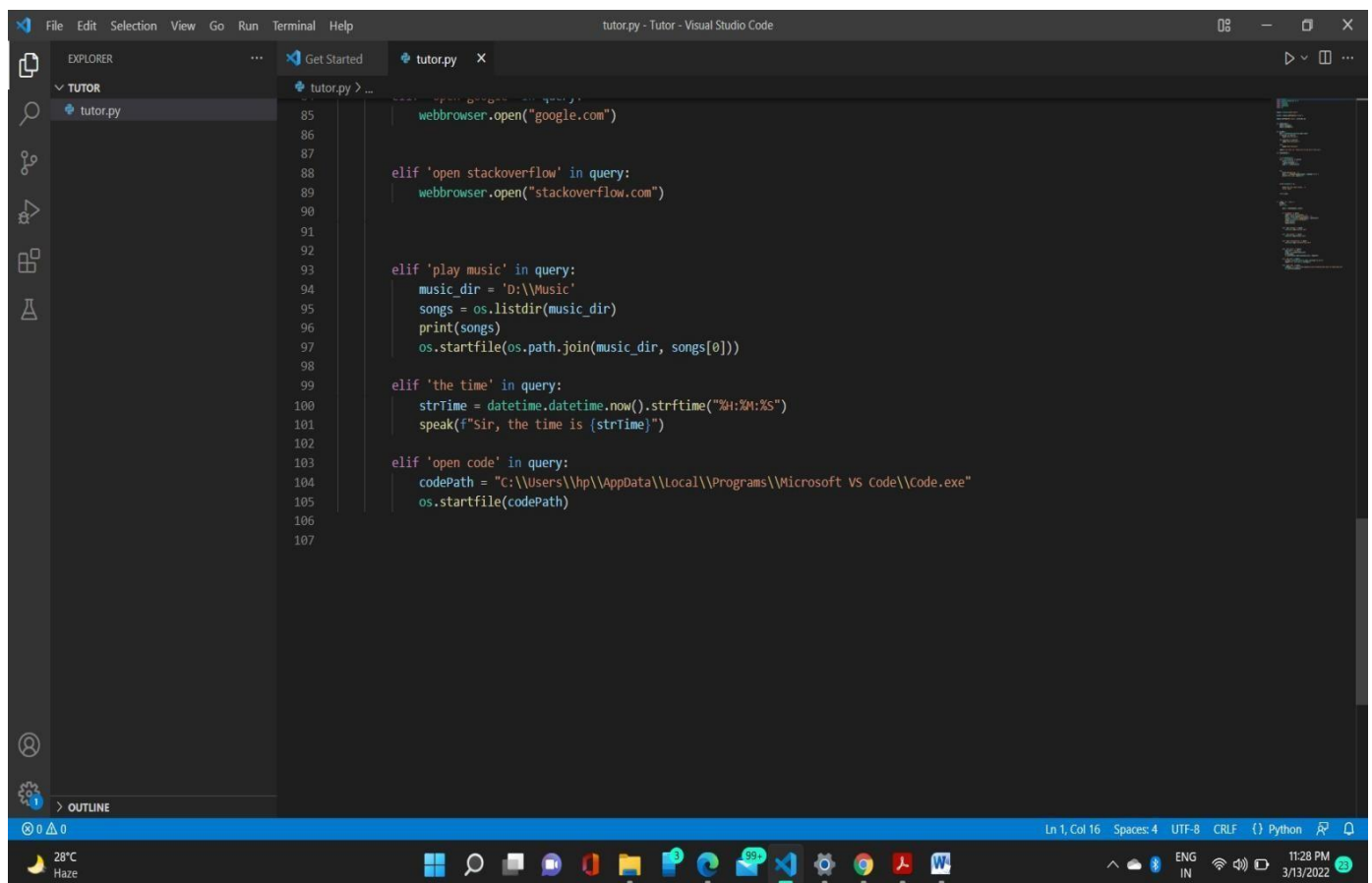


Figure 5.1..2. Shows the screenshot of the code

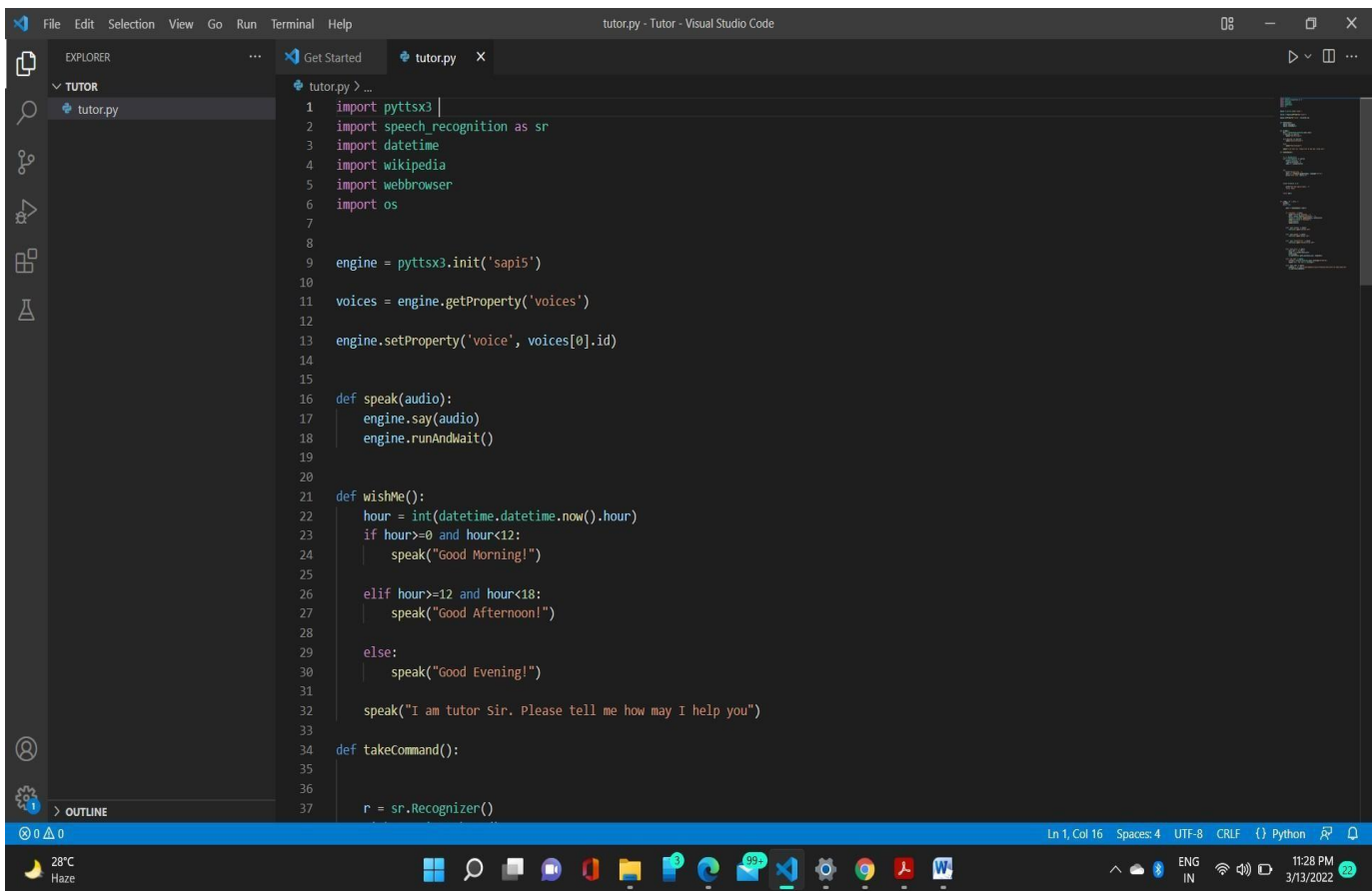


The screenshot shows the Visual Studio Code editor with a Python file named `tutor.py`. The code is a chatbot script that handles various queries. The Explorer sidebar on the left shows the file structure with `TUTOR` and `tutor.py`. The main editor area displays the following code:

```
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
```

The status bar at the bottom indicates the current position is Line 1, Column 16, with 4 spaces, UTF-8 encoding, CRLF line endings, and Python syntax highlighting. The system tray shows the date and time as 11:28 PM on 3/13/2022.

