IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⫶ Peer-reviewed & Refereed journal ⫶ Vol. 13, Issue 8, August 2025

DOI: 10.17148/IJIREEICE.2025.13810

VOICE ASSISTANT HOME AUTOMATION

G S Sunitha[1], Alfiya Taneem[2], OM Ganesh[3], Prajwal R H [4], Sathish P Nidagal[5]

Assistant Professor, E&CE, BIET, Davangere, Karnataka India[1]

Student, UG, E&CE, BIET, Davangere, Karnataka India[2,3,4,5]

Abstract: This project presents a hybrid home automation system using the ESP32 microcontroller, combining voice control (via Google Assistant and IFTTT) and a custom mobile app to manage appliances like lights and fans. It enhances convenience, especially for the elderly and differently-abled, by enabling hands-free and remote operation. The system uses relay modules for device control and connects to Firebase Realtime Database for cloud-based monitoring and real-time status updates. It offers reliable performance even during internet outages and supports both individual and group device control, making it a flexible and user-friendly IoT solution for modern smart homes.

Keyword: hybrid home automation system, Firebase Realtime Database, real-time status updates.

I. INTRODUCTION

Home automation systems are gaining popularity due to their ability to provide remote control and intelligent management of household appliances. This project introduces a hybrid home automation model that combines multiple control modes, including voice commands and mobile applications, to operate devices like lights and fans. The ESP32 microcontroller, known for its Wi-Fi capabilities, is at the heart of the system, which integrates with cloud platforms like Firebase for real-time monitoring and remote control. Google Assistant for voice control through IFTTT allows users to manage appliances using simple spoken commands. The custom mobile app ensures real-time status updates and synchronization of commands. This hybrid approach promotes a more accessible and efficient lifestyle.

II. OBJECTIVES

The objectives of the project:
- Design a hybrid home automation system using ESP32.
- Integrate voice control, and mobile app.
- Enable remote access and real-time monitoring via Firebase.

III. METHODOLOGY

A. BLOCK DIAGRAM

A voice assistant-based home automation system was created using a hybrid architecture, allowing control through Google Assistant and a custom mobile app. The ESP32 microcontroller processed voice and app commands, triggering a relay module for appliance control. Firebase was used for real-time cloud communication. The hardware setup involved careful wiring, while the software involved Arduino IDE programming and MIT App Inventor. Voice commands were integrated using IFTTT.

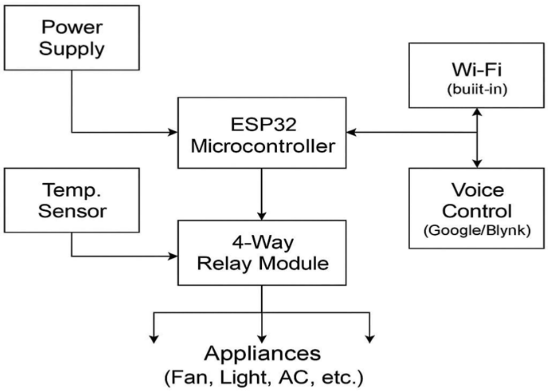

Figure 1 Block Diagram of voice assistant home automation.

DOI: 10.17148/IJIREEICE.2025.13810

B. FLOW CHART

The flowchart shows a voice-controlled system for turning on a light using an ESP32 microcontroller and Firebase. The system starts in standby mode, waiting for the "Turn on light" command. If recognized, it's sent to Firebase, which communicates the instruction to the ESP32, which activates the light.

