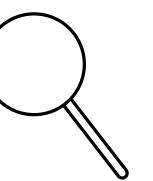


AI POWERED

DESKTOP ASSISTANT

By : Emani Rakesh



JARVI'S

Supervised by Dr. Bhrama Naidu

AGENDA

- | | | | |
|----------|-----------------------------|----------|--|
| 1 | Introduction | 5 | Functional &
Non Functional
requirements |
| 2 | Problem
Statement | 6 | Tools & Tech |
| 3 | Objective | 7 | System
Architecture |
| 4 | Modules &
Libraries Used | 8 | References &
Conclusion |



INTRODUCTION !

"A Python-based virtual assistant Jarvis. It's like having a helpful friend on your computer who listens to your voice and does things for you, making computer tasks easier. Think of it as your very own AI-powered desktop assistant!"

Let's begin!

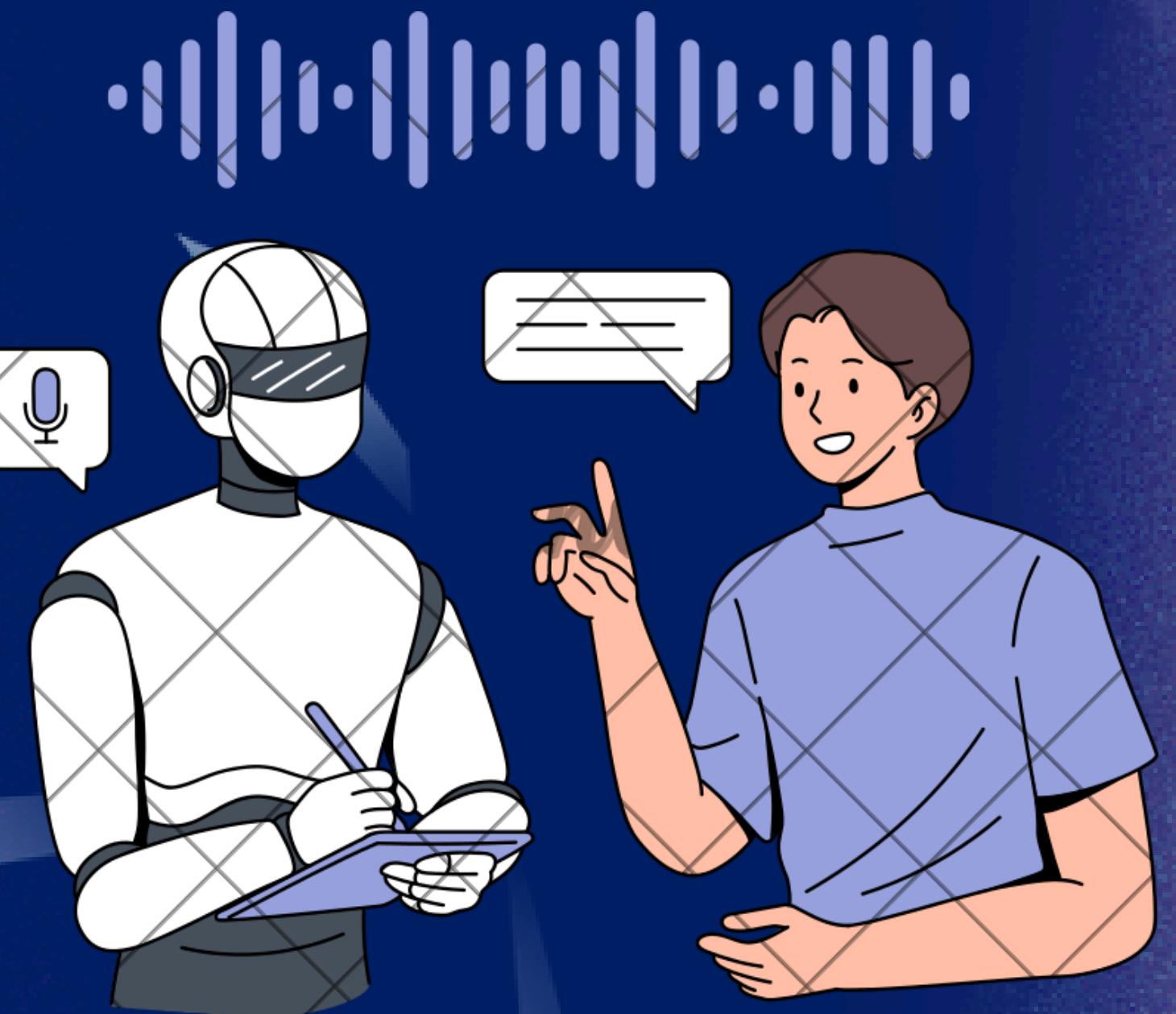


PROBLEM STATEMENT

my vision



We noticed that sometimes using computers can be hard, especially if you have a lot of things to do. So, we wanted to create a friend on the computer that can make things easier by understanding what you say and helping you with tasks."



OBJECTIVE

Build using Python



1. To Become familiar with the concepts of Python Language.
2. To Implement the concepts of basic and intermediate python operations for making this project used in real life.



Automate using voice



1. To become familiar with different modules and libraries.
2. To make ease for the users to automate the things by using own voice commands.