Figure 5.1.3 Shows the screenshot of the code



The screenshot displays the Visual Studio Code editor interface with a Python file named `tutor.py` open. The code implements a chatbot named 'TUTOR' with voice interaction capabilities. It uses the `pyttsx3` library for text-to-speech and `speech_recognition` for voice input. The chatbot can greet the user based on the time of day and respond to a specific command.

```
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 voices = engine.getProperty('voices')
11 engine.setProperty('voice', voices[0].id)
12
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
35 def takeCommand():
36
37     r = sr.Recognizer()
```

Figure 5.1.4 Shows the screenshot of the code

5.2 INPUT/OUTPUT SCREENSHOT

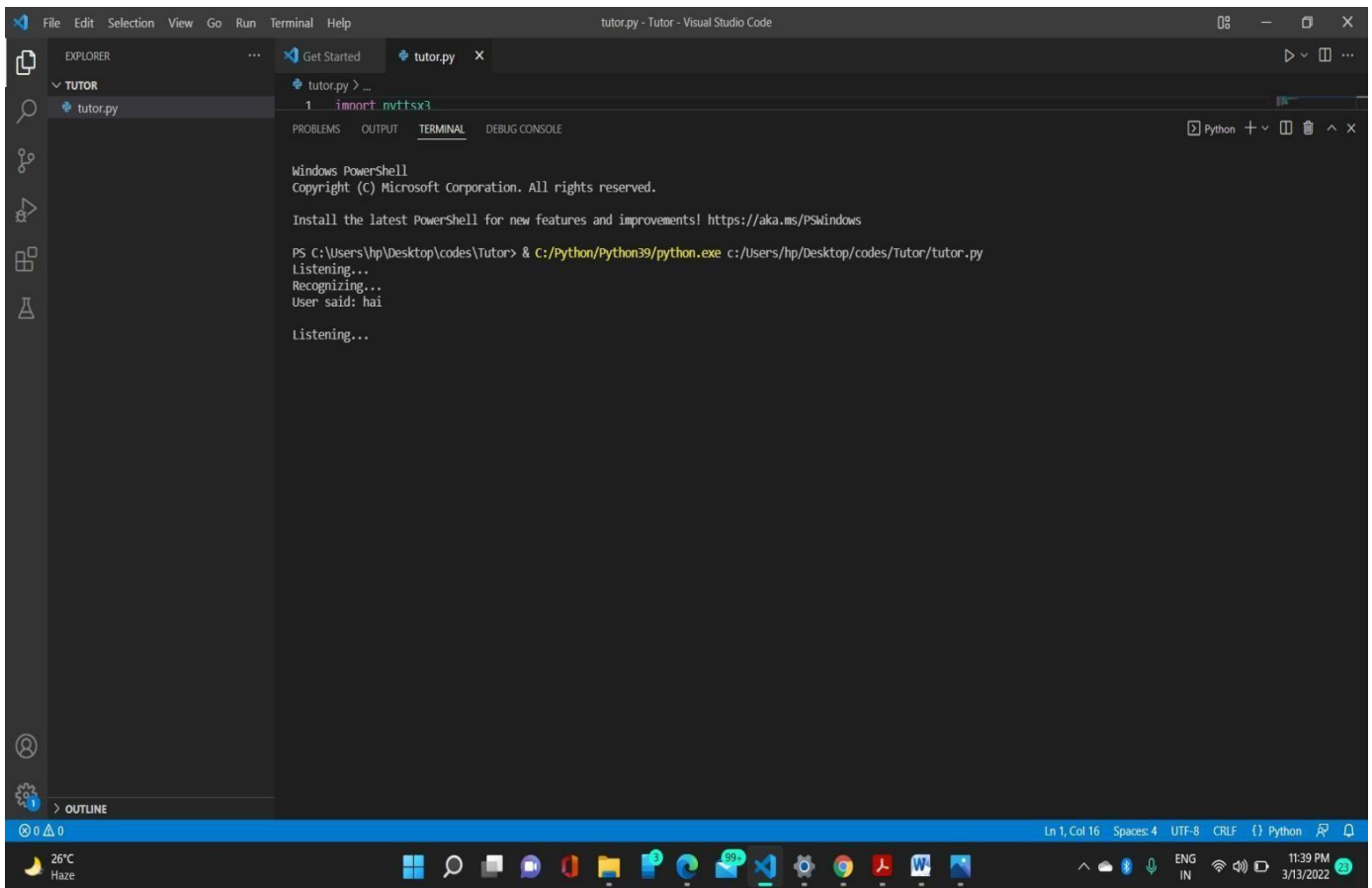


Figure 5.2..1. Shows the screenshot of the output

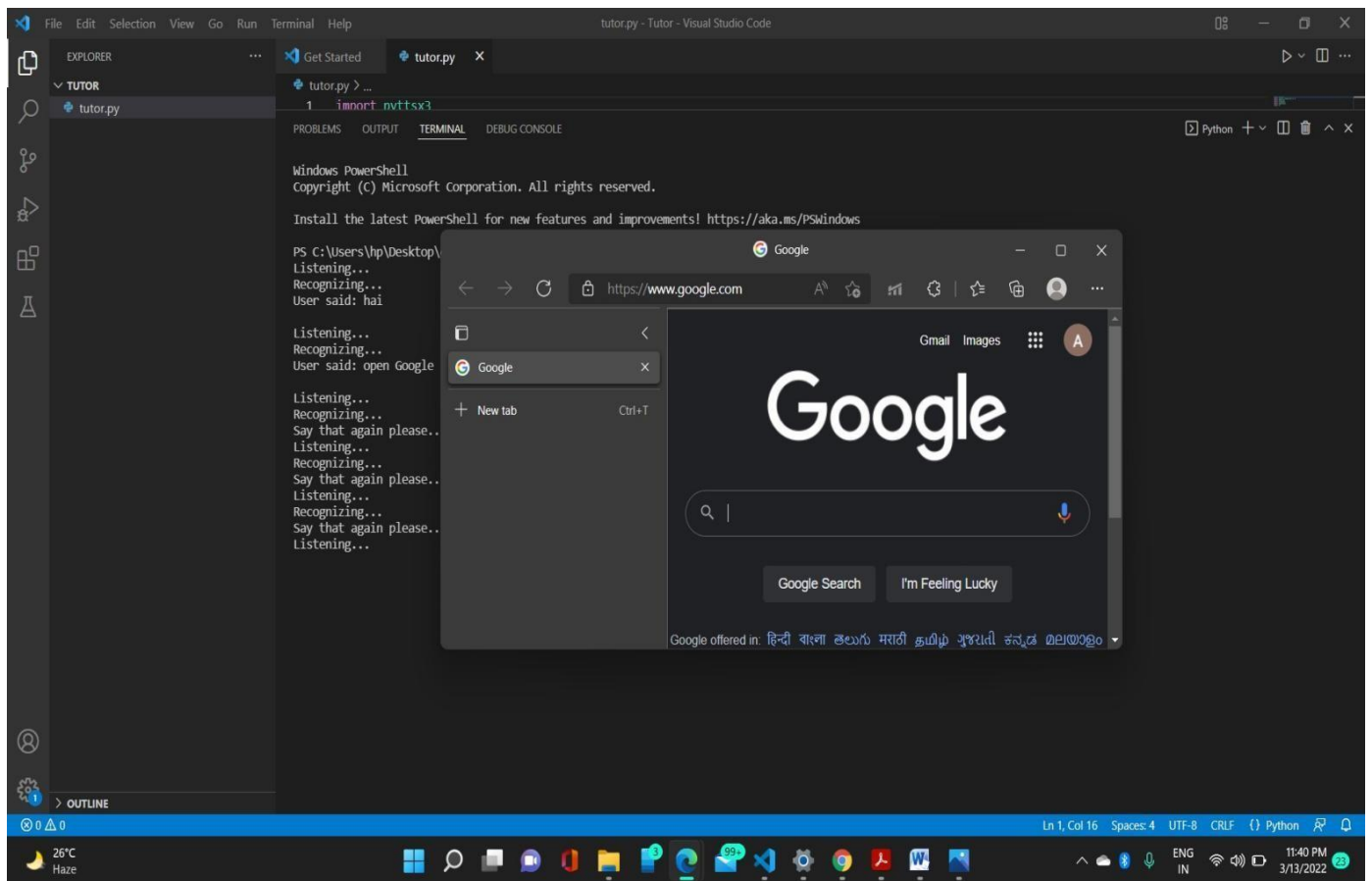


Figure 5.2.2 Shows the output of the code “hello google”

