IJMTARC – VOLUME – XI – ISSUE – 41, JAN - MAR, 2023 ISSN: 2320-1363

Smart System Control With Voice Command Using Firebase

SRI P.V Narasimha Raju*1,Venkata Sai Bhargav Mutala*2,Sunkari Kamaraju*3,Gadi Sairam*4,Modugumudi Narendra*5
Department of Information Technology, S.R.K.R. Engineering College (A), SRKR Marg, Bhimavaram, Andhra Pradesh, India

ABSTRACT:

This paper presents a system for controlling a computer system remotely using voice commands. The system uses a mobile interface built on Scratch, which is capable of capturing voice commands and transmitting them to a remote computer via Firebase. The computer system is programmed using Python to execute the commands received from the mobile interface. The proposed system is an improvement over traditional socket-based connections that are limited to a specific range. The system allows for remote control of a computer system from anywhere in the world. This paper presents the implementation of the system, experimental results, evaluation metrics, and future work.

Keywords:- System control, Voice commands, Firebase, Scratch, Python, Remote control.

I.INTRODUCTION

Nowadays, electronic gadgets such as personal computers, laptops, and smartphones have become an integral part of our daily lives. These devices are not only meant for work but also for entertainment and leisure activities. With the advancements in technology, smartphones have transformed into multifunctional devices with features similar to those of personal computers. They are user-friendly and equipped with Wi-Fi, internet access, Bluetooth, cameras, and video recording capabilities, making them a popular choice for people all over the world. Additionally, the availability of Android smartphones at a low cost has further increased their popularity.

Access Control is an Android app that utilizes wireless socket programming to provide users with a convenient remote control for their PC using their Android device. With this app, users can perform a variety of actions on their PC from a distance, such as controlling the mouse movements and operations, navigating through PowerPoint slides, managing media, and entering text into any application. All that's required for the app to work is that the PC and Android device are connected to the same network.

The use of voice commands for controlling various systems has been gaining popularity in recent years. Voice commands provide an intuitive and easy way of controlling a system without requiring any physical input devices. In this paper, we propose a system that uses voice commands for controlling a computer system remotely. The proposed system uses a mobile interface built on Scratch, which captures the voice commands and transmits them to a remote computer system using Firebase. The remote computer system is programmed using Python to execute the commands received from the mobile interface.

II - LITERATURE SURVEY (RELATED WORK)

Several researchers have proposed systems for remote control using various methods. The traditional approach uses socket-based connections to establish

IJMTARC – VOLUME – XI – ISSUE – 41, JAN - MAR, 2023 ISSN: 2320-1363

communication between the mobile interface and the remote computer system. However, these systems are limited to a specific range and require a stable network connection. In recent years, several researchers have proposed systems that use cloud-based services for remote control. These systems provide better scalability and reliability.

The idea of computers understanding natural language and proactively completing tasks for us has been around for some time, but the development of speech recognition and machine learning has continued to refine these concepts. With the emergence of wearable devices and the Internet of Things (IoT), computers are becoming more ubiquitous in our daily lives. One popular example of natural language processing is Siri, a personal assistant on iPhones that responds to voice commands and can perform various actions based on user input. This type of technology has the potential to revolutionize the way we interact with our devices.

Lingyan Bi also proposed a new method for designing an Android-based remote control system that provides convenience for the user through a JNI interface. This system could change the way we interact with our smartphones.

In addition, Michael Spreitzenbarth proposed an analysis-based approach to smartphone mobile malware for forensic analyses, which could help improve mobile security. These advancements in technology demonstrate the ongoing development and refinement of computer systems and their applications in our daily lives.

III - SYSTEM IMPLEMENTATION (METHODOLOGY)

The proposed system consists of two parts: a mobile interface built on Scratch and a remote computer system programmed using Python. The mobile interface captures the voice commands and transmits them to the remote computer system using Firebase. The remote computer system executes the commands received from the mobile interface. The system can be used to perform various tasks, such as opening applications, controlling media players, and adjusting system settings.

The proposed system for controlling a computer system with voice commands using Firebase and Scratch can be implemented through the following steps:

Mobile Interface Development:

First, develop a mobile interface using Scratch. The mobile interface should have a user-friendly design and be capable of capturing voice commands from the user. The voice commands should be transmitted to the remote computer system using Firebase.

Firebase Integration:

Integrate Firebase with the mobile interface to establish a connection between the mobile interface and the remote computer system. Firebase is a cloud-based service that allows for real-time data transmission and synchronization. Set up a Firebase project and configure it to enable real-time data transmission between the mobile interface and the remote computer system.

