R.M.K. ENGINEERING COLLEGE

(An Autonomous Institution)

R.S.M. Nagar, Kavaraipettai-601 206

Department of Computer Science and Engineering

22ME311 - PRODUCT DEVELOPMENT LAB-3

FRIDAY

(FRiendly Intelligent Desktop Assistant for You)

Submitted by

Divyabharathi L	(111722102031)
Haritha B	(111722102044)
Arulmozhi S	(111722102011)

R.M.K. ENGINEERING COLLEGE

(An Autonomous Institution)

R.S.M. Nagar, Kavaraipettai-601 206

BONAFIDE CERTIFICATE

Certified that this project report **"F.R.I.D.A.Y"** is the bonafide work of **Divyabharathi L (111722102031), Haritha B (111722102044), Arulmozhi S (111722102011)** who carried out the **22ME311 Product Development Lab 3** work under my supervision**.**

SIGNATURE

Dr. T. Sethukarasi, M.E., M.S. Ph.D.,
Professor and Head
Department of Computer Science and Engineering
R.M.K. Engineering College
R.S.M. Nagar, Kavaraipettai,
Tiruvallur District– 601206.

SIGNATURE

Dr.A.Thilagavathy/ Dr.K.Manikannan
Supervisor
Associate Professor/ Associate Professor
Department of Computer Science and Engineering
R.M.K. Engineering College
R.S.M. Nagar, Kavaraipettai,
Tiruvallur District–601206.

Submitted for the Product Development Lab – 3 held on …… at **R.M.K. Engineering College**, Kavaraipettai, Tiruvallur District– 601206.

INTERNAL EXAMINER

ACKNOWLEDGEMENT

We earnestly portray our sincere gratitude and regard to our beloved **Chairman Shri. R. S. Munirathinam, our Vice Chairman, Shri. R. M. Kishore** and **our Director, Shri. R. Jyothi Naidu,** for the interest and affection shown towards us throughout the course.

We convey our sincere thanks to our **Principal**, **Dr. K. A. Mohamed Junaid,** for being the source of inspiration in this college.

We reveal our sincere thanks to our **Professor and Head of the Department, Computer Science and Engineering, Dr. T. Sethukarasi,** for her commendable support and encouragement for the completion of our project.

We would like to express our sincere gratitude for our Product Development Lab 3 Coordinator and project guide Dr.A.Thilagavathy, Associate Professor, Department of Computer Science and Engineering and Dr.K.Manikannan, Associate Professor, Department of Computer Science and Engineering for their valuable suggestions towards the successful completion for this project in a global manner.

We take this opportunity to extend our thanks to all faculty members of Department of Computer Science and Engineering, parents and friends for all that they meant to us during the crucial times of the completion of our project.

TABLE OF CONTENTS

ABSTRACT

In the era of advanced technology, the integration of artificial intelligence (AI) into our daily lives has become increasingly prevalent. This project **FRIDAY** abbreviated as **FRiendly Intelligent Desktop Assistant for You** explores the development and implementation of a Voice-Based AI Virtual Assistant for Windows platform, leveraging the cutting-edge capabilities of OpenAI's GPT-3 technology. The primary objective of this project is to create an intuitive and efficient virtual assistant that can understand natural language commands and perform tasks seamlessly through voice interactions.

The project focuses on harnessing the power of GPT-3, a state-of-the-art language generation model, to enable the virtual assistant to comprehend and respond to user queries accurately. By utilizing GPT-3's deep learning algorithms, the virtual assistant gains the ability to process complex language patterns, enhancing its comprehension of user inputs and improving the overall user experience.

Key features of the Voice-Based AI Virtual Assistant include natural language processing (NLP) capabilities, real-time speech recognition, context-aware responses, and personalized user interactions. Users can interact with the virtual assistant using voice commands, enabling them to perform tasks such as setting reminders, sending emails, searching the web, controlling system settings, and accessing relevant information from the internet. The system also ensures privacy and security by implementing robust encryption protocols to safeguard user data and interactions.

The project's significance lies in its potential to revolutionize the way users interact with their Windows devices, making the interaction more intuitive, efficient, and hands-free. By

integrating GPT-3 technology, the virtual assistant can adapt and learn from user interactions, continually improving its responses and enhancing user satisfaction.

Through this innovative Virtual Assistant, this project aims to pave the way for a future where AI technology seamlessly integrates into our daily lives, simplifying tasks, and enhancing productivity. The project's outcomes contribute to the broader field of AI research and open new avenues for the development of intelligent virtual assistants across various platforms.

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

In the rapidly evolving digital landscape, users increasingly rely on technology to simplify tasks and enhance productivity. However, the existing virtual assistant solutions for Windows platforms often lack the natural language understanding and contextual comprehension necessary for seamless user interactions. Current virtual assistants struggle with accurately interpreting complex user commands, leading to frustration and inefficiency in user-device interactions. The existing gap in the market highlights the need for an advanced Voice-Based AI Virtual Assistant specifically tailored for Windows users, capable of understanding natural language commands and executing tasks with precision.

1.2 Project Scope and Objective:

1.2.1 Scope of the Project:

The scope of the project encompasses the design, development, and implementation of a Voice-Based AI Virtual Assistant for Windows utilizing GPT-3 technology. The virtual assistant will be tailored to Windows platforms, ensuring seamless integration with the operating system's functionalities. The project will focus on natural language processing, real-time speech recognition, context-aware responses, and personalized user interactions. Additionally, the virtual assistant will prioritize user privacy and data security, implementing encryption protocols to safeguard user interactions and sensitive information.

1.2.2 Objective of the Project:

- ✓ **Natural Language Understanding:** Develop advanced natural language processing algorithms to enable the virtual assistant to understand and interpret complex user commands accurately.

- ✓ **Real-Time Speech Recognition:** Implement robust speech recognition algorithms to enable the virtual assistant to process voice commands in real time, ensuring prompt and efficient user interactions.

- ✓ **Context-Aware Responses:** Utilize GPT-3 technology to provide context-aware responses, allowing the virtual assistant to understand the conversation context and respond intelligently to user queries.

- ✓ **Personalized User Interactions:** Implement machine learning techniques to enable the virtual assistant to learn from user interactions, adapting its responses and behavior based on individual preferences and usage patterns.

- ✓ **Task Automation:** Enable the virtual assistant to perform a wide range of tasks, including setting reminders, sending emails, searching the web, controlling system settings, and accessing relevant information, all through voice commands.

- ✓ **User Experience Enhancement:** Focus on creating an intuitive and user-friendly interface, ensuring a seamless and enjoyable user experience while interacting with the virtual assistant.

- ✓ **Privacy and Security:** Implement robust encryption protocols to safeguard user data, ensuring privacy and security in all interactions between the virtual assistant and the user.

- ✓ **Testing and Validation:** Conduct rigorous testing procedures to validate the accuracy, efficiency, and reliability of the virtual assistant in understanding and executing user commands.

