e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:03/March-2024 Impact Factor- 7.868 www.irjmets.com

AI - VOICE DESKTOP ASSISTANT

Ms. Aarti Dharmani[*1], Mayuri Khatpe[*2], Priyanka Gayake[*3], Suhasini Sharma[*4]

[*1]Assistant Professor Data Science Usha Mittal Institute Of Technology SNDT Women's University, India.

[*2,3,4]Data Science Usha Mittal Institute Of Technology SNDT Women's University, India.

DOI : https://www.doi.org/10.56726/IRJMETS51616

ABSTRACT

AI desktop assistants like Apple's "SIRI" and Google's "Google Voice Search," can perform tasks and provide services based on user commands. These systems use speech recognition to respond to synthetic speech, allowing users to communicate with their devices. The proposed system, which can work with or without internet connectivity, uses voice recognition to process user input and provide various outputs. AI-based personal assistants aim to bridge the communication gap between humans and machines, creating a more engaging user experience.

Keywords: AI, SIRI, Google Voice Search, Speech Recognition, Internet, Personal Assistants, User Experience.

I. INTRODUCTION

AI assistants have a long history, starting with the Turing Test in the 1950s. Advances in machine learning, particularly deep learning and neural networks, led to breakthroughs like OpenAI's GPT-3 in 2020. Today, AI assistants are used in customer service, translation, content generation, and personal virtual assistants like Siri and Alexa.

The project aims to create an AI desktop assistant that speaks like humans, improving user experience, enhancing task execution, making technology more accessible, inclusive, personalized, and pushing the boundaries of natural language processing.

The research aims to enhance natural language understanding in AI desktop assistants, train models effectively, mitigate biases, and improve user experience, promoting inclusivity, enhancing productivity, and pushing NLP capabilities.

The research paper explores the development of artificial intelligence desktop assistants, focusing on recent developments and techniques. It presents a literature survey in section II, discusses the proposed work in section III, and analyzes the outcomes. Section V presents the results and discussion that provides insights into the assistant's limitations and efficacy. Section IV gives the implementation details including the design and implementation of the virtual desktop assistant along with the corresponding results. The study concludes with a conclusion and future directions for further research in Section V. A list of references is provided at the end of the paper, highlighting the sources that provided information and support for the research project.

II. LITERATURE SURVEY

In the realm of technology, researchers are continuously advancing the capabilities of virtual assistants and AI- driven communication systems. Leandro Tibola and Liane Margarida Rockenbach Tarouco [1] emphasize the importance of interoperability in virtual worlds, highlighting the role of WWW services using HTTP and XML to enhance communication between virtual and real-world entities while bolstering security measures against modern operating systems.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever [2] showcase significant advancements in natural language understanding through generative pre-training and discriminative fine-tuning. Their task-agnostic model outperforms discriminatively trained models in various language understanding tasks, marking a notable stride in language processing capabilities.

Deepak Shende, Ria Umahiya, Monika Raghorte, Aishwarya Bhisikar, and Anup Bhange [3] present an AI-based voice assistant project implemented using Python, leveraging open-source software modules and community support to ensure adaptability to future updates.

Sangpal, Ravivanshikumar, Gawand, Tanvee, Vaykar, Sahil, and Madhavi, Neha [4], explore the integration of gTTS, AIML, and Python in their interpretation of JARVIS, highlighting the benefits while acknowledging

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:03/March-2024 Impact Factor- 7.868 www.irjmets.com

dependencyissues on specific platforms.

Rajat Sharma and Adweteeya Dwivedi [5] introduce"JARVIS," an AI voice assistant system employing speech recognition, gTTS, neural networks, and natural language processing to deliver intelligent and responsive interactions tailored to specific circumstances.

Afra Ali, Shweta Dubey, Shyam Dwivedi, Divisha Pandey, Md. Saif Raza, and Muskan Srivastava [6] unveil a voice assistant service for desktop users, integrating internet- of-things technology, speech recognition, and modern AI technologies to provide enhanced functionalities and a seamless user experience.

Vedant Kulkarni, Shreyas Kallurkar, Vipul Waikar, Saurabh Patil, and Swarupa Deshpande [7] present a framework for a virtual assistant that overcomes existing constraints, promising improved effectiveness and usability in processing voice inputs.