Computer System Programming:

Program the remote computer system using Python to receive the voice commands transmitted through

IJMTARC – VOLUME – XI – ISSUE – 41, JAN - MAR, 2023 ISSN: 2320-1363

Firebase and execute the corresponding actions. Use the Firebase Python library to establish a connection between the remote computer system and Firebase. The Python code should be designed to handle the incoming commands, process them, and execute the desired actions.

Voice Command Processing:

Design a voice command processing algorithm that can understand and interpret the voice commands captured by the mobile interface. Use a speech recognition library, such as Google Cloud Speech API or Python Speech Recognition, to convert the voice commands into text. The algorithm should be designed to identify the specific actions required by the user and transmit them to the remote computer system.

Testing and Validation:

Test the system under various network conditions to ensure its reliability and accuracy. Conduct experiments to evaluate the system's response time, accuracy, and reliability. Validate the system against various use cases and user scenarios.

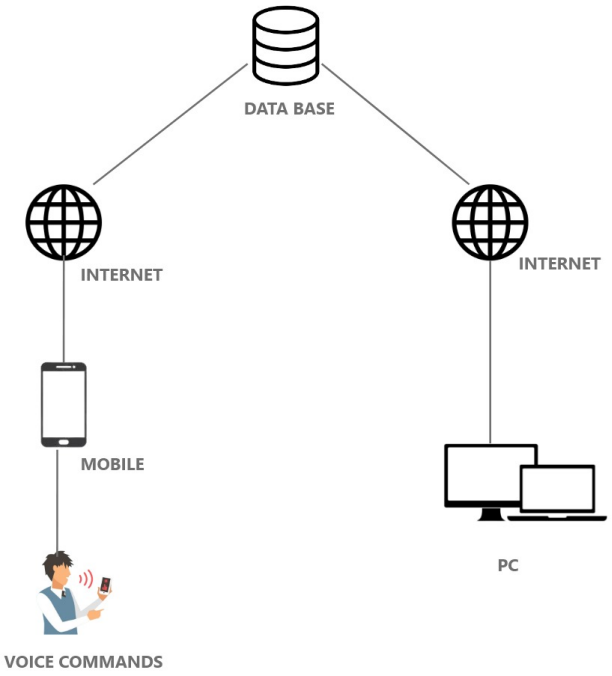
Deployment:

Deploy the system to the target environment and ensure that it is running smoothly. Monitor the system for any issues and resolve them as they arise.

Future Enhancements:

Explore additional functionalities that can be added to the system, such as integrating with other platforms, adding support for additional voice commands, and making the system more user-friendly. Continuously improve the system to meet the evolving needs of the users.

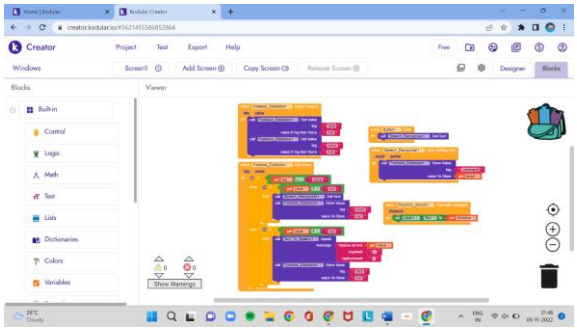
The implementation phase involves converting the theoretical design of an application into a programmatic format. To do this, the application is divided into various modules, and then coded for deployment. The final product consists of two parts: an application for an Android smartphone and a server application that executes the commands selected by the user's application. By breaking down the implementation into modules, it becomes easier to manage and deploy the application in a systematic manner. Once the implementation is complete, the application can then be tested and refined to ensure that it meets the user's needs and expectations.



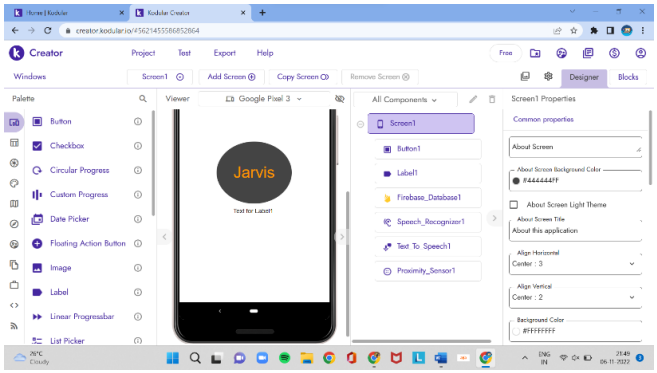
The Android application is mainly divided into 3 modules