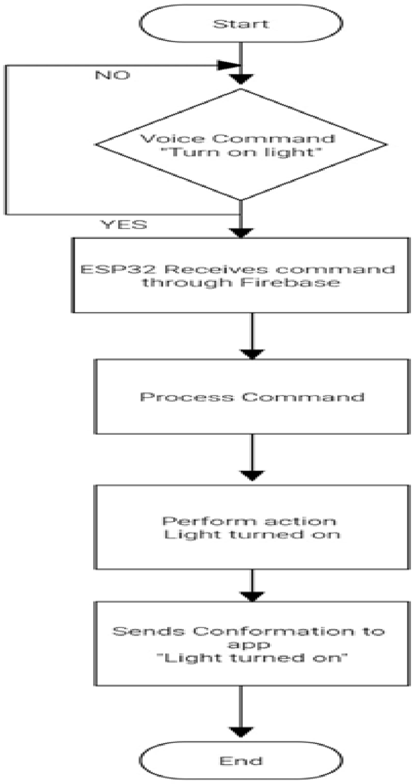

Figure 2 Flow Diagram of voice assistant home automation

After the light is turned on, the ESP32 sends a confirmation message back to the mobile app through Firebase, informing the user that the action was completed successfully. This setup highlights how voice recognition and IoT technologies can work together to create efficient home automation systems.

IV. HARDWARE DESCRIPTION

4.1 ESP 32 Development Board

The ESP32 is a low-cost, low-power SoC microcontroller by Espressif Systems, widely used in IoT, embedded systems, and smart automation projects due to its robust features and built-in wireless communication capabilities. Its dual-core Tensilica Xtensa LX6 processor is efficient for real-time tasks and supports Wi-Fi and Bluetooth. In this project, it serves as the central processing unit, coordinating inputs, displaying data, and sending real-time updates.

Figure 3 ESP32 Development Board

DOI: 10.17148/IJIREEICE.2025.13810

The device features a dual-core Xtensa LX6 microprocessor, built-in Wi-Fi and Bluetooth v4.2, with up to 34 GPIO pins, 520 KB SRAM and 4 MB Flash memory, operating voltage between 3.0V – 3.3V, and supports various protocols like UART, SPI, I2C, I2S, PWM, ADC, DAC, and CAN.

4.2 Relay Module (4 Channels)

A relay module is a critical intermediary in home automation systems that allows low-power control circuits, such as those from microcontrollers like the ESP32, to reliably turn on and off high-power electrical equipment. This feature is required for automating appliances such as lights, fans, and other household items.

Figure 4 Relay Module (4-Channel)

Relay modules offer electrical isolation, high voltage and current handling, versatility, and multiple channels for controlling various loads. They provide galvanic isolation between the control circuit and high-voltage loads, ensuring user safety. They can handle devices operating at higher voltages and currents than microcontrollers can handle. Relay modules can be configured in single-channel or multi-channel configurations.

4.3 Power Supply(5V/3A Adapter)

The power supply plays a crucial role in the functioning of the ESP32 microcontroller, relay module, and other system components. It provides the necessary voltage and current, with a 5V output and at least 3A current. It ensures a stable output, preventing system crashes or malfunctions. The purpose is to power the ESP32 microcontroller, relay module, and other components, ensuring continuous and reliable operation.

Figure 5 Power supply

4.4 Jumper Wires & Connectors

Jumper wires and connectors are necessary for making connections between components such as the ESP32, relay module, switches, and LED indicators. The ESP32 router features flexible connections and secure connectors for secure connections between hardware components. It allows easy reconfiguration and modification, and can be controlled via a mobile app or voice assistant. The router also provides internet access and enables data transmission from cloud services like Firebase.

Figure 6 Jumping Wires

V. SCHEMATIC DIAGRAM

The diagram shows a smart home automation system using an ESP32 NodeMCU microcontroller, a 4-channel relay module, a DHT11 temperature and humidity sensor, and light bulbs as the load. The ESP32 serves as the central control

ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⋇ Peer-reviewed & Refereed journal ⋇ Vol. 13, Issue 8, August 2025
DOI: 10.17148/IJIREEICE.2025.13810

unit, connected to a power supply module. The DHT11 sensor provides real-time environmental data, automating relay-controlled devices. Each relay is connected to a light bulb, allowing the ESP32 to control each bulb based on sensor data or remote commands via Wi-Fi. This IoT-based home automation system enables smart control of electrical appliances. The relay module receives control signals from specific GPIO pins of the ESP32, and its VCC and GND are powered accordingly. This setup demonstrates a basic yet effective implementation of IoT-based home automation, enabling smart control of electrical appliances through a combination of sensor feedback and microcontroller programming.