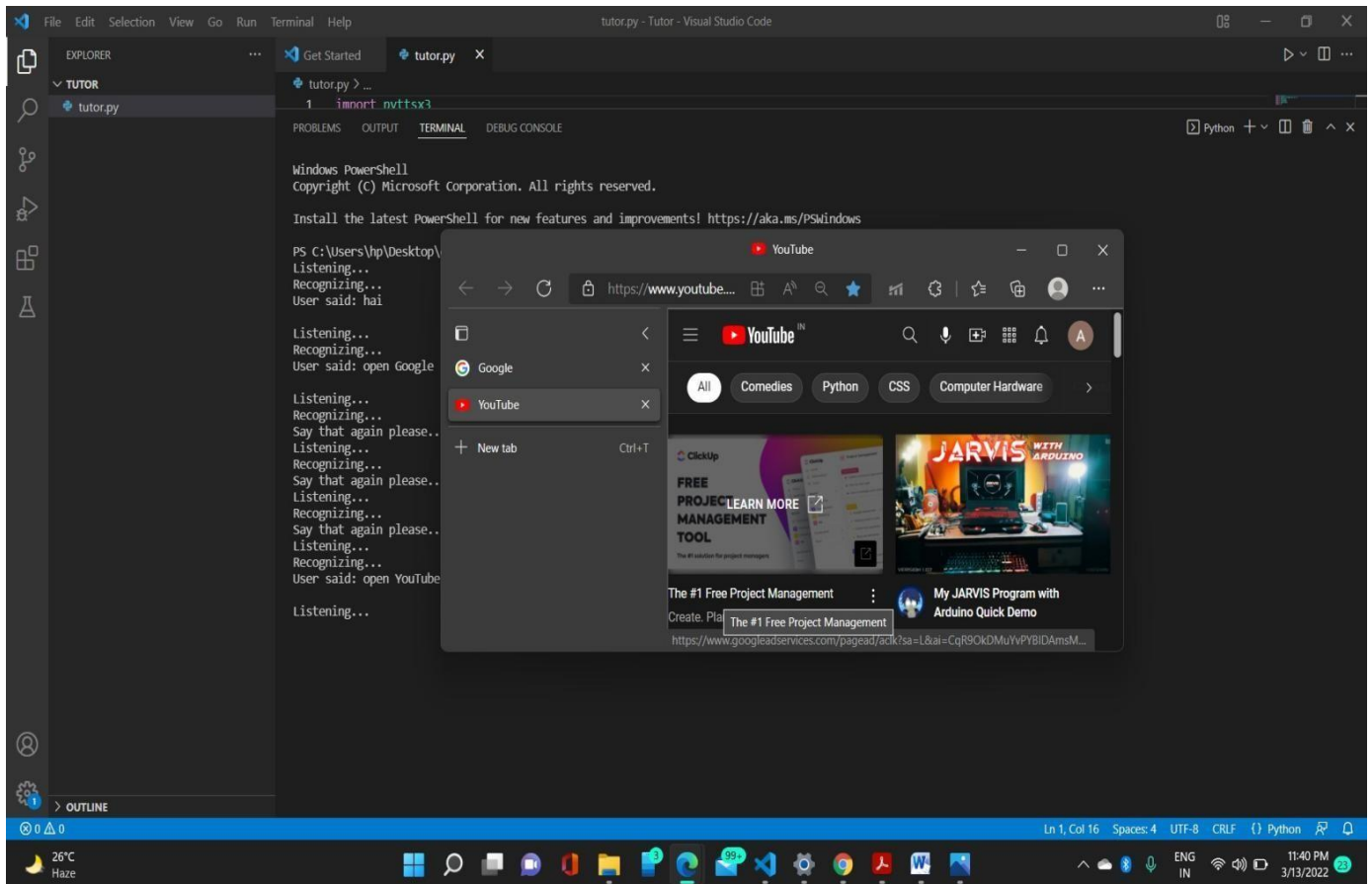


Figure 5.2.3 Shows the output of the code “open youtube”