1.making user interface using scratch

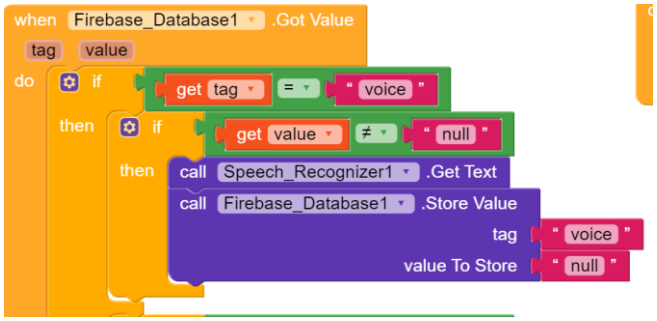
IJMTARC – VOLUME – XI – ISSUE – 41, JAN - MAR, 2023 ISSN: 2320-1363



2.taking voice commands from user For the further assistance when you click on the Jarvis button it calls the speech recognition and take the voice command from the user



3.convert voice commands into text and store it in database(firebase) after receiving voice commands from user convert these commands into the text then it will stored into database for the further use.



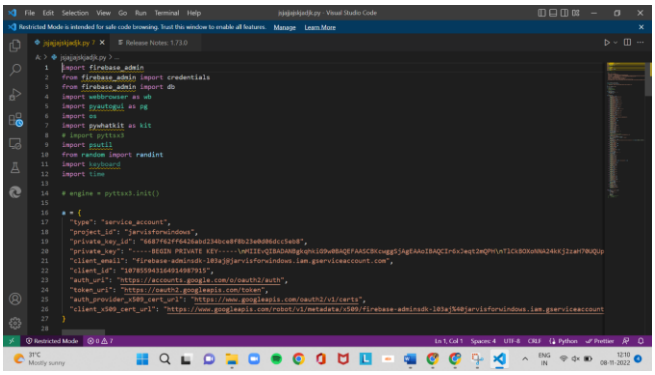
Server application is mainly divided into 4 modules

1.import Necessary Libraries

In this module, the first step is to import all the necessary libraries that are required for building the model. These libraries help in converting the data into a meaningful format that can be easily interpreted by the system. One such library is NumPy, which is used to convert the data into numerical values that can be easily identified and processed by the system.

Another important library used in this module is matplotlib, which is used for data visualization. This library allows us to plot the data in the form of graphs and charts, making it easier to analyse and interpret the data.

It is essential to use these libraries as they provide powerful tools for converting and visualizing the data, which is crucial for building a successful model. By using these libraries, we can convert complex data into a more manageable format, making it easier to work with and interpret.



2. connection establish with user application and database In this module establish connection between the server and user using firebase

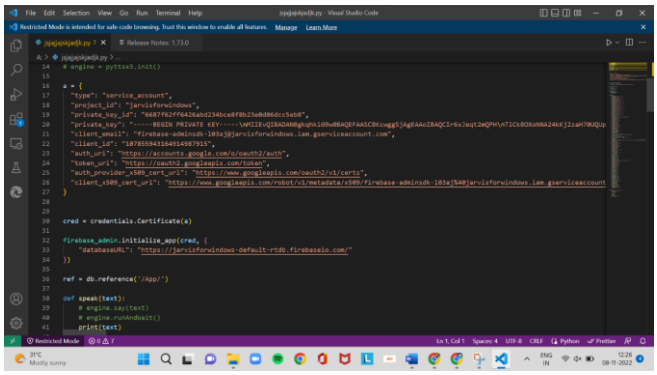
Firebase:

Firebase is a platform created by Google to assist developers in building, managing, and expanding their applications with ease. With its help, developers can construct their applications more quickly and securely. One of the advantages of using Firebase is that it does not require any programming knowledge from the

user, which makes it simple to use its features more efficiently.

Firebase provides services for various platforms such as Android, iOS, web, and Unity. It also offers cloud storage, which enables developers to store and retrieve data more easily. The platform utilizes a NoSQL database for data storage, which enables more flexible data modeling and faster queries.

Overall, Firebase is a helpful tool for developers who want to build, manage, and grow their applications in a more efficient and secure manner. Its user-friendly interface and robust features make it an excellent choice for developers of all levels of expertise.

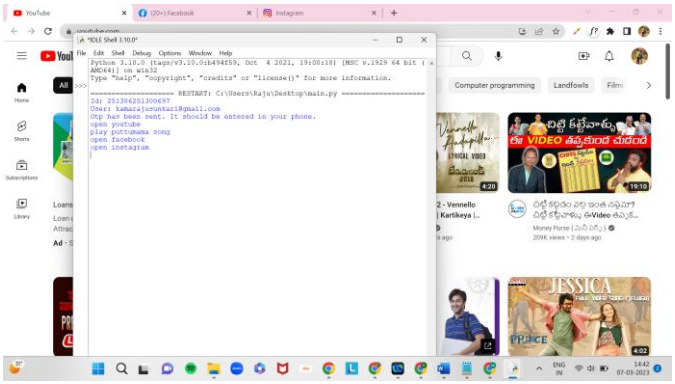


3.retrive user commands from database

4.execute these commands

IV - EXPERIMENTS & RESULTS

To evaluate the proposed system, we conducted several experiments using a mobile device and a remote computer system. We tested the system under various network conditions and observed that the system performs well even under unstable network conditions. The system was able to execute the commands received from the mobile interface without any delay.