Dr. C. K. Gomathy, Redrouthu Venkata Narayana, Thota Vamsi Khrishna, and DR. V. Geetha [8] contribute to automated communication systems with an artificial intelligence chatbot using Python, showcasing the potentialof leveraging technologies like Pip, NumPy, TensorFlow, and random, albeit with limited functionality for specific queries and conversation types.

T R and Mahesh [9] propose a personal AI desktop assistant utilizing Python's speech recognition and text-to-speech libraries, aiming to enhance user convenience and productivity in daily computer-related tasks.

Finally, Nagappan, Umapathi, Ganesan, Karthick, Venkateswaran, Natesan, Ramalingam, Jegadeesan, and Srinivas, Dava [10], contribute to the discourse with the development of a desktop virtual assistant using Python, leveraging speech recognition, APIs, and text-to-speech capabilities to bridge the gap between human and computer interactions.

As the literature survey unfolds, a tapestry of innovation and exploration emerges, weaving together diverse perspectives and methodologies in pursuit of more intuitive, responsive,and inclusive AI-driven solutions.

III. PROPOSED WORK

In this section, we present our proposed work aimed atdeveloping an advanced AI-driven voice assistant system leveraging Python and various modern technologies. Building upon the foundational research conducted by experts in the field, our project endeavors to create a versatile and intelligent voice assistant capable of seamlessly integrating with users' daily routines while offering enhanced functionalities and user experiences.

A. Methodology

To achieve our objectives, we will adopt a multi-faceted methodology that encompasses several key components. The flowchart given below for an overview of the overall steps in the methodology:

The various steps involved in the methodology are briefly explained as follows:

1) **Installing necessary libraries:** We began by installing the necessary Python libraries and modules. It's essential to prioritize the important libraries that are crucial for the functionality of virtual assistant. Some of them are as follows:

- **Speech Recognition:** Converts speech to text, enablingthe AI assistant to understand spoken commands.
- **Pyttsx3:** Converts text to speech, allowing the AI assistant to provide spoken responses to users.
- **Datetime:** Provides access to current date and timeinformation, facilitating time-based functionalities such as setting alarms or providing time-related responses.
- **OS:** Enables interaction with the operating system, allowing the AI assistant to perform system-relatedtasks such as file operations or launching applications.
- **Pyaudio:** Handles audio input and output, essential for voice-based applications like speech recognition and voice assistants.
- Additionally, other important libraries may include:
- **Wikipedia:** Conducts searches on Wikipedia, providingaccess to vast amounts of information.
- **Requests:** Sends HTTP requests, useful for accessing web-based data or APIs.
- **Webbrowser:** Allows for opening web browsers ordisplaying web-based documents directly from code.
- **Random:** Generates pseudo-random numbers, facilitating randomness in various actions.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:03/March-2024 Impact Factor- 7.868 www.irjmets.com

- **Beautiful Soup:** Extracts data from XML and HTMLfiles, useful for web scraping tasks.
- And so on for additional libraries.

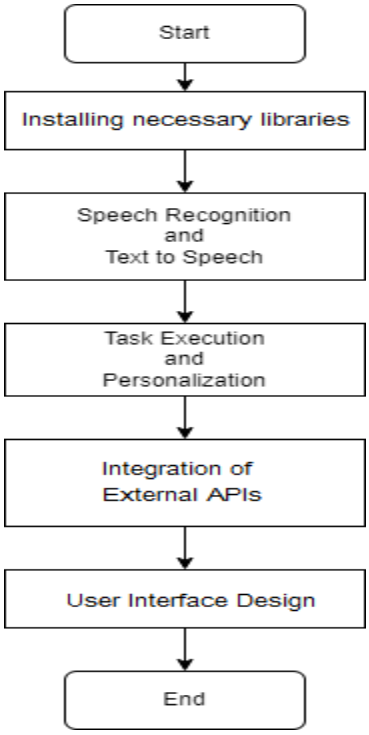

Figure 1: The flowchart details entails the various steps involved in thedevelopment of virtual desktop assistant.