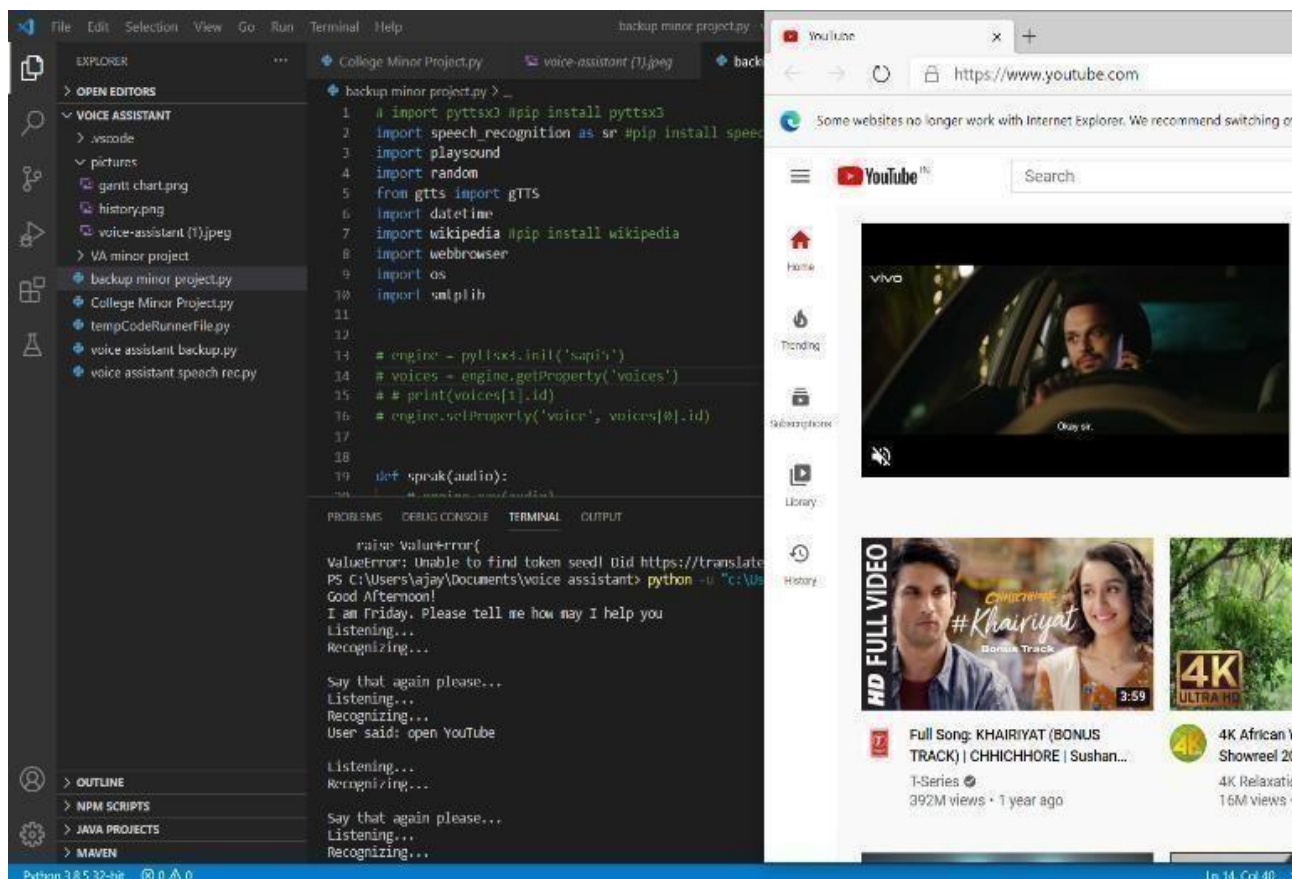


Figure 5.2.4. Output of code

CHAPTER-6

Result AND Outcome

Using a virtual assistant saves time. When a customer gives a job to a virtual assistant, the programme takes over and completes it. Using ASR, 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.