LIBRARIES USED



PyTTSX3



Datetime



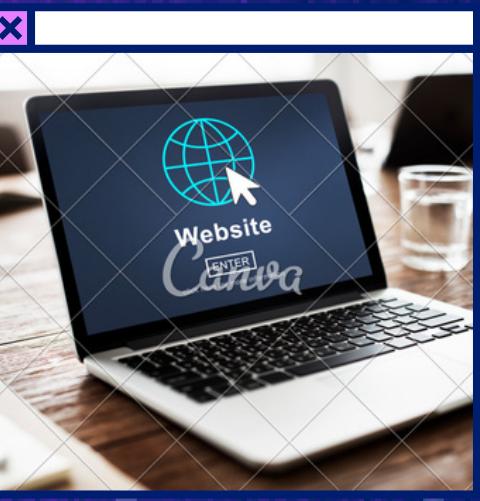
Speech Recognition



Wikipedia



PyAudio



Web Browser



OS



Random





Pyttsx3 

1. A python library that will help us to convert text to speech. In short, it is a text-to-speech library.
2. It works offline, and it is compatible with Python 2 as well as Python 3



Speech Recognition 

1. Library for performing speech recognition, with support for several engines and APIs, online and offline.
2. Used for taking input from microphone as a source to perform tasks.



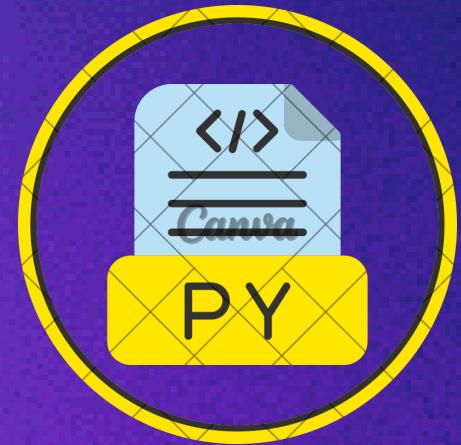
Datetime 

1. To provide current or live time to Assistant.
2. Used for greeting user according to time.



Wikipedia 

1. Wikipedia is a Python library that makes it easy to access and parse data from Wikipedia.
2. It helps the user to get results for a particular query or search.



PyAudio



1. PyAudio provides Python bindings for PortAudio v19, the cross-platform audio I/O library.
2. With PyAudio, you can easily use Python to play and record audio on a variety of platforms, such as GNU/Linux, Microsoft Windows, and Apple macOS



OS



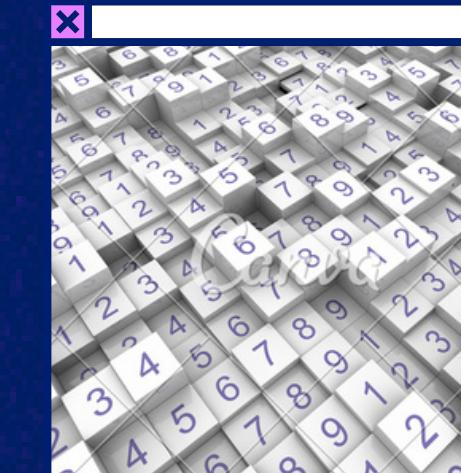
1. The OS module in Python provides functions for interacting with the operating system.
2. This module provides a portable way of using operating system-dependent functionality.