- ✓ **Documentation and User Guide:** Provide comprehensive documentation and user guides to assist users in effectively interacting with the virtual assistant, understanding its capabilities, and maximizing its utility.

By achieving these objectives within the defined scope, the project aims to deliver an intelligent and efficient Voice-Based AI Virtual Assistant for Windows, enhancing user productivity and revolutionizing the way users interact with their Windows devices.

CHAPTER 2

OVERALL DESCRIPTION

2 PROJECT SPECIFICATION

2.1. SYSTEM STUDY

In the system study phase, an in depth analysis of the existing system is conducted to identify its strengths, weaknesses, and limitations. This phase involves understanding user requirements, current workflows, and technology constraints. The findings from the system study guide the development of the proposed system.

2.1.1 EXISTING SYSTEM

- ✓ **Limited Natural Language Processing:** The existing virtual assistant lacks advanced natural language processing capabilities, making it challenging to comprehend complex user commands accurately.
- ✓ **Basic Speech Recognition:** Speech recognition in the current system is basic, leading to occasional misinterpretation of voice commands and delayed responses.
- ✓ **Static Responses:** The virtual assistant provides static responses and lacks context-awareness, leading to limited user engagement and interaction.
- ✓ **Limited Task Automation:** Task automation features are limited, restricting the virtual assistant's ability to perform a wide range of tasks through voice commands.
- ✓ **Privacy Concerns:** There might be concerns related to user privacy and data security, as the existing system might lack robust encryption protocols.

2.1.2 PROPOSED SYSTEM

- ✓ **Advanced Natural Language Processing:** Implement advanced natural language processing techniques, leveraging GPT-3 technology, to enhance the virtual assistant's understanding of complex user commands, ensuring accurate interpretation.
- ✓ **Real-Time Speech Recognition:** Integrate robust and real-time speech recognition algorithms, enabling the virtual assistant to promptly and accurately process voice commands from users.
- ✓ **Context-Aware Responses:** Utilize GPT-3's capabilities to provide context-aware responses, enabling the virtual assistant to understand conversation context and respond intelligently to user queries.
- ✓ **Comprehensive Task Automation:** Enhance task automation capabilities, allowing the virtual assistant to perform a wide array of tasks such as setting reminders, sending emails, searching the web, and controlling system settings, all through voice commands.
- ✓ **Privacy and Security:** Implement strong encryption protocols to safeguard user data and interactions, addressing privacy concerns and ensuring secure communication between the virtual assistant and the user.

By addressing the limitations of the existing system and incorporating advanced features powered by GPT-3 technology, the proposed system aims to create a sophisticated Voice-Based AI Virtual Assistant for Windows. This system will revolutionize user-device interactions, offering a seamless, intuitive, and secure experience for users while maximizing productivity and user engagement.

2.2 USE CASE DIAGRAM

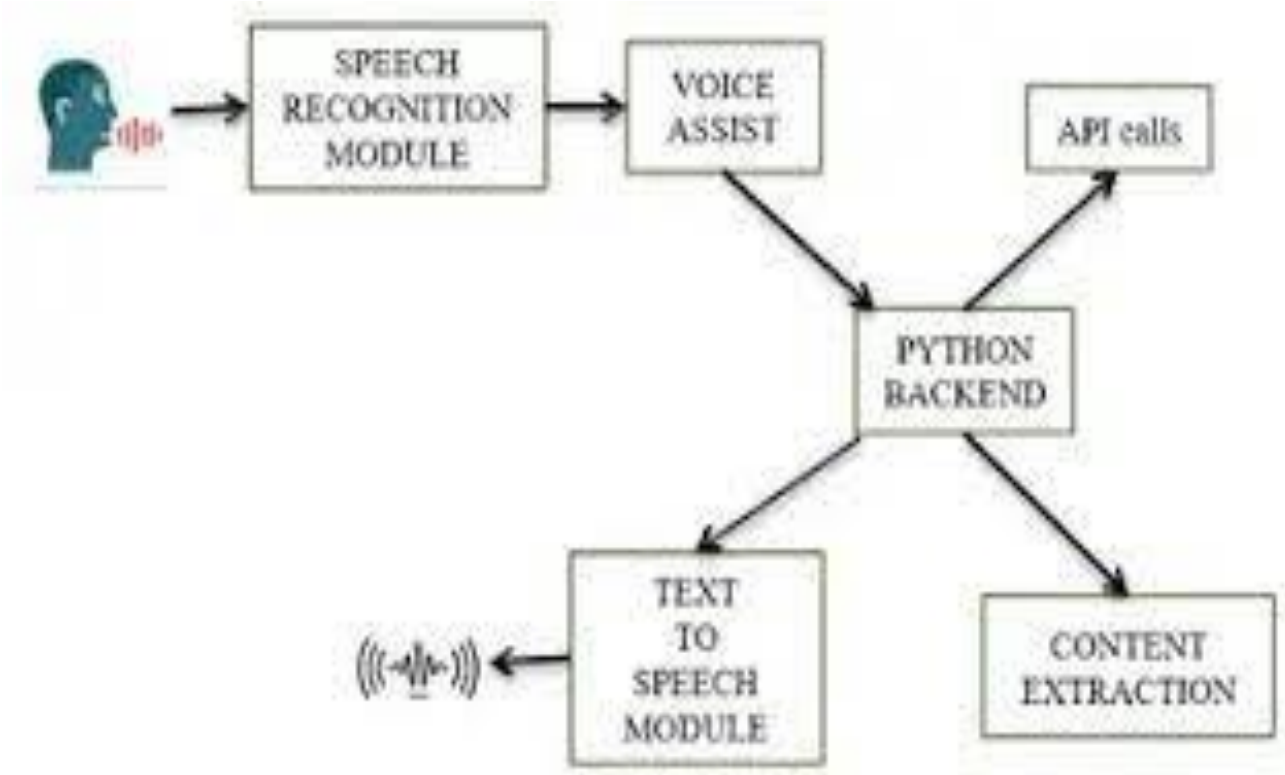

Fig.1

2.3 DESIGN

- ✓ **User Interactions:**

- Activate Virtual Assistant: The user activates the virtual assistant using a wake word or a specific command.
- Send Voice Command: The user sends voice commands to the virtual assistant for various tasks.
- Receive Response: The virtual assistant processes the user's command and responds with appropriate actions or information.
- End Interaction: The user ends the interaction with the virtual assistant.

- ✓ **System Actions:**

- Process Voice Command: The system processes the voice command using speech recognition and natural language processing techniques.
- Perform Task: The system performs tasks based on the user's command, such as sending emails, setting reminders, or searching the web.
- Generate Response: The system generates context-aware responses using GPT-3 technology to provide intelligent and relevant answers to user queries.

- ✓ **External Systems:**

- Access Database: The system may access a database to retrieve user-specific information or store user preferences.
- Internet API: The system interacts with internet APIs to fetch real-time information, conduct web searches, or access online services based on user requests.

