Desktop Voice Assistant using Python

Prof. Diksha Fulzele[1], Lalit Bisen[2], Sahil Humane[3]

Assistant Professor, Dept. of Computer Science & Engineering[1]

Students, Department of Computer Science & Engineering[2,3]

Abha Gaikwad-Patil College of Engineering and Technology, Nagpur, Maharashtra, India

Abstract: *A voice assistant developed to facilitate desktop functions and deliver real-time responses to user queries through natural language interaction. The system leverages speech recognition, natural language processing (NLP), and system command execution to perform tasks like opening applications, accessing system details, adjusting volume, and providing informative answers via APIs such as ChatGPT and WolframAlpha. Additionally, the assistant features preliminary integration with IoT modules aimed at healthcare applications like IV bag monitoring. Designed to improve efficiency and enable hands-free computing, Spiral offers a user-friendly interfacefor students, working professionals, and users with accessibility needs*

Keywords: Python, Speech Recognition,NLP,pyttsx3,OpenAI, WolframAlpha, Tkinter, IoT, Desktop Automation

I. INTRODUCTION

Voice assistants are becoming popular because they make using computers easier and faster. This project introduces the introduction , a smart voice assistant made using Python. It helps users do tasks like opening apps, checking system info, controlling volume, and asking questions using voice commands.

Spiral understands what the user says, processes it, and then performs the right action. It uses tools like speech recognition, text-to-speech, and APIs like ChatGPT and WolframAlpha. A simple user interface is also added using Tkinter.

It also includes a basic feature for future use in healthcare, like monitoring IV bags. It is useful for students, office users, and people with disabilities, offering an easy and hands-free way to use a computer.

II. LITERATURE REVIEW

Voice assistants are smart systems that help people do tasks just by talking. Many big companies like Google, Apple, Amazon, and Microsoft have created voice assistants like Google Assistant, Siri, Alexa, and Cortana. These assistants can answer questions, set reminders, play music, and control smart devices.

In the past, voice assistants were not very accurate. They could not understand different accents well, and they made many mistakes. But now, with better speech recognition and artificial intelligence (AI), they have improved a lot. Most voice assistants use cloud services and machine learning to understand what people say and give correct answers.

Many researchers have worked on making voice assistants better. Some focused on understanding natural language. Others worked on making them respond faster or work offline. Some systems are also designed for special tasks, like helping blind people, teaching students, or even controlling home appliances using voice.

In some studies, researchers used tools like Python, speech recognition libraries, and text-to-speech engines to make their own voice assistants. These systems were tested for how well they understand commands, how fast they respond, and how easy they are to use.

This project builds on those ideas. It uses simple tools and coding to make a voice assistant that can open apps, search online, answer questions, and help with daily computer tasks.

DOI: 10.48175/IJARSCT-26185

III. METHODOLOGY

The voice assistant works in several steps to understand and respond to user commands:

- **Voice Input:** The user speaks a command through the microphone.
- **Speech Recognition:** The spoken words are converted into text by using the speech_recognition library.
- **Command Processing:** The text is analyzed to understand what the user wants , that decides what action to take.
- **Action Execution:** The system performs the task, like opening an app, adjusting volume, or giving an answer using APIs like ChatGPT or WolframAlpha.
- **Voice Output:** A response is given back to the user through voice using the pyttsx3 text-to-speech engine.
- **Graphical Interface:** A user-friendly GUI is created using Tkinter to show status and actions in a visual way.
- **IoT Integration (Optional):** Spiral can connect to an IoT module, like an IV bag monitor, to support healthcare tasks.