6.1 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

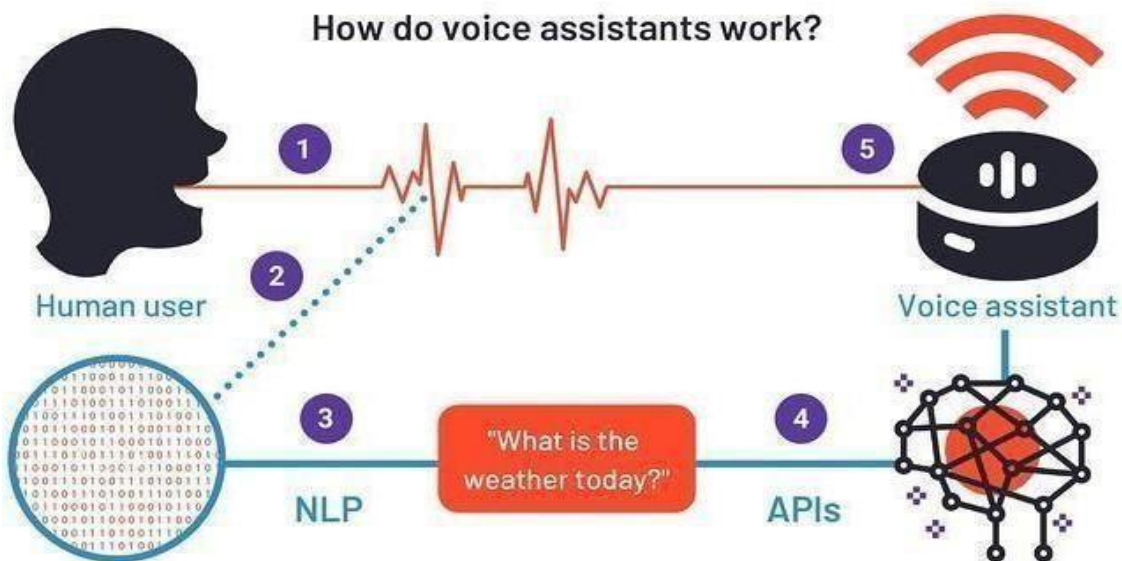


Figure 6.1.1 shows voice assistance work

Summary

In this research, we created a voice assistant that can carry out any job provided by the user 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..

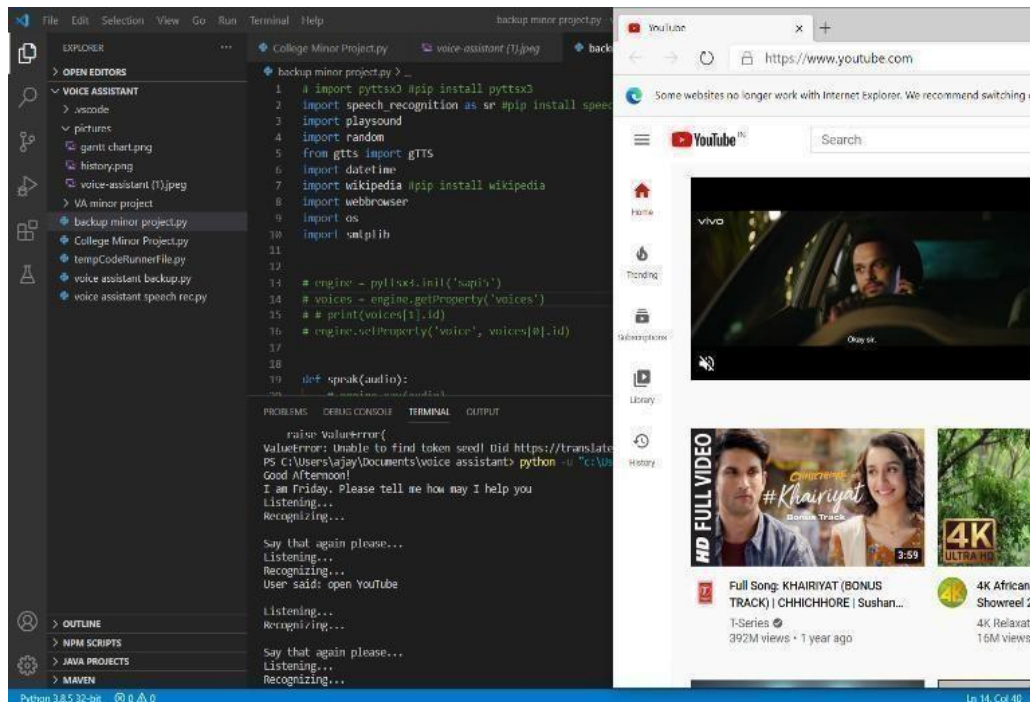


figure 6.1.2 output of command

6.2 LIMITATIONS

1. The lack of voice command encryption raises concerns about the project's overall security.
2. Voices in the background might cause issues.
3. 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."