2) **Speech Recognition and Text to Speech:** We implemented a speech recognition system that converts spoken language into text. For this we will use the Speech Recognition library in Python. Along with thiswe implement a text-to-speech system that converts the assistant's responses (text) into natural-sounding speech.
3) **Task Execution and Personalization:** Develop modules to perform various tasks based on user intents, such as opening websites, playing music, asking time and date, etc.
4) **User Interface Design:** A user-friendly interface willbe developed to facilitate intuitive interactions with the voice assistant, ensuring a smooth user experience acrossvarious devices and platforms.
5) **Integration of External APIs:** Integrate external APIs for accessing relevant information and services, such as weather forecasts, news updates, and online search functionalities, to enrich the assistant's capabilities.

B. System Design

The virtual desktop assistant takes voice commands, recog- nizes them, processes them, and executes the requested task.It responds dynamically to user input, executing tasks basedon their desires, as depicted by the data flow diagrams (DFD) of Figure 2 and Figure 3 respectively.

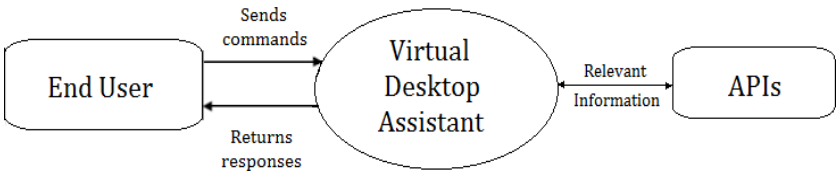

Figure 2: The diagram depicts Level 0 DFD of a end user-virtual assistant interaction.

The user sends a command, the assistant processes it taking the relevant information using APIs and the user receives a response. The diagramshows the user's role in the communication process, with the assistant using various APIs.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:03/March-2024 Impact Factor- 7.868 www.irjmets.com

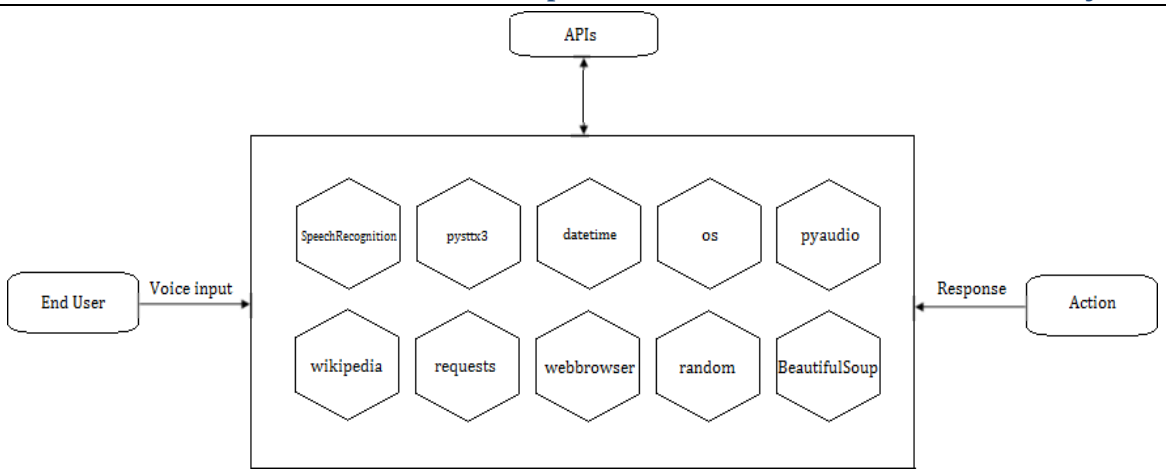

Figure 3: The diagram depicts Level 1 DFD illustrating end user input process and subsequent system response and actions.

IV. IMPLEMENTATION AND RESULTS

A. Implementation Details

For implementation of the voice desktop assistant following are the minimum **hardware configurations** that are required:

- **Processor:** The program needs a processor that is at least as powerful as an Intel Core 2 Duo. Although this is the bare minimum, a more potent CPU is advised for lag-free performance. Performance will be much enhanced with a multi-core processor, like an Intel Core i5 or i7, especially for jobs requiring voice recognition and artificial intelligence.
- **RAM (Random Access Memory):** A minimum of 6 GB of RAM is required. However, to handle resource-intensive tasks and ensure smooth multitasking, it's ad- visable to have more RAM, such as 8 GB or 16 GB. The amount of RAM affects the software's ability to process voice commands and AI tasks efficiently.
- **Hard Drive (HDD/SSD):** A minimum of 256 GB of storage space is essential for software installation, data storage, and updates. Consider using a Solid State Drive (SSD) instead of a Hard Disk Drive (HDD) for faster data access and improved overall system performance. Additionally, having extra storage space is beneficial if you plan to store a significant amount of data.