2.4 SYSTEM ARCHITECTURE

The system architecture of the FRIDAY involves several key components:

- ✓ **User Interface (UI):** The UI component allows users to interact with the virtual assistant through voice commands. It captures user input and initiates the processing of voice commands.

- ✓ **Speech Recognition Module:** This module processes the audio input from the user, converting it into text data using advanced speech recognition algorithms. It plays a crucial role in accurately understanding user commands.

- ✓ **Natural Language Processing (NLP) Module:** The NLP module analyzes the textual input, identifying intents and entities within the user's command. It utilizes GPT-3 technology to comprehend complex language patterns and context, ensuring accurate interpretation of user requests.

- ✓ **Task Automation Engine:** This component is responsible for executing tasks based on the user's commands. It interacts with the Windows operating system and relevant applications to perform actions like sending emails, setting reminders, or controlling system settings.

- ✓ **Context Management System:** The context management system maintains the conversation context, enabling the virtual assistant to understand ongoing discussions, reference previous interactions, and generate context-aware responses.

- ✓ **External Services Integration:** This component interacts with external services, such as databases and internet APIs, to fetch or store relevant information, enhancing the virtual assistant's capabilities and providing up-to-date responses to user queries.

- ✓ **Security and Privacy Layer:** This layer ensures the encryption of user data during communication, safeguarding user privacy and data security. It implements robust security protocols to protect sensitive user information.

By integrating these components, the system architecture forms a cohesive framework that enables the Voice-Based AI Virtual Assistant to provide intelligent, efficient, and secure interactions, enhancing user experience and productivity on Windows platforms.

CHAPTER 3

EXTERNAL INTERFACE REQUIREMENTS

3.1 USER INTERFACE

The user interface of the Voice-Based AI Virtual Assistant for Windows should be intuitive and user-friendly, allowing users to interact with the system effortlessly. It should include the following features:

- ✓ **Voice Input:** Users can provide voice commands to the virtual assistant using a microphone or the built-in microphone of the device.

- ✓ **Visual Feedback:** The system may provide visual feedback to users, such as on-screen notifications or animations, to indicate that the virtual assistant is processing the command.

- ✓ **Error Messages:** Clear and concise error messages should be displayed in case the system encounters difficulties understanding the user's command.

3.2 HARDWARE INTERFACE

The virtual assistant system requires specific hardware components to function effectively. These include:

- ✓ **Microphone:** A functional microphone is necessary for capturing user voice commands accurately. It can be an external microphone or the built-in microphone of the device.

- ✓ **Speaker:** A speaker is needed to provide audio responses to the user. It can be the device's built-in speaker or an external speaker connected to the system.

3.3 SOFTWARE INTERFACE

The virtual assistant relies on various software interfaces and technologies to operate seamlessly. These include:

- ✓ **Operating System:** The system should be compatible with Windows operating systems, ensuring that it can run smoothly on different versions of Windows, such as Windows 10, Windows 11, etc.

- ✓ **Speech Recognition API:** Integration with a speech recognition API or library is necessary to convert user voice commands into text data for processing.

- ✓ **Natural Language Processing (NLP) Library:** Utilization of an NLP library or API is essential to analyze and understand the user's textual input, extracting intents and entities from the commands.

- ✓ **GPT-3 API:** Integration with the GPT-3 API is crucial for leveraging its language processing capabilities, enabling the virtual assistant to generate context-aware responses to user queries

- ✓ **Database Interface :** If the system interacts with a database to retrieve or store user-specific information, it needs a database interface to connect, query, and manage the database.

- ✓ **Internet API:** Integration with internet APIs is necessary for conducting web searches, fetching real-time information, or accessing online services basis.

CHAPTER 4

TESTING

4.1 TEST PLAN

The test plan outlines the approach, scope, resources, and schedule for testing the Voice-Based AI Virtual Assistant for Windows. It includes various test scenarios, methodologies, and acceptance criteria to ensure the system functions as intended. The primary objectives of the test plan include validating the system's accuracy, efficiency, user-friendliness, and security features.

4.2 TEST PROCEDURE

a. Unit Testing:

- Objective: Verify the functionality of individual components, such as speech recognition, natural language processing, task automation, and external service integration.
- Procedure: Test each component in isolation, ensuring they perform as expected and handle different inputs and edge cases effectively.

b. Integration Testing:

- Objective: Validate the interactions between system components to ensure seamless communication and data flow.
- Procedure: Test the integration points between modules, focusing on input-output consistency, error handling, and exception scenarios.

c. System Testing:

- Objective: Evaluate the complete system to ensure it meets the specified requirements and user expectations.
- Procedure: Conduct end-to-end tests, including user interactions, voice command processing, task execution, context-aware responses, and external service integrations. Test various user scenarios to validate the system's behavior in real-world situations.

d. User Acceptance Testing (UAT):

- Objective: Allow end-users to validate the system's functionality, usability, and overall satisfaction.
- Procedure: Engage real users to interact with the virtual assistant, providing feedback on their experience. Address user concerns and ensure the system aligns with user expectations.

4.3 TEST DELIVERABLES

✓ **Test Cases:** Detailed test cases covering various scenarios, inputs, and expected outcomes for unit, integration, and system testing.

✓ **Test Results:** Comprehensive documentation of test results, including successful tests, issues encountered, and their resolutions.

- ✓ **User Feedback Report:** Summary of user feedback from the UAT phase, highlighting user satisfaction, identified issues, and suggestions for improvement.

- ✓ **Bug Reports:** Detailed reports on any defects found during testing, including steps to reproduce, expected behavior, and actual behavior.

- ✓ **Test Summary Report:** A comprehensive report summarizing the testing process, outcomes, challenges faced, and recommendations for further improvements.

By adhering to this testing plan and procedures, the Voice-Based AI Virtual Assistant for Windows can be thoroughly evaluated, ensuring a high-quality, reliable, and user-friendly product for end-users.

CHAPTER 6

CONCLUSION

In conclusion, the development of the Voice-Based AI Virtual Assistant for Windows using GPT-3 technology represents a significant leap forward in human-computer interaction. By leveraging the power of advanced natural language processing and real-time speech recognition, this virtual assistant offers users an intuitive, efficient, and hands-free way to interact with their Windows devices. Throughout the course of this project, we have addressed the limitations of existing virtual assistant solutions, providing an innovative and intelligent alternative.

By understanding complex user commands, providing context-aware responses, and executing tasks seamlessly through voice interactions, the virtual assistant enhances user productivity and simplifies tasks in an unprecedented manner. Moreover, the integration of GPT-3 technology enables the virtual assistant to continuously learn, adapt, and improve, ensuring that users receive accurate and relevant information with each interaction.

With a focus on user privacy and data security, robust encryption protocols have been implemented, guaranteeing the confidentiality of user interactions and sensitive information. The project's commitment to user satisfaction is reflected in the implementation of personalized user interactions and the continuous pursuit of future enhancements, ensuring that the virtual assistant remains adaptable and responsive to evolving user needs.