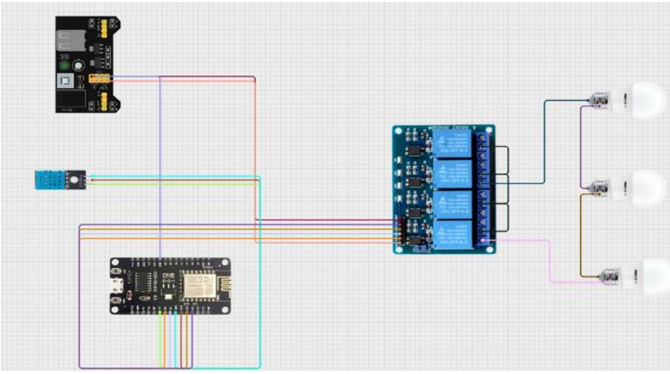

Figure 7 Schematic of Voice Assistant Home Automation

VI. SOFTWARE DESCRIPTION

6.1 ESP32 with Arduino IDE

For ESP32 Programming, we can use Arduino IDE. Open the Arduino IDE and go to File ► Preference as shown in the Figure 8 .Then on the Preference window as shown in Figure 9, enter the below link in Additional Boards Manager URLs https://raw.githubusercontent.com/espressif/arduino-esp32/gh- pages/package_esp32_index.json.Then click on OK button.

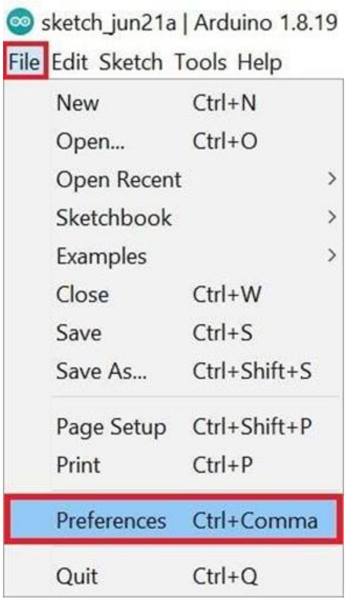

Figure 8 File bar

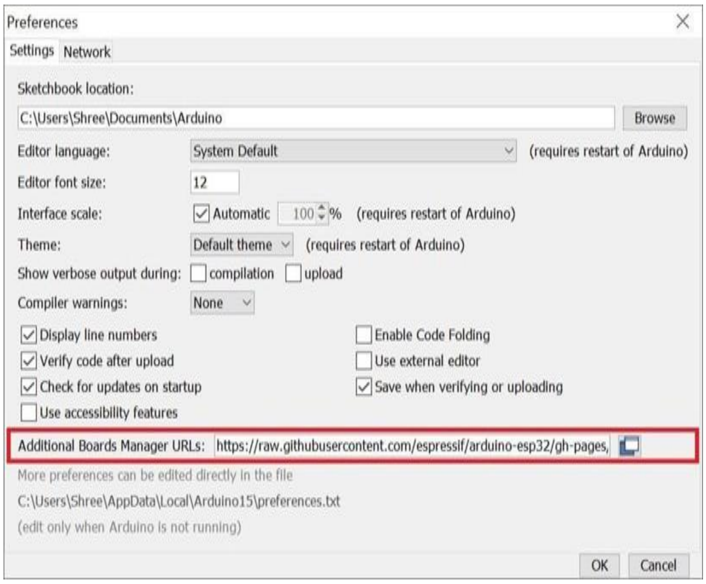

Figure 9 Preference window

To add the ESP32 board, go to the path Tools ► Board ► Boards Manager in Figure 10(a). Then, type on esp32 on the search bar and click on the install button in Figure 10(b). Wait for installation window as shown in Figure 10(c)

ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 13, Issue 8, August 2025
DOI: 10.17148/IJIREEICE.2025.13810