The minimum **software configurations** required are as fol- lows:

- **Operating System:** This software is compatible with Windows 11 (64-bit) to access the latest features and security updates and ensure that your system meets the required requirements.
- **Integrated Development Environment (IDE):** Visual Studio Code (VS Code) is a free, open-source code editor that supports Python and offers real-time functionality, creating an efficient environment for code development, debugging, and collaboration.
- **Programming Language (Python):** The software, de- veloped using Python, is suitable for AI and voice recog- nition tasks and requires Python version 3.9.11 for proper functionality.

Why Choose Python Language for Vikram? Python, originating before the surge in popularity of machine learning and AI, remains a preferred choice owing to its distinctive attributes, which distinguish it from other programming languages. These qualities include:

1) **Rich Collection of Packages and Libraries:** Py- thon boasts an extensive array of packages and libraries, which, despite its inception predating the rise of machine learning and AI, render it invalu- able for these domains.
2) **Enhanced Code Readability:** Python's elegant and concise syntax significantly aids machine learning endeavors by simplifying program composition and enhancing readability, facilitating comprehension and maintenance.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:03/March-2024 Impact Factor- 7.868 www.irjmets.com

3) Versatility and Flexibility: Python's inherent ver- satility empowers developers to tackle complex tasks efficiently with minimal overhead, making it an ideal choice for diverse applications and rapid prototyping.

The enduring popularity of the Python language is evident, as illustrated in Figure 4.

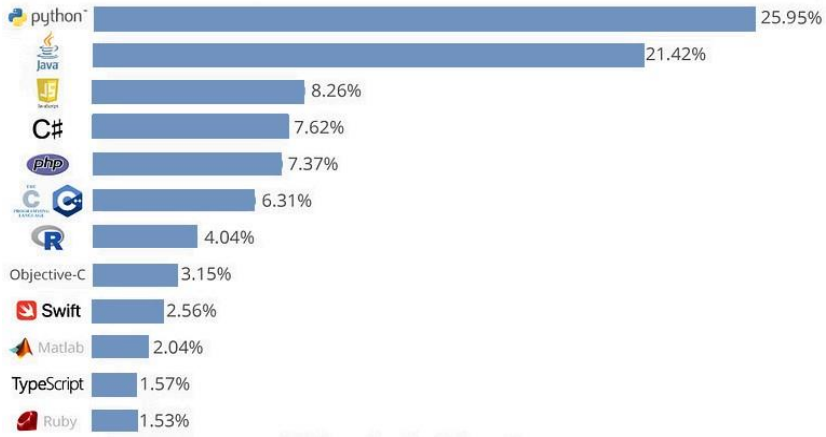

Figure 4: The above figure showcases the top 10 most popular programminglanguages globally, including Python, Java, and JavaScript.

B. Results

Following figures of provides visual results of our AI assistant's functionality and user interface, highlighting user interaction flow, interface design, and tasks executed.