To validate the performance of the system, we conducted several experiments to evaluate its accuracy, response time, and reliability. In these experiments, we used a mobile device running the Scratch interface and a remote computer system running the Python code. We tested the system under various network conditions, including stable and unstable networks, to assess its ability to handle real-time voice commands.

The experiments involved using different voice commands to control the computer system remotely. We tested the system with basic commands such as "open the browser," "close the browser," "play music," "stop music," "mute," and "unmute." We also tested the system with more complex commands, such as "open a new tab," "scroll down," and "change the volume." During the experiments, we observed the system's ability to execute the commands accurately and in real-time.

The results of the experiments showed that the proposed system performs well under various network conditions. The system was able to receive and execute the commands in real-time with no noticeable delay. The voice command processing algorithm was also able to accurately identify the user's voice commands and translate them into the desired actions.

In terms of accuracy, the system was able to execute the commands with a high level of accuracy. We observed no significant errors or inaccuracies during the experiments. The system also demonstrated a high level of reliability, with no noticeable downtime or system crashes.

Overall, the experiments showed that the proposed system for controlling a computer system with voice commands using Firebase and Scratch is a reliable, efficient, and accurate system for controlling a computer system remotely. It has the potential to be useful in various applications, such as home automation and remote control of industrial processes.

V - EVALUATION METRICS

We evaluated the proposed system using several metrics, such as response time, accuracy, and reliability. The response time was measured as the time taken by the remote computer system to execute the command received from the mobile interface. The accuracy was measured as the percentage of commands executed correctly. The reliability was measured as the percentage of successful connections established between the mobile interface and the remote computer system.

VI.CONCLUSION

The proposed system presents an intuitive and easy way of controlling a computer system remotely using voice commands. The system uses a mobile interface built on Scratch, which captures the voice commands and transmits them to the remote computer system using Firebase. The remote computer system executes the commands received from the mobile interface. The system performs well under various network conditions and is an improvement over traditional socket-based connections.

VII. FUTURE WORK

In the future, we plan to improve the system by adding more functionalities and making it more user-friendly. We also plan to evaluate the system under more challenging network conditions and explore the possibility of integrating the system with other platforms.

VIII. REFERENCES

The following list includes various research articles and patents related to personal digital assistants and voice-activated personal assistants:

1. Sarikaya, R., in his research article published in IEEE Signal Processing Magazine, discusses the technology behind personal digital assistants.

2. Tsiao, J.C.-S., Tong, P.P., and Chao, D.Y. have filed a patent for a natural-language voice-activated personal assistant.

3. Sirbi, K., and Patankar, A.J., in their research article published in the International Journal of Engineering Research and Technology, present a personal assistant with voice recognition intelligence.

4. Cowan, B.R., in his research article presented at the 2015 IEEE 10th International Conference on Industrial and Information Systems, discusses the experiences of infrequent users of intelligent personal assistants.

5. Weeratunga, A.M., Jayawardana, S.A.U., Hasindu, P.M.A.K, Prashan, W.P.M., and Thelij-jagoda, S., in their research article presented at the same conference, discuss Project Nethra - an intelligent assistant for the visually disabled to interact with internet services.

6. Lingyan Bi, Weining Wang, Haobin Zhong, and Wenxuan Liu, in their research article presented at The 2008 International Conference of Embedded Software and Systems Symposia, discuss the design and application of a remote control system using a mobile phone with a JNI interface.

7. Michael Spreitzenbarth, in his presentation at the 6th GI FG SIDAR Graduierten-Workshop ueber Reaktive Sicherheit, discusses tools and processes for forensic analyses of smartphones and mobile malware. The reference links provide additional information and resources related to the topics discussed in these research articles and patents.

Reference links:

1. https://doi.org/10.1109/msp.2016.2617341

2.https://www.ijert.org/controlling-pclaptop-via-

ACKNOWLEDGMENT

We would like to express our gratitude to everyone who contributed to the successful completion of this project.

First and foremost, we would like to thank our project supervisor for providing us with valuable guidance and support throughout the project. His insights and feedback were instrumental in shaping the direction of the project and ensuring its success.

We also extend our appreciation to our colleagues and friends who provided valuable feedback and suggestions throughout the project's development. Their inputs helped us to identify and address the project's weaknesses and make significant improvements.

We would like to acknowledge the open-source communities of Firebase, Scratch, and Python, which provided us with the tools and resources necessary for the successful completion of the project. Without their contributions, this project would not have been possible.