CHAPTER 7

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

Bibliography

- [1] Beirl, D., Yuill, N., & Rogers, Y. (2019). Using Voice Assistant Skills in Family Life. In Lund K. Niccolai . P. Lavou . weon C. . aker M. ds. A ide 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/cscl2019.96
- [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](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 ArtificialIntelligence:Foundations of Computational Agents ,DavidL. PooleandAlanK.Mackwort
- [10] Chatbot Learning: Everything you need to know about machine learning chatbots (2020). <https://www.whoson.com/chatbots-ai/chatbot-learning-everything-need-knowmachine-learningchatbots/>
- [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>

Submission Information

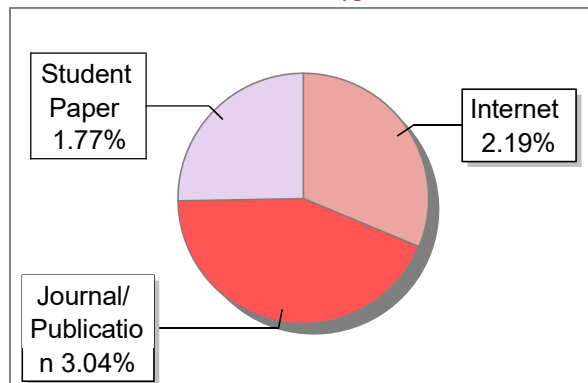
Author Name	Manu BB
Title	Manu BB
Paper/Submission ID	804392
Submission Date	2023-07-08 15:07:37
Total Pages	31
Document type	Project Work

Result Information

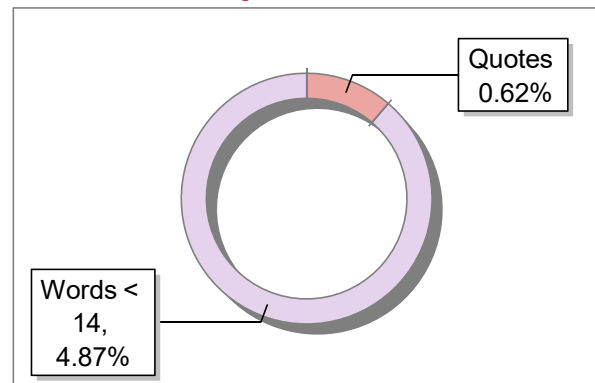
Similarity **7 %**



Sources Type



Report Content



Exclude Information

Quotes	Excluded
References/Bibliography	Excluded
Sources: Less than 14 Words Similarity	Excluded
Excluded Source	0 %
Excluded Phrases	Excluded

A Unique QR Code use to View/Download/Share Pdf File





DrillBit Similarity Report

7

SIMILARITY %

13

MATCHED SOURCES

A

GRADE

A-Satisfactory (0-10%)

B-Upgrade (11-40%)

C-Poor (41-60%)

D-Unacceptable (61-100%)

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	REPOSITORY - Submitted to Bannari Amman Institute of Technology on 2023-01-27 12-28	1	Student Paper
2	ijirce.com	1	Publication
3	repository.futminna.edu.ng 8080	1	Publication
4	s-space.snu.ac.kr	1	Publication
5	artsdocbox.com	<1	Internet Data
6	Family Vacation 2050 Socially and Technologically-Driven Scenarios of the Futur by Whittington-2014	<1	Publication
7	www.nap.edu	<1	Internet Data
8	PAPER SUBMITTED TO AMITY UNIVERSITY, YEAR 2020	<1	Student Paper
9	International Conference on Innovative Computing and Communications Proceedings by Deepa-2021	<1	Publication
10	www.slideshare.net	<1	Internet Data
11	artsdocbox.com	<1	Internet Data
12	religiondocbox.com	<1	Internet Data

EXCLUDED PHRASES

1 nitte meenakshi institute of technology