As we move forward, the Voice-Based AI Virtual Assistant for Windows stands as a testament to the potential of artificial intelligence to enhance our daily lives. By providing a seamless bridge between users and technology, this virtual assistant paves the way for a future where human-computer interactions are not just efficient but also deeply intuitive and natural. Through this project, we have taken a significant step towards redefining the way users interact with their digital environments, promising a future where technology truly understands and empowers us.

CHAPTER 5

FUTURE ENHANCEMENTS

- ✓ Multi-Platform Support: Extend the virtual assistant's capabilities to operate seamlessly across various platforms, including macOS, Linux, iOS, and Android, allowing users to have a consistent experience across their devices.

- ✓ Voice Recognition Improvement: Invest in research and development to enhance voice recognition accuracy, enabling the virtual assistant to understand diverse accents, languages, and speech patterns effectively.

- ✓ Personalization: Implement advanced machine learning algorithms to learn user preferences over time, providing personalized recommendations, responses, and tailored user experiences.

- ✓ Contextual Understanding: Enhance the virtual assistant's ability to understand complex contexts within conversations, enabling it to provide more accurate and contextually relevant responses.

- ✓ Third-Party Integrations: Integrate the virtual assistant with popular third-party applications and services, allowing users to seamlessly interact with their favorite apps through voice commands.

- ✓ Offline Mode: Develop an offline mode for the virtual assistant, enabling basic functionalities such as setting alarms, reminders, and accessing local information without an internet connection, ensuring continuous usability.

- ✓ Interactive Learning: Implement interactive learning mechanisms, allowing users to provide feedback on the virtual assistant's responses, helping the system improve its accuracy and user satisfaction continuously.

- ✓ Expanded Task Automation: Expand the range of tasks the virtual assistant can perform, such as making reservations, managing calendar events, handling financial transactions, and controlling smart home devices, making it even more versatile.

- ✓ Language Support: Increase language support to cater to a global audience, allowing users from different regions to interact with the virtual assistant in their native languages.

- ✓ Accessibility Features: Implement accessibility features, such as voice-guided navigation, screen reader compatibility, and support for users with disabilities, ensuring an inclusive user experience for all.

- ✓ Emotional Intelligence: Research and incorporate emotional intelligence algorithms, enabling the virtual assistant to recognize and respond to users' emotions appropriately, providing empathetic and supportive interactions.

- ✓ Security Enhancements: Strengthen security measures, including biometric authentication, encrypted voice data transmission, and user data anonymization, ensuring user privacy and data protection.

- ✓ Collaborative Workflows: Introduce features that facilitate collaborative workflows, allowing users to share tasks, schedules, and information with others, making the virtual assistant a valuable tool for teamwork and productivity.

- ✓ Continuous Integration of AI Advances: Stay updated with the latest advancements in AI and natural language processing, integrating cutting-edge technologies to enhance the virtual assistant's capabilities and user experience continually.

By focusing on these future enhancements, the Voice-Based AI Virtual Assistant can evolve into a highly intelligent, versatile, and indispensable tool for users, revolutionizing the way they interact with technology and enhancing their daily lives.

CHAPTER 6

CONCLUSION

In conclusion, the development of the Voice-Based AI Virtual Assistant for Windows using GPT-3 technology represents a significant leap forward in human-computer interaction. By leveraging the power of advanced natural language processing and real-time speech recognition, this virtual assistant offers users an intuitive, efficient, and hands-free way to interact with their Windows devices. Throughout the course of this project, we have addressed the limitations of existing virtual assistant solutions, providing an innovative and intelligent alternative.

By understanding complex user commands, providing context-aware responses, and executing tasks seamlessly through voice interactions, the virtual assistant enhances user productivity and simplifies tasks in an unprecedented manner. Moreover, the integration of GPT-3 technology enables the virtual assistant to continuously learn, adapt, and improve, ensuring that users receive accurate and relevant information with each interaction.

With a focus on user privacy and data security, robust encryption protocols have been implemented, guaranteeing the confidentiality of user interactions and sensitive information. The project's commitment to user satisfaction is reflected in the implementation of personalized user interactions and the continuous pursuit of future enhancements, ensuring that the virtual assistant remains adaptable and responsive to evolving user needs.

As we move forward, the Voice-Based AI Virtual Assistant for Windows stands as a testament to the potential of artificial intelligence to enhance our daily lives. By providing a seamless bridge between users and technology, this virtual assistant paves the way for a future where human-computer interactions are not just efficient but also deeply intuitive and natural. Through this project, we have taken a significant step towards redefining the way users interact with their digital environments, promising a future where technology truly understands and empowers us.

REFERENCES

[1] Ms. Preethi G, Mr. Thiruppugal S, Mr. Vishwaa D A, Mr. Abishek K, “Voice Assistant using Artificial Intelligence”, International Journal of Engineering Research, Vol. 11 , ISSN: 2278-0181, May-2022.

[2] Harshil Asodariya , Keval Vachhani , Eishan Ghori , Brijesh Babariya,Tejal Patel, “Desktop Voice Assistant”, International Journal of Innovative Science and Research Technology, Volume 8, Issue 2, February – 2023.

[3] Saurabh Biradar, Prasad Bramhapurkar, Rakesh Choudhari, Snehal Patil, Prof. Deepa Kulkarni, “Personal Virtual Voice Desktop Assistant And Intelligent Decision Maker”, International Research Journal of Modernization in Engineering Technology and Science,Volume 05,Issue 02,February -2023.

[4] Anjali Fapal, Trupti Kanade, Bharati Janrao, Mrunalini Kamble, Megha Raule, “Personal Virtual Assistant for Windows Using Python”, International Research Journal of Modernization in Engineering Technology and Science, Volume 03, Issue07, July-2021.

[5] Rose Thomas, Surya V S, Tincy A Mathew, Tinu Thomas, “Voice Based Intelligent Virtual Assistant for Windows Using Python”, International Journal of Engineering Research & Technology (IJERT),2023.

[6] Impana N.R., Prof. G.R.Manjul, “Voice and Text Based Virtual Personal Assistant For Desktop”, International Journal of Engineering Applied Sciences and Technology, 2022.

[7] Shubham Thorbole , Anuradha Pandit ,Gayatri Raut , Tejas Sirsat, “AI-Based Desktop Voice Assistant”,Journal of Emerging Technologies & Innovative Research, Volume 10, Issue 5,May 2023.

[8] Shrinivas Kulkarni, Praveen More, Varad Kulkarni, Vaishnavi Patil, Harsh Patel, Mrs. Pooja Patil, “Alpha: The Desktop Assistant,International Research Journal of Modernization in Engineering Technology and Science, Volume 04,Issue 06,June-2022.

[9] Oleg , Yunakov, “ Personal Virtual Assistant “, Pace University, New York, NY School of Computer Science and Information Systems May 13, 2004.