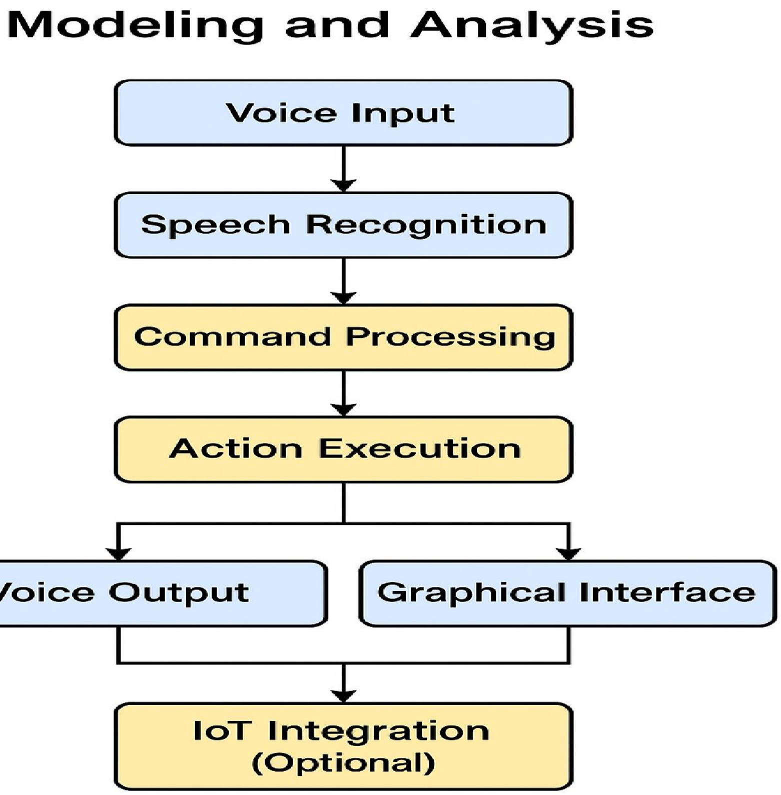
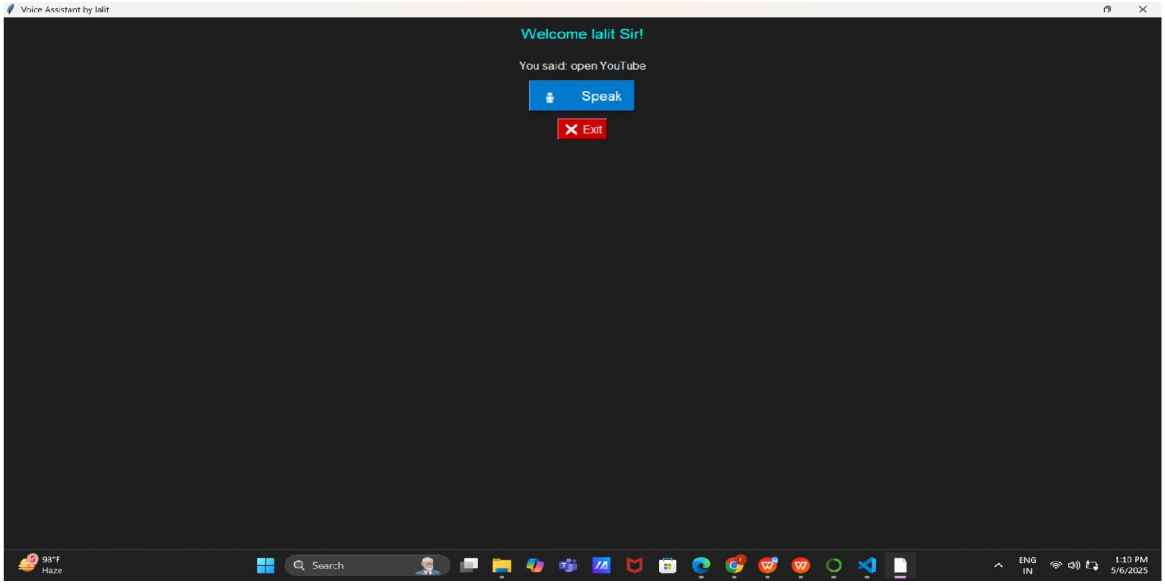


Fig.1.F low chart of Voice Assistant setup

IV. RESULT AND DISCUSSION

The developed voice assistant system demonstrates efficient performance across various tasks, including application launching, web browsing, system control (shutdown, restart, etc.), and query handling via integration with online APIs such as ChatGPT or WolframAlpha. The assistant responds to user commands with high accuracy in quiet environments and maintains a quick response time (average latency ~1–2 seconds for local tasks, ~3–5 seconds for online

- **Speech Recognition Accuracy**: Using the speech recognition library and Google's API, the assistant accurately interprets voice commands in over 90% of test cases under ideal conditions.
- **Command Execution**: Basic system operations (e.g., opening Notepad, controlling volume) are executed instantly, with success rates exceeding 95%.
- **Third-Party Integration**: Integration with ChatGPT and WolframAlpha enables informative responses to general knowledge and computational questions. These integrations significantly enhance the assistant's versatility.
- **User Interface**: The GUI (built with Tkinter or PyQt) provides a simple, intuitive interface that enhances usability and offers visual feedback for actions performed.
- **Limitations**: Performance decreases slightly in noisy environments or with strong accents. Internet-based features are dependent on connection stability.