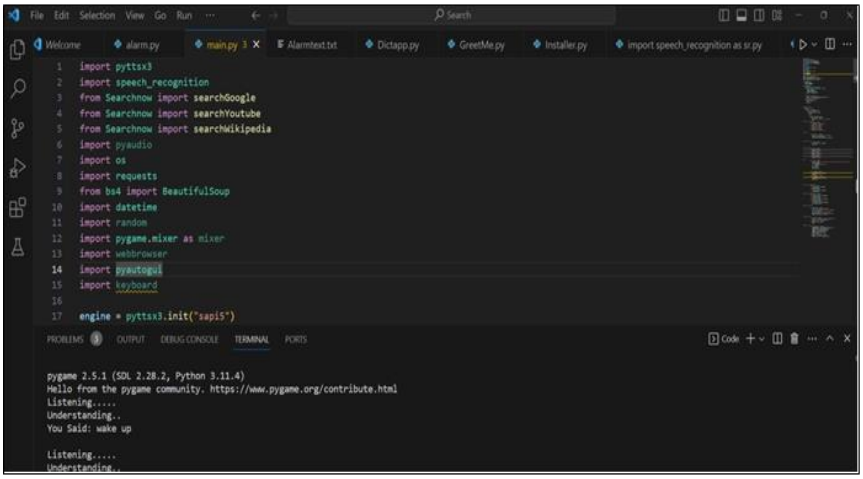

Figure 5: Greets the user based on the time of day and asks how it can assist

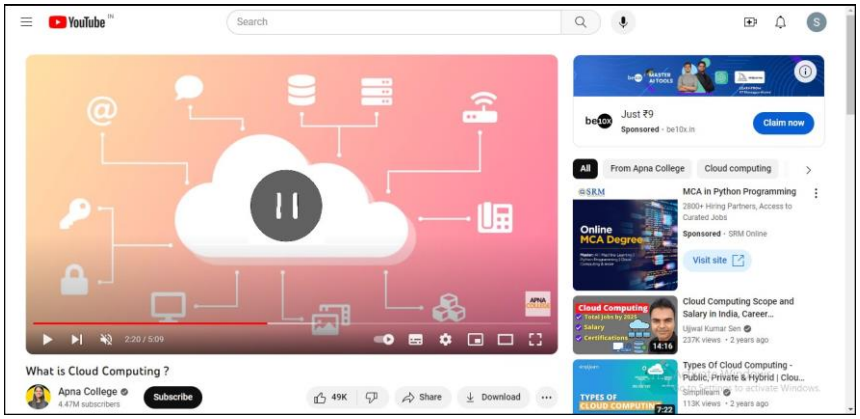

Figure 6: Controlling video playback on YouTube; allows user to perform actions like pausing, playing, seeking backward or forward, toggling fullscreen, muting, etc. Here on user's request to "pause" the video, the assistant pauses the video.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:03/March-2024 Impact Factor- 7.868 www.irjmets.com

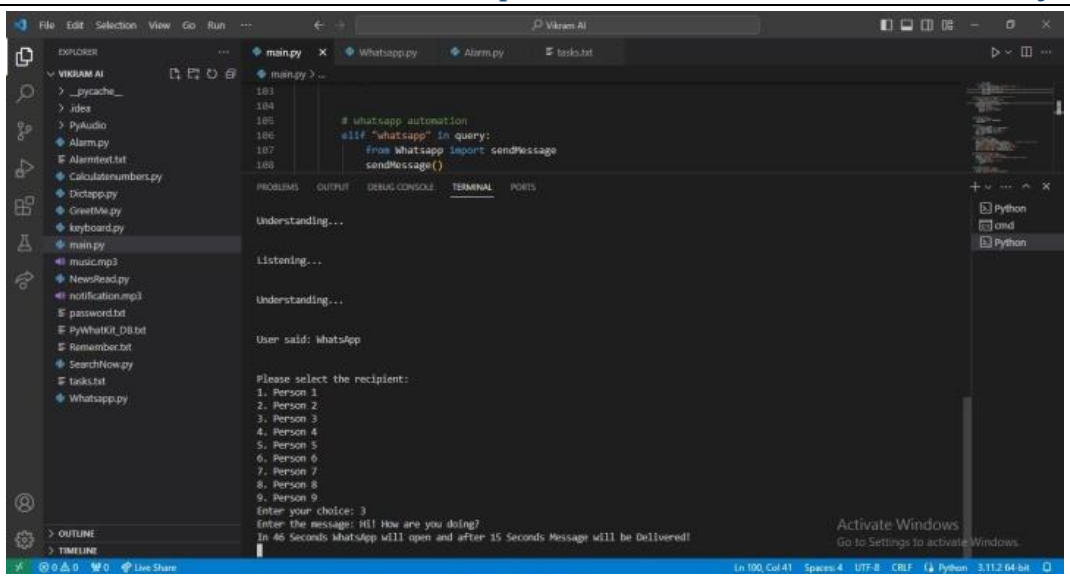

Figure 7: On saying "WhatsApp", the assistant enables the user to send WhatsApp messages to specified contacts by dictating the message via voice input.