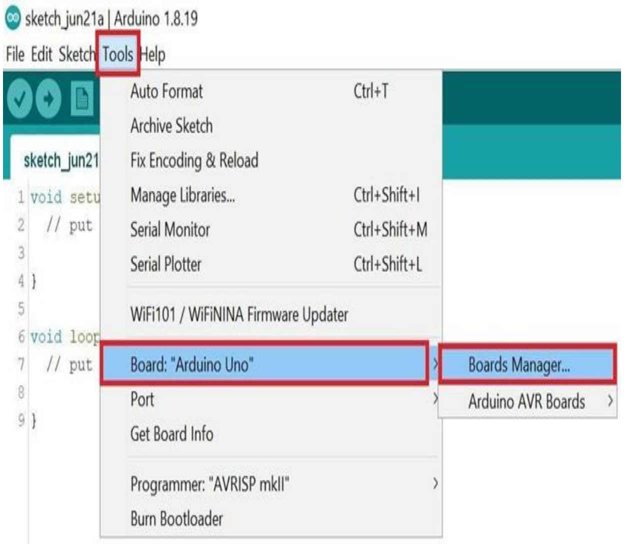

Figure 10(a)

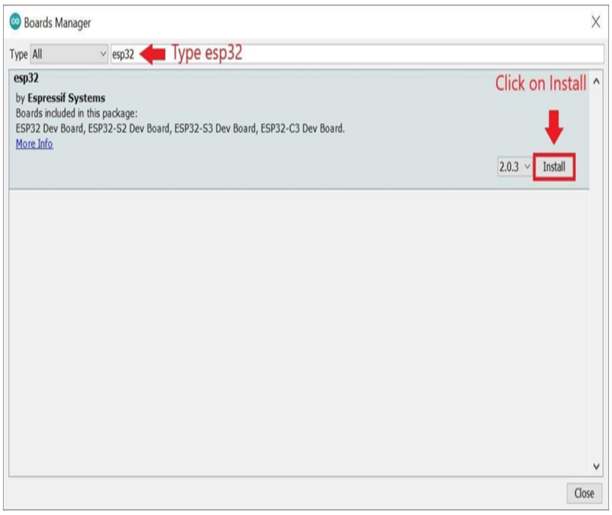

Figure 10(b)

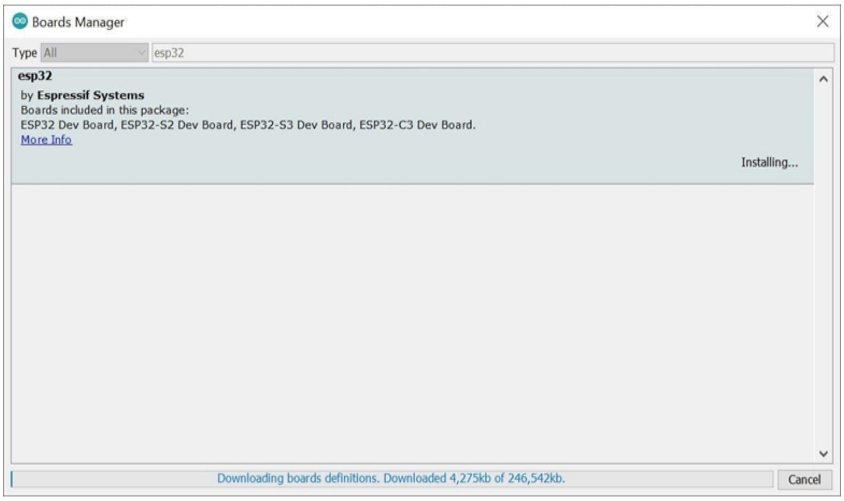

Figure 10(c)

After board installation select the ESP32 board using the path Tools ► Board ► ESP32 Arduino ► DOIT ESP32 DEVKIT V1 as shown in Figure 11(a) and then Select the COM Port using Tools ► Port ► COM* path. In our computer, the ESP32's COM port shows COM5 as shown in Figure 11(b).