Web Browser



1. The web browser module provides a high-level interface to allow displaying Web-based documents to users
2. Under most circumstances, simply calling the open() function from this module will do the right thing.



Random



1. We can generate random numbers in Python by using random module.
2. These are pseudo-random number as the sequence of number generated depends on the seed.

FUNCTIONAL REQUIREMENTS

1 VOICE RECOGNITION

The system should be able to accurately recognize and understand spoken commands and queries.

2 NLP

It should have advanced NLP capabilities to understand and respond to natural language

3 TASK AUTOMATION

Jarvis should be able to automate various tasks, such as setting reminders, sending messages, and controlling smart home devices.

4 INFORMATION RETRIEVAL

It should be able to provide information on a wide range of topics, from news updates to general knowledge.

5 CONTEXTUAL AWARENESS

Jarvis should maintain context during a conversation to provide relevant responses.

6 MULTIMODAL INTERFACE

Support for text, voice, and potentially other forms of interaction.

NON FUNCTIONAL REQUIREMENTS

1 SCALABILITY

The system should be able to handle a growing user base and increased workload.

2 PERFORMANCE

Response times should be fast, and the system should be able to handle multiple simultaneous requests.

3 RELIABILITY

It should be available and responsive 24/7 with minimal downtime.

4 PRIVACY

Strong protection of user data and adherence to privacy regulations.

5 ACCURACY

High accuracy in understanding and responding to user queries.

6 ROBUSTNESS

The system should handle ambiguous or incomplete inputs gracefully.