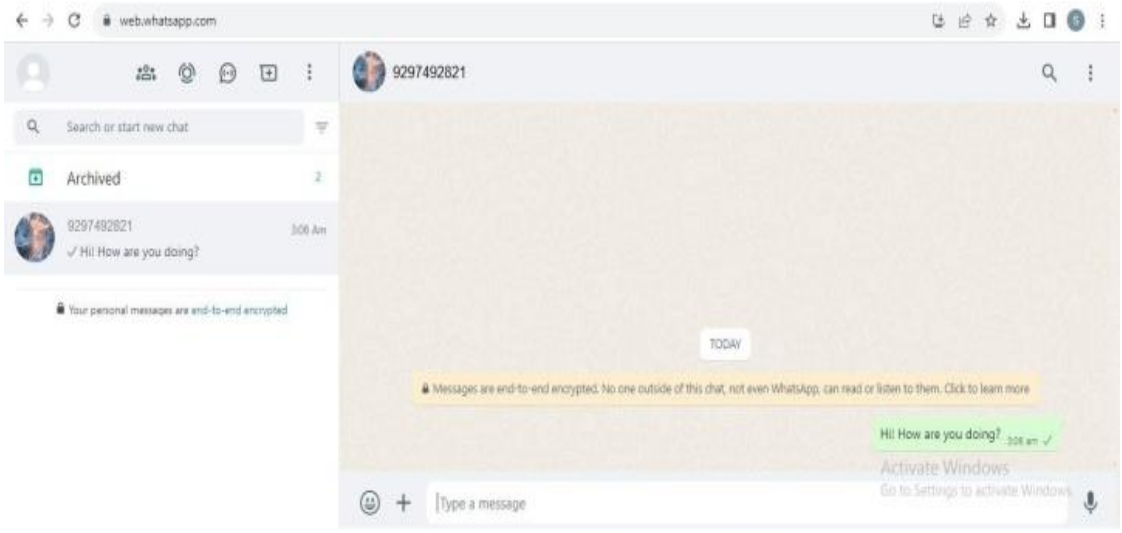

Figure 8: As it can be seen the WhatsApp message is sent to the specified person.

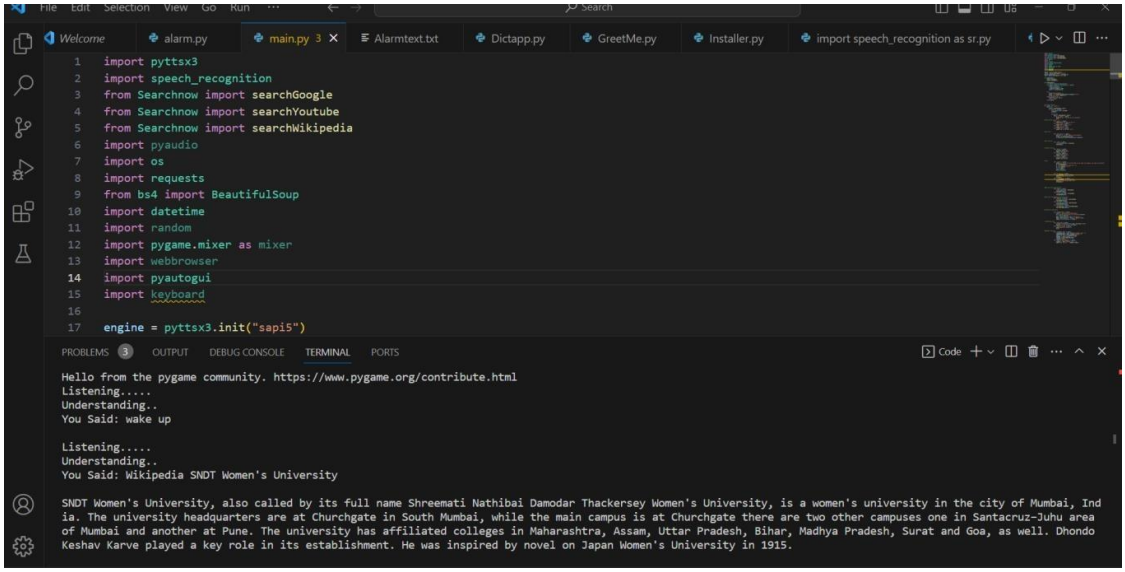

Figure 9: On saying "Wikipedia (query)", the assistant gives the relevantinformation for the given query