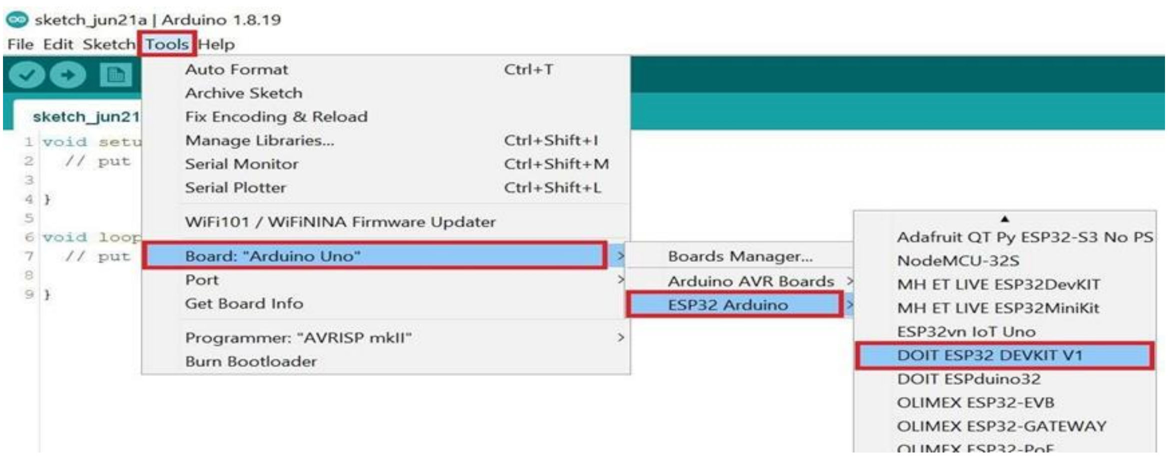

Figure 11 (a)

DOI: 10.17148/IJIREEICE.2025.13810

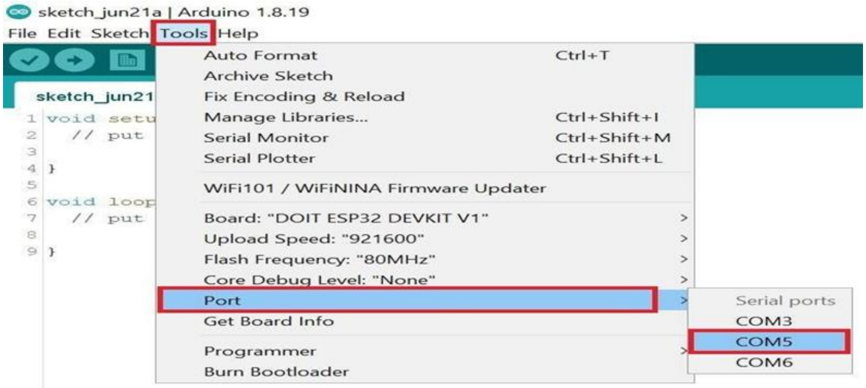

Figure 11(b)

6.2 Arduino IDE for Software Development
In Arduino IDE, navigate the mouse cursor to the Tools bar. Then click on Tool bar menu then select in Arduino then we need to use ESP – 32. Among the list shown in figure select the Which ESP board you going to use. Then go to Preferences. Select the Path and use the link which is given ashttps://dl.espressif.com/dl/package_esp32_index.json Double click the Boards Manager. Type ESP32 select 2nd one over there and install them.

6.3 Web Dashboard
The Smart Home Web Dashboard is a web-based interface for users to interact with their smart home system, developed using HTML, CSS, and JavaScript, and connects to IoT microcontrollers for real-time control and feedback.