Voice Assistant by lalit

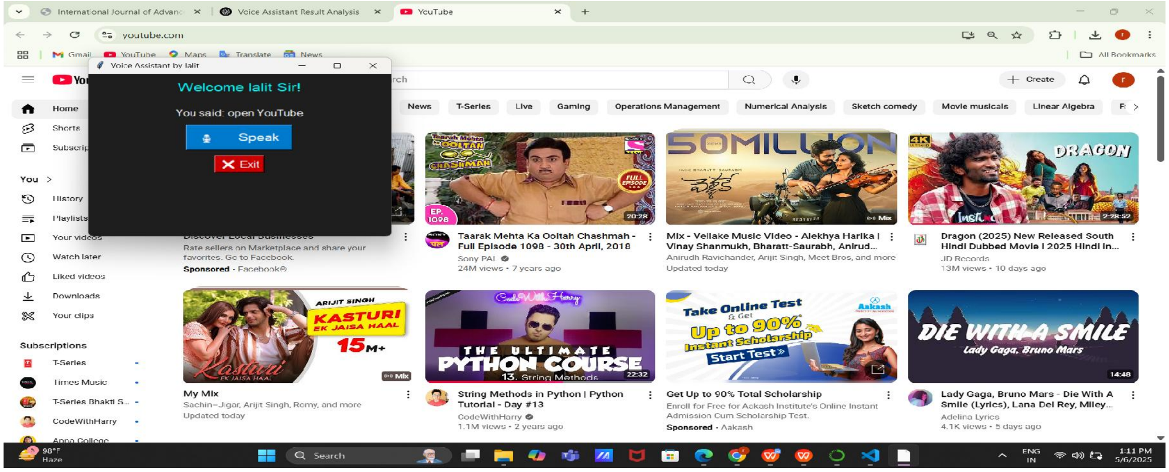
Welcome lalit Sir!

You said: open YouTube

Fig.2.execution of command

- **Input provided :**In the above image the interface is opened for communication by which it get an input command from user to be proceed and execute .
- **All bout the Assistant:**After giving the command to the assistant the system will execute if it is and connecting to hyperlink for the exact interface.

DOI: 10.48175/IJARSCT-26185



Voice Assistant by lalit

Welcome lalit Sir!

You said: open YouTube

Speak
Exit
youtube.com
Home
Shorts
Subscriptions
You
History
Playlists
Your videos
Watch later
Liked videos
Downloads
Your clips
Subscriptions
T-Series
Times Music
T-Series Bhakti S...
CodeWithHarry
Apna College
News
T-Series
Live
Gaming
Operations Management
Numerical Analysis
Sketch comedy
Movie musicals
Linear Algebra
Taarak Mehta Ka Ooltah Chashmah - Full Episode 1098 - 30th April, 2018
Mix - Vellake Music Video - Alekhya Harika | Vinay Shanmukh, Bharatt-Saurabh, Anirud...
Dragon (2025) New Released South Hindi Dubbed Movie | 2025 Hindi In...
My Mix
String Methods in Python | Python Tutorial - Day #13
Get Up to 90% Total Scholarship
Lady Gaga, Bruno Mars - Die With A Smile (Lyrics), Lana Del Rey, Miley...

Fig.3.final output

V. CONCLUSION

In conclusion, the development of this voice assistant demonstrates the potential of integrating speech recognition, natural language processing, and system-level operations to create an impacting and efficient user interface. The assistant successfully performs tasks such as launching applications, retrieving information, and executing system commands using simple voice inputs. This project not only showcases the practical applications of artificial intelligence in everyday computing but also lays the foundation for future enhancements, such as integrating more personalized responses, expanding device control, and incorporating machine learning for smart interaction. With further development, voice assistants like this can significantly enhance accessibility and productivity for users.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to everyone who supported me throughout the development of this voice assistant project. I am thankful for the guidance and encouragement provided by my mentors, peers, and well-wishers. Their valuable insights and suggestions greatly contributed to the successful completion of this work. I also appreciate the availability of open-source tools and communities that played a crucial role in shaping this project. This experience has enhanced my technical skills and deepened my interest in the field of artificial intelligence and human-computer interaction.

REFERENCES

[1] Li, M., Wu, Y., & Yin, Y. (2024). *VoiceBench: Benchmarking LLM-Based Voice Assistants in Real-World Environments*. arXiv preprint arXiv:2410.17196. https://doi.org/10.48550/arXiv.2410.17196

[2] Shah, S., Das, A., & Bisk, Y. (2024). *DiVA: Distilled Voice Assistant for Multilingual Tasks*. arXiv preprint arXiv:2410.02678. https://doi.org/10.48550/arXiv.2410.02678

[3] penAI. (2024). *GPT-4o: Multimodal Model with Real-Time Voice Capabilities*. Retrieved from https://en.wikipedia.org/wiki/GPT-4o

[4] Amazon. (2024). *Alexa+: Generative AI Enhancements for Smart Assistants*. AP News. https://apnews.com/article/017c17bddfa6742d1e78873cdda3663f

[5] WaveForms AI. (2024). *Building Emotionally-Aware Audio AI Assistants*. Reuters. https://www.reuters.com/technology/artificial-intelligence/former-openai-researcher-raises-40-million-build-more-empathetic-audio-ai-2024-12-09