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:03/March-2024 Impact Factor- 7.868 www.irjmets.com

V. CONCLUSION

AI desktop assistants are revolutionizing human-machine interactions, offering benefits across sectors like customer service, healthcare, and education. They streamline tasks, automate processes, and provide personalized support, enhancing productivity and efficiency. In customer service, they handle inquiries and provide personalized assistance. In healthcare, they automate administrative tasks, aid in diagnosis and treatment planning, and facilitate communication. In education, they offer personalized learning experiences. Beyond these sectors, AI desktop assistants drive innovation across finance, manufacturing, and research and development. Their adaptability and ability to adapt to user needs make them significant catalysts for progress in the digital age.

VI. FUTURE SCOPE

While AI desktop assistants have already demonstrated their value, there are several avenues for future development and enhancement to fully unlock their potential considering the factors of limitations, security and stability of the system. Hereare some suggestions for future scope:

- **Offline Functionality:** Enhances utility and accessibility in diverse environments without internet connection.
- **Enhanced Security Features:** Implements encryption, multi-factor authentication, and biometric recognition foruser data protection.
- **AI Model Training:** Continuously refines AI models for improved accuracy, responsiveness, and natural languageunderstanding.
- **Voice Recognition Improvement:** Invests in research and development for better understanding of diverse accents and environments.
- **Integration with Smart Home Devices:** Enables seam- less control and automation of connected devices.
- **User Feedback Mechanisms:** Gathers feedback for con- tinuous improvement of features, usability, and user ex- perience.

By addressing these areas for future development, we can further elevate the capabilities, security, and user experience of AI desktop assistants, ensuring their continued relevance and effectiveness in meeting the evolving needs of users across various domains.

Divisha Pandey, Afra Ali, Shweta Dubey, Muskan Srivastava, Shyam Dwivedi, Md. Saif Raza (2022). "Voice Assistant Using Python and AI."

ACKNOWLEDGMENT

First, we would like to thank Professor Rajesh Kolte, Head of Department (Data Science) and our guide, Ms. Aarti Dharmani for her valuable guidance and continuous support during the project; her patience, motivation, enthusiasm, and immense knowledge. Her direction and mentoring helped us to work successfully on the project topic.

Our sincere gratitude to Dr. Yogesh Nerkar, Principal (Usha Mittal Institute of Technology) for his valuable encouragement and insightful comments.

We would also like to thank to all the teaching and non-teaching staff for their valuable support.

Last but not the least we would like to thank to our parents and friends.

VII. REFERENCES

[1] Leandro Tibola, Liane Margarida Rockenbach Tarouco (2013). "Interop- erability in Virtual World."

[2] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever (2018). "Improving Language Understanding by Generative PreTraining."

[3] Deepak Shende, Ria Umahiya, Monika Raghorte, Aishwarya Bhisikar, Anup Bhange (2019). "AI Based Voice Assistant Using Python."

[4] Sangpal, Ravivanshikumar, Gawand, Tanvee, Vaykar, Sahil, Madhavi, Neha (2019). "JARVIS: An interpretation of AIML with integration of gTTS and Python."

[5] Rajat Sharma, Adweteeya Dwivedi (2022). "JARVIS - AI Voice Assist- ant."

[6] Vedant Kulkarni, Shreyas Kallurkar, Vipul Waikar, Saurabh Patil, Swarupa Deshpande (2022). "Virtual Assistant Using Python."

[7] Dr. C. K. Gomathy, Redrouthu Venkata Narayana, Thota Vamsi Khrishna,DR. V. Geetha (2022). "Artificial Intelligence Chatbot using Python."

[8] T R, Mahesh. (2023). Personal A.I. Desktop Assistant. International Journal of Information Technology, Research and Applications. 2. 54-60.

[9] Nagappan, Umapathi, Ganesan, Karthick, Venkateswaran, Natesan, Ramalingam, Jegadeesan, Srinivas, Dava (2023). "DESKTOP'S VIR-TUAL ASSISTANT USING PYTHON."