TOOLS & TECHNOLOGIES



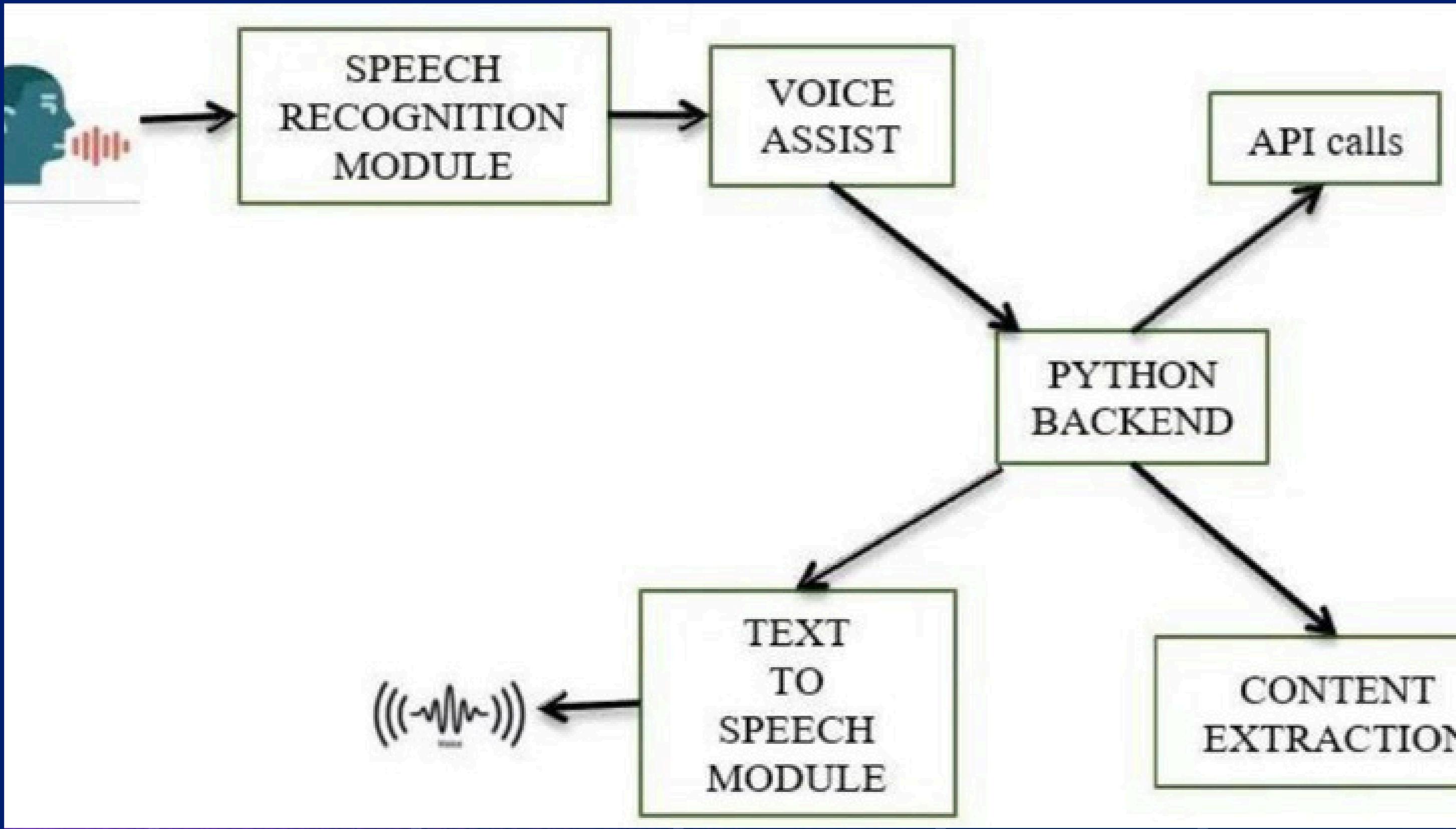
SYSTEM REQUIREMENTS

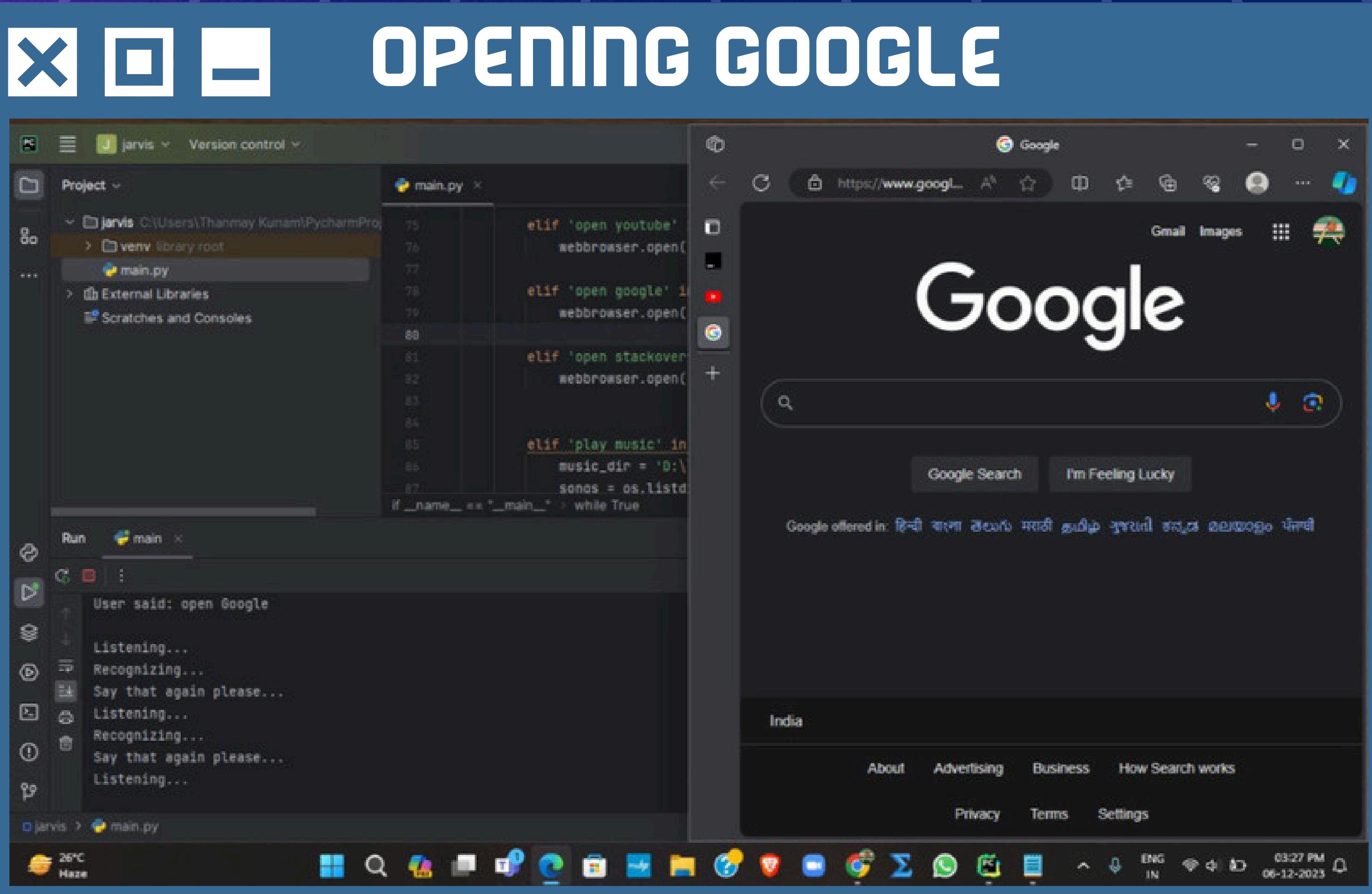
- OS : Windows 7 or above
- RAM : 4GB or above
- Hard Drive : 250 MB HDD
- Processor : Intel Dual Core