[10] Hyunji Chung and Sangjin Lee, “ Intelligent Virtual Assistant knows Your Life”, International Journal of Engineering Applied Sciences and Technology, Oct 3 2020

SAMPLE CODING

```
import pyttsx3
import speech_recognition as sr
import pyaudio
import datetime
import os
import wikipedia
import webbrowser

engine=pyttsx3.init('sapi5')
voices=engine.getProperty('voices')
#print(voices[0].id)
engine.setProperty('voices',voices[0].id)

#text to speech
def speak(audio):
    engine.say(audio)
    print (audio)
    engine.runAndWait()

#to convert voice into text
def takecommand():
    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Listening...")
```

```
        r.pause_threshold=1
        audio = r.listen(source,timeout=1,phrase_time_limit=50)

    try:
        print("Recognizing..")
        query = r.recognize_google(audio,language='en-in')
        print(f"user said: {query}")

    except Exception as e:
        speak("Say that again please...")
        return "none"
    return query

#to wish

def wish():
    hour=int (datetime.datetime.now().hour)

    if hour>=0 and hour<=12:
        speak("Good morning")
    elif hour>12 and hour<18:
        speak("Good afternoon")
    else:
        speak("Good Evening")
    speak("I am friday sir.Please tell me how can I help you")

if _name=="__main_":
```

```
#takecommand()
wish()
#while True:
if 1:
    query=takecommand().lower()

    #logic building for tasks
    if "open notepad" in query:
        npath="C:\\Windows\\notepad.exe"
        os.startfile(npath)

    elif "open command prompt" in query:
        os.system("start cmd")

    elif "wikipedia" in query:
        speak("Searching wikipedia...")
        query=query.replace("wikipedia","")
        results=wikipedia.summary(query,sentences=2)
        speak("According to wikipedia")
        speak(results)
        print(results)
```

SCREENSHOTS

GUI:

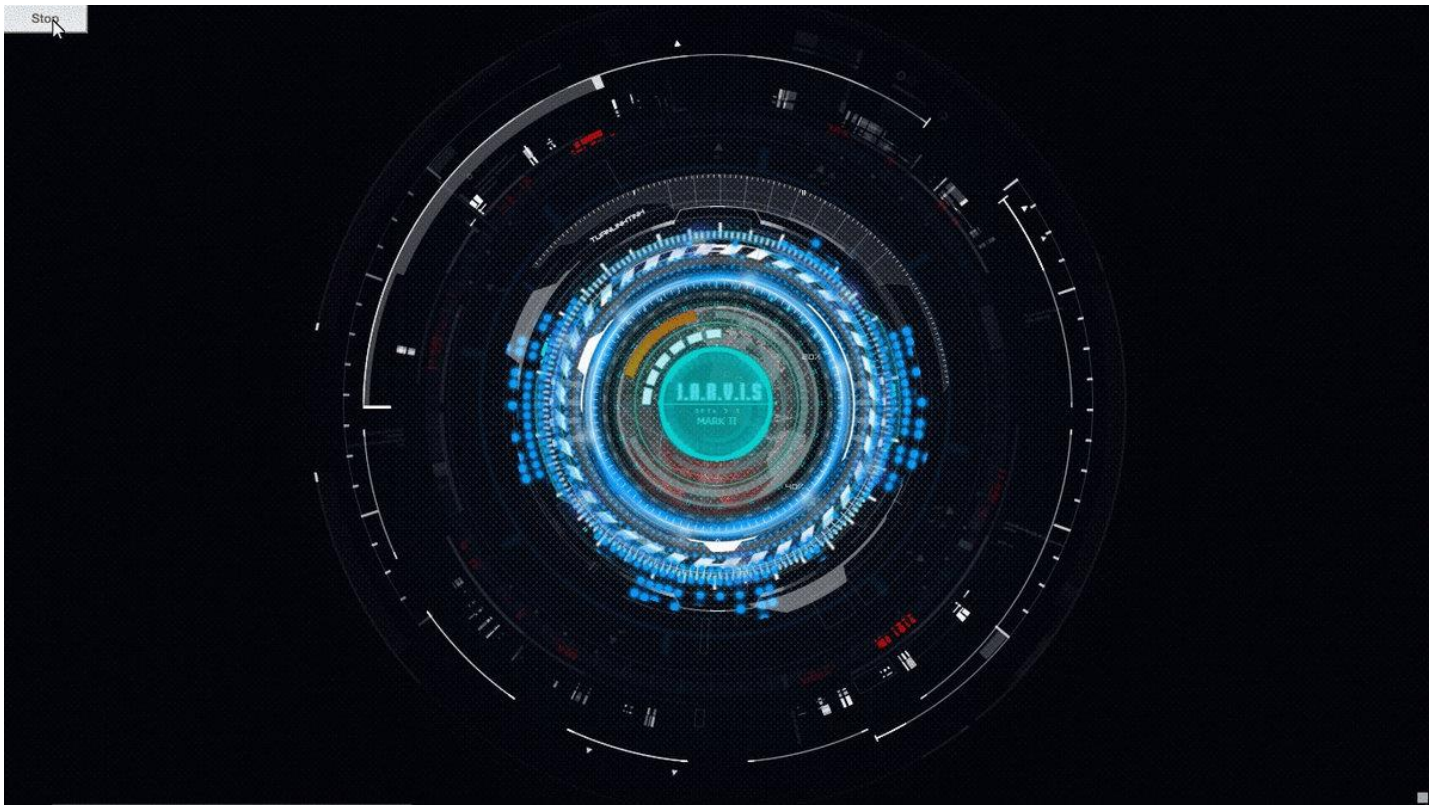

Fig.2

CODE AND OUTPUT:

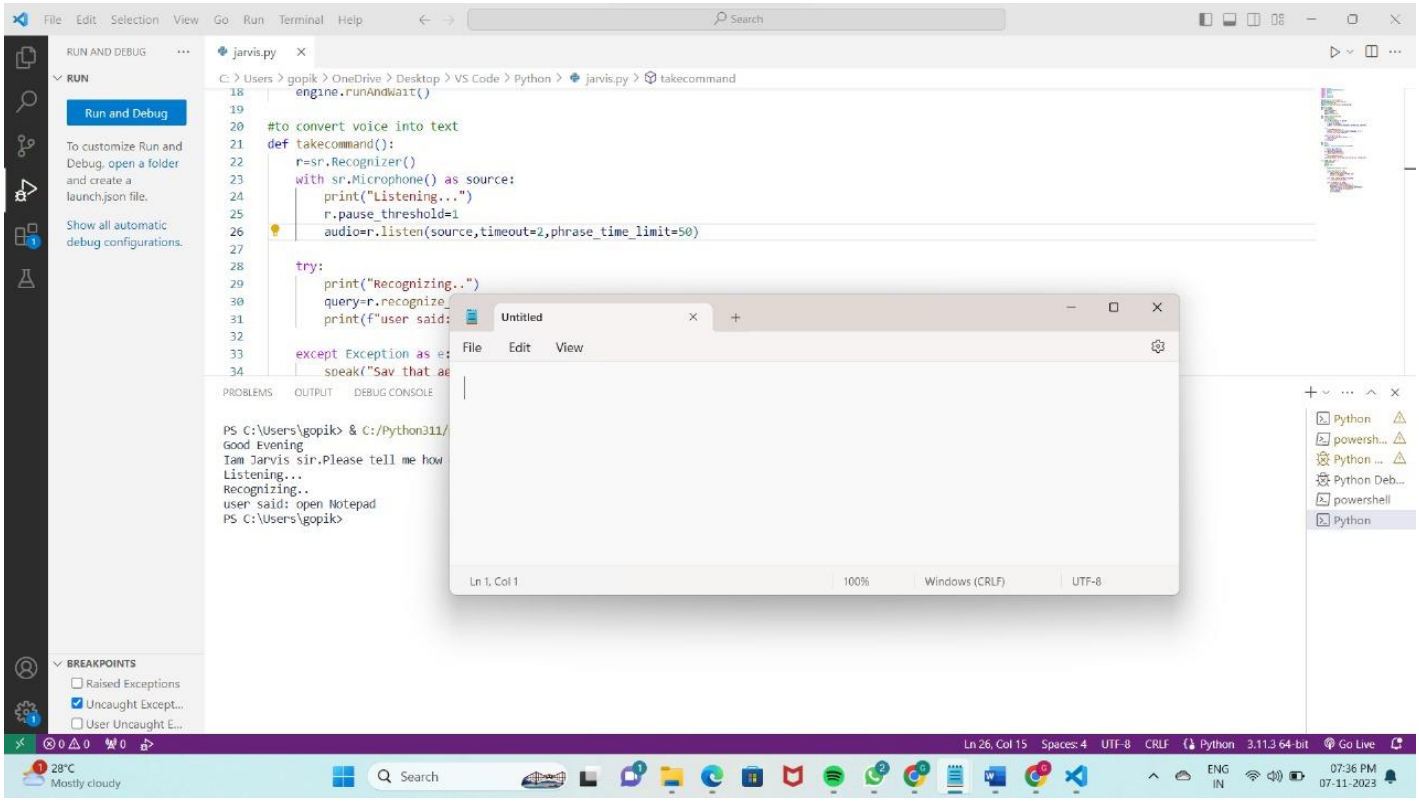

Fig. 3

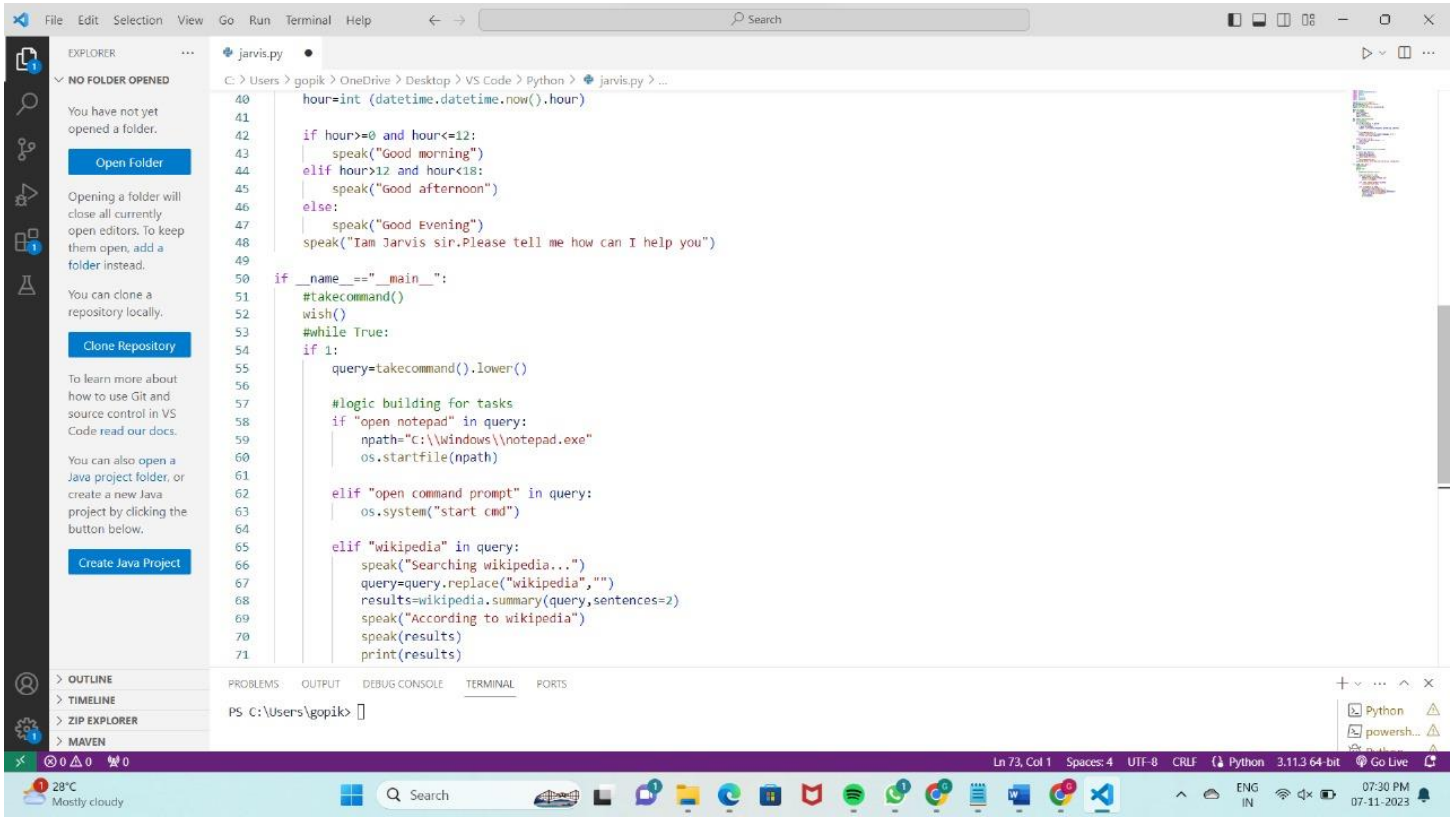

Fig. 4

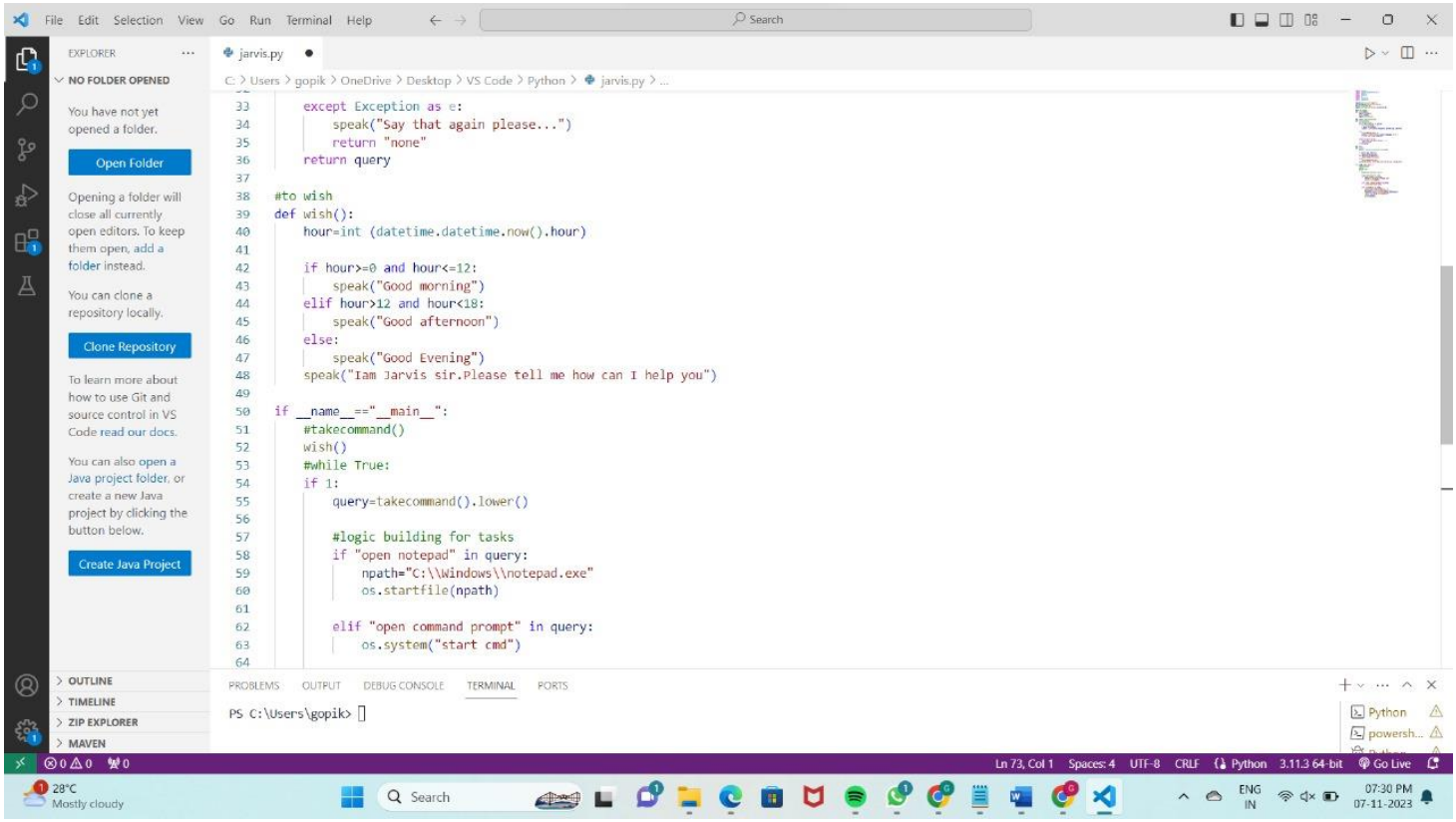
except Exception as e:
speak("Say that again please...")
return "none"
return query
#to wish
def wish():
hour=int (datetime.datetime.now().hour)
if hour>=0 and hour<=12:
speak("Good morning")
elif hour>12 and hour<18:
speak("Good afternoon")
else:
speak("Good Evening")
speak("Iam Jarvis sir.Please tell me how can I help you")
if __name__=="__main__":
#takecommand()
wish()
#while True:
if 1:
query=takecommand().lower()
#logic building for tasks
if "open notepad" in query:
npath="C:\\Windows\\notepad.exe"