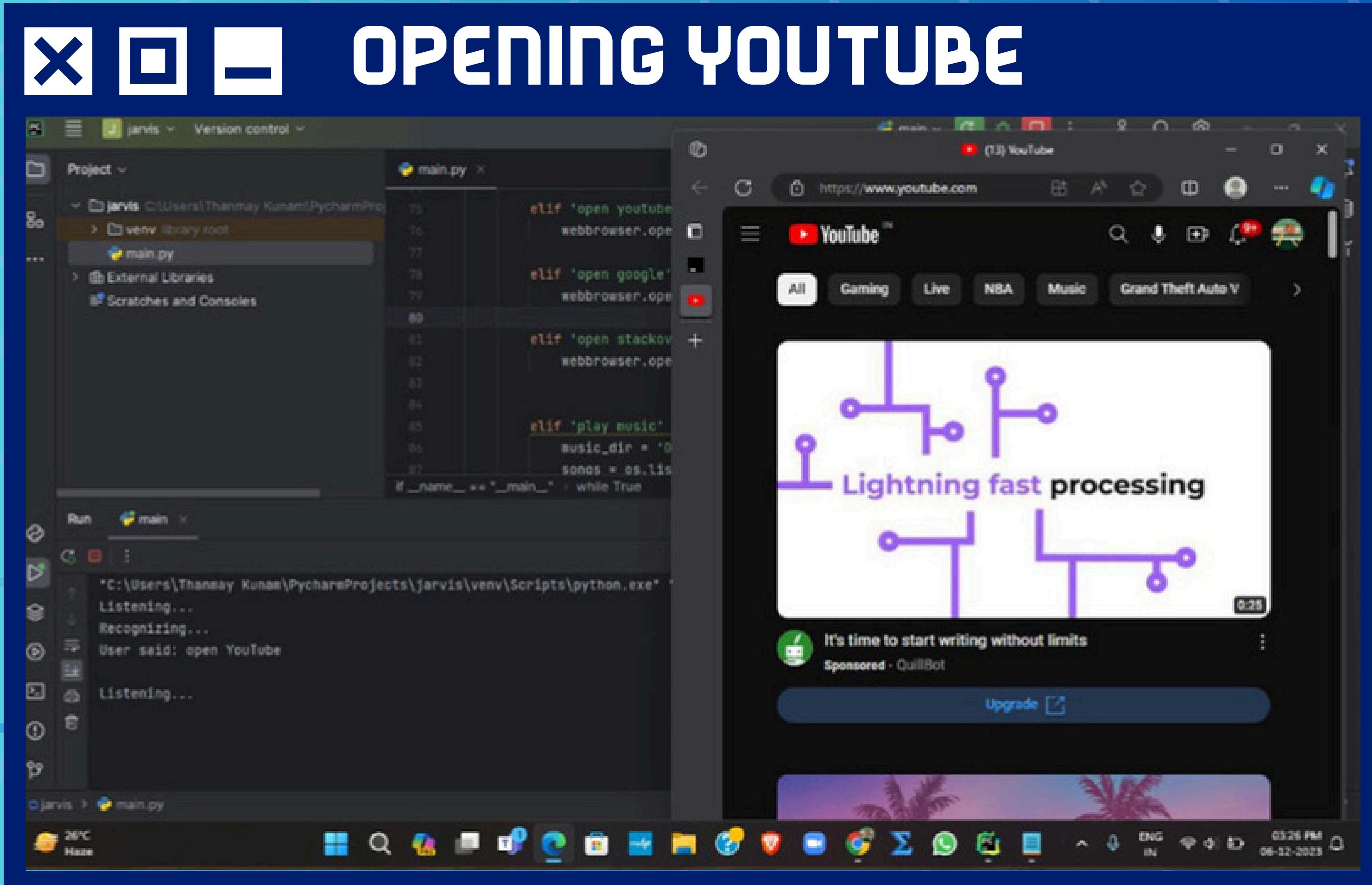




SYSTEM ARCHITECTURE









CODE SCREENSHOT

```
⚡ jarvis.py > ⚡ news
114  class MainThread(QThread):
115
141
142      def TaskExecution(self):
143          wishMe()
144          while True:
145              query = self.takeCommand().lower()
146              print(query)
147              if 'open notepad' in query:
148                  npath = "C:\\\\Windows\\\\System32\\\\notepad.exe"
149                  os.startfile(npath)
150              elif 'close notepad' in query:
151                  os.system("taskkill /f /im notepad.exe")
152              elif 'open command prompt' in query:
153                  os.system("start cmd")
154              elif 'close command prompt' in query:
155                  os.system("taskkill /f /im cmd.exe")
156              elif 'search on youtube' in query:
157                  speak("What would you like to watch?")
158                  video_query = self.takeCommand().lower()
159                  wk.playonyt(video_query)
160              elif 'close browser' in query:
161                  os.system("taskkill /f /im msedge.exe")
162              elif 'search on google' in query:
163                  speak("what should i search?")
164                  query = self.takeCommand().lower()
165                  webbrowser.open(f"{{query}}")
166
167      open word in google
```



REFERENCES

- Python : <https://www.python.org/>
- Pyttsx3 : <https://pypi.org/project/pyttsx3/>
- Pyaudio : <https://cutt.ly/gbvJuT9>
- YouTube : <https://youtu.be/Lp9Ftuq2sVI>
- Pycharm
:<https://www.jetbrains.com/pycharm/>

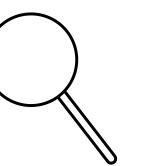
CONCLUSION

Jarvis is designed to be a comprehensive server assistant, automating various services with a single command. Its current capabilities include web searches, weather forecasts, vocabulary assistance, and medical queries. Future plans involve integrating Jarvis with mobile using React Native for a synchronized experience. Long-term goals include expanding its functionality to encompass auto deployment, supporting elastic beanstalk, backup file management, and all other operations typically handled by a server administrator, ultimately aiming to seamlessly replace the need for a human server administrator with Jarvis.

THANK YOU!

Don't hesitate to contact us

SEC :- A :- 21STUCHH010046



RAKESH