os.startfile(npath)
elif "open command prompt" in query:
os.system("start cmd")
PS C:\Users\gopik>

Fig. 5

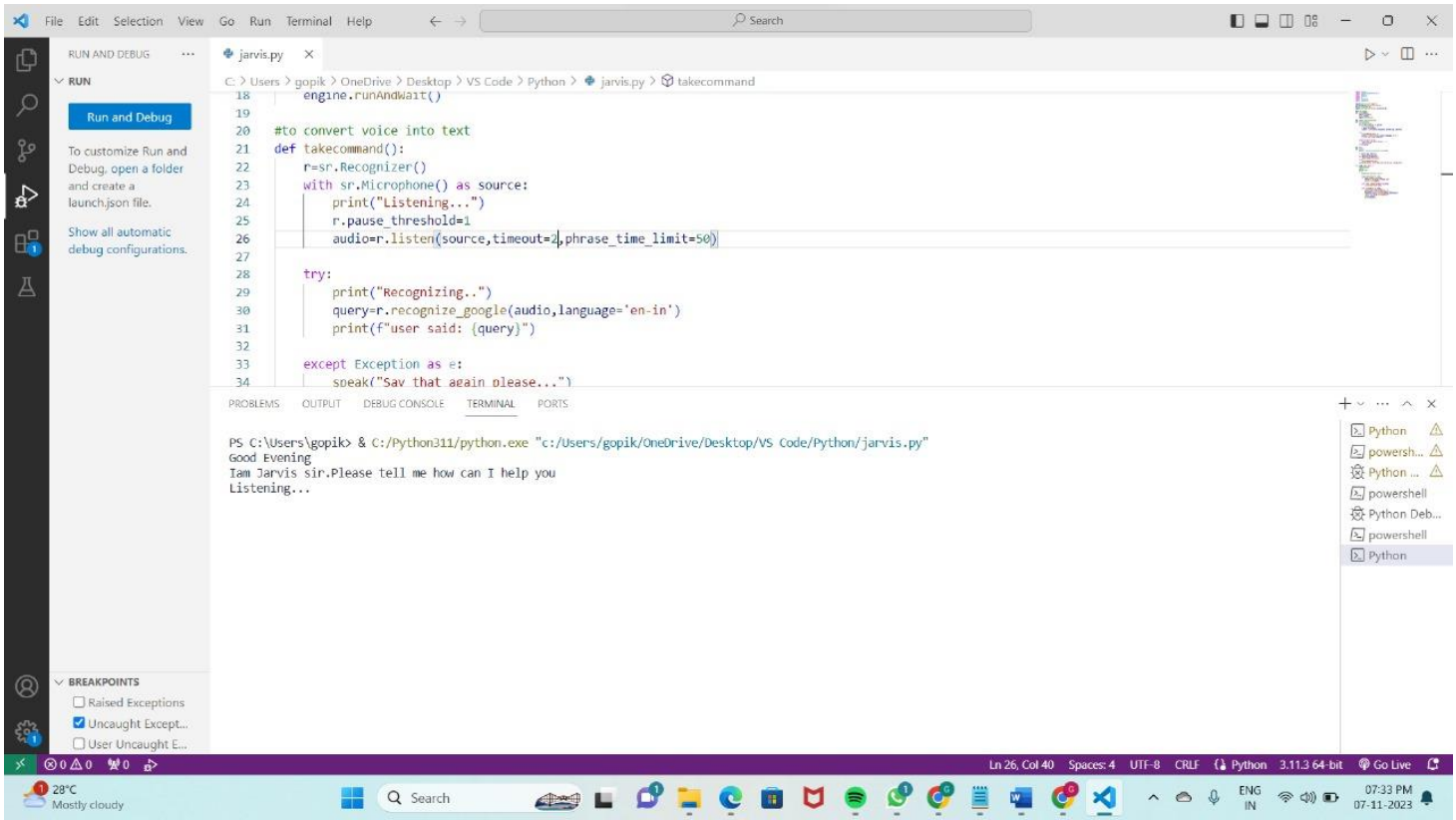
#to convert voice into text
def takecommand():
r=sr.Recognizer()
with sr.Microphone() as source:
print("Listening...")
r.pause_threshold=1
audio=r.listen(source,timeout=2,phrase_time_limit=50)
try:
print("Recognizing..")
query=r.recognize_google(audio,language='en-in')
print(f"user said: {query}")
except Exception as e:
PS C:\Users\gopik> & C:/Python311/python.exe "c:/Users/gopik/OneDrive/Desktop/VS Code/Python/jarvis.py"
Good Evening
Iam Jarvis sir.Please tell me how can I help you
Listening...

Fig. 6

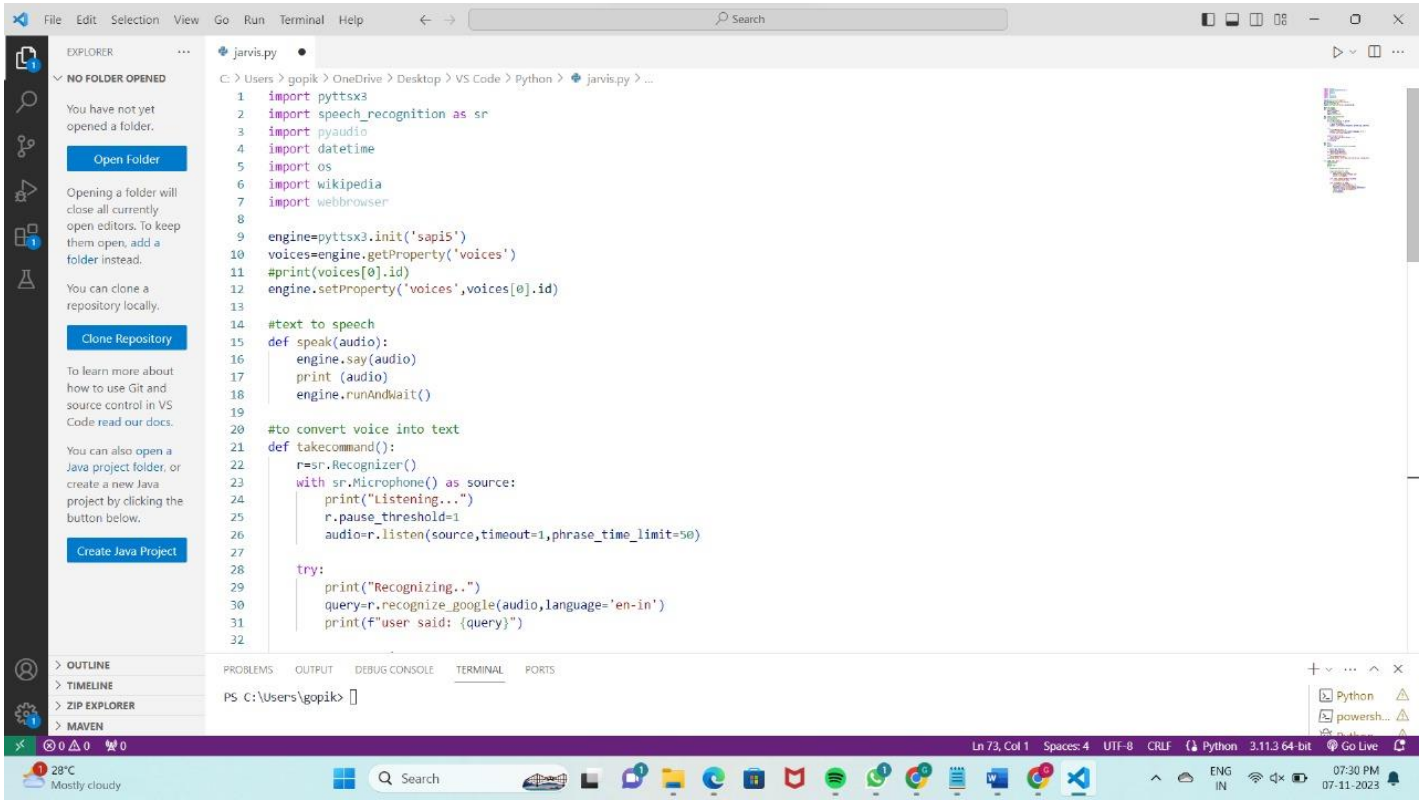
File Edit Selection View Go Run Terminal Help
Search
EXPLORER
NO FOLDER OPENED
You have not yet opened a folder.
Open Folder
Opening a folder will close all currently open editors. To keep them open, add a folder instead.
You can clone a repository locally.
Clone Repository
To learn more about how to use Git and source control in VS Code read our docs.
You can also open a Java project folder, or create a new Java project by clicking the button below.
Create Java Project
jarvis.py
C: > Users > gopik > OneDrive > Desktop > VS Code > Python > jarvis.py > ...
import pyttsx3
import speech_recognition as sr
import pyaudio
import datetime
import os
import wikipedia
import webbrowser
engine=pyttsx3.init('sapi5')
voices=engine.getProperty('voices')
#print(voices[0].id)
engine.setProperty('voices',voices[0].id)
#text to speech
def speak(audio):
engine.say(audio)
print (audio)
engine.runAndWait()
#to convert voice into text
def takecommand():
r=sr.Recognizer()
with sr.Microphone() as source:
print("listening...")
r.pause_threshold=1
audio=r.listen(source,timeout=1,phrase_time_limit=50)
try:
print("Recognizing..")
query=r.recognize_google(audio,language='en-in')
print(f"user said: {query}")
OUTLINE
TIMELINE
ZIP EXPLORER
MAVEN
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\gopik>
Python
powersh...
Ln 73, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.3 64-bit Go Live
28°C
Mostly cloudy
Search
ENG
IN
07:30 PM
07-11-2023

Fig. 7