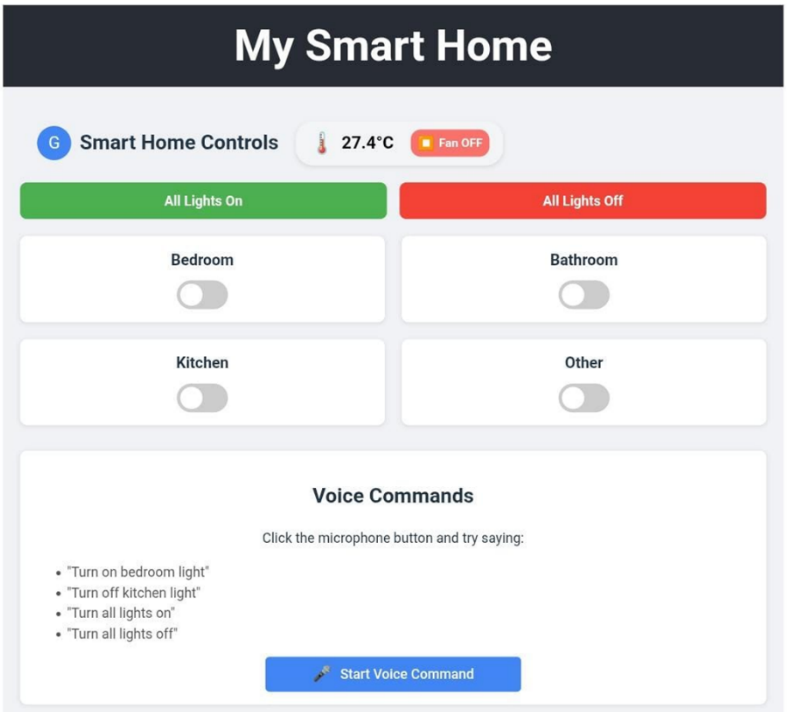

Figure 12 Web Dashboard to control lights and fans and update real time temperature

The Smart Home Web Dashboard is a user-friendly interface designed using HTML, CSS, and JavaScript. HTML defines the structure, including headers, buttons, and voice commands. CSS enhances visual appeal through color coding and layout design. JavaScript handles interactivity, capturing button click events, sending control commands, retrieving real-time temperature data, and integrating voice recognition. The dashboard dynamically updates the user interface based on device status. The voice control feature uses the Web Speech API in modern browsers like Chrome. The dashboard syncs device states using Firebase and HTTP requests. Key features include temperature display, fan status, room toggles, all-on/off controls, voice commands via the Web Speech API, and real-time updates.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 8, August 2025

DOI: 10.17148/IJIREEICE.2025.13810

VII. RESULTS AND DISCUSSION

The Voice Assistant-Based Home Automation project enhances convenience, control, and energy efficiency in smart living environments. It integrates voice control through platforms like Google Assistant, allowing users to operate appliances using simple commands. The system uses microcontrollers, relay modules, and IoT integration to interface with household devices, improving accessibility for the elderly and disabled. It also contributes to energy conservation by enabling scheduled or remote control of devices.

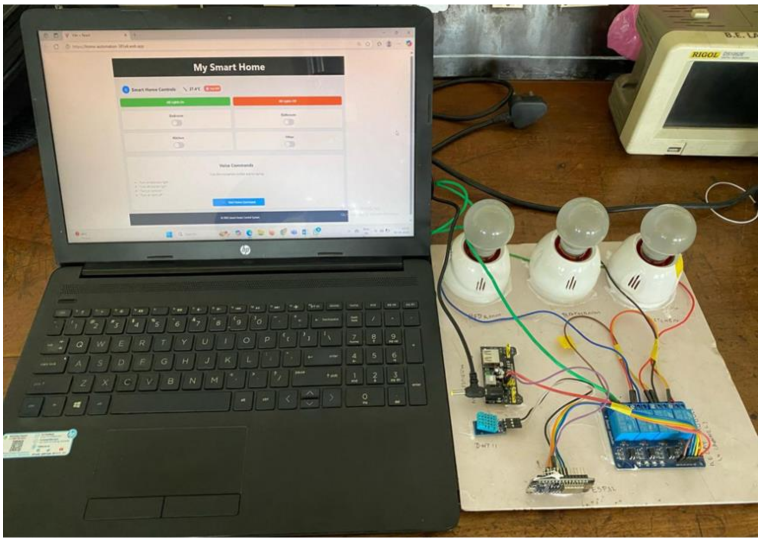

Figure 13 Voice Assistant Home Automation Model

VIII. APPLICATIONS

Some of the applications are:

1. Elderly and Differently-Abled Users: Provides hands-free operation via voice and mobile app control, reducing reliance on physical switches for users with mobility challenges and enhancing independence.
2. Remote Appliance Management: Enables users to monitor and control appliances from anywhere using the mobile app, offering flexibility.
3. Energy-Efficient Living Spaces: Facilitates remote power management by allowing appliances to be turned off from afar, contributing to energy savings and reduced electricity bills.
4. Small Offices or Commercial Spaces: Offers centralized control of lighting and devices, streamlining operations and improving energy management in compact professional environments.
5. Educational & IoT Learning Projects: Serves as an excellent demonstration of hardware (ESP32, relays) and cloud/software (Firebase, IFTTT) integration, making it ideal for IoT learning and development projects.

IX. ADVANTAGES

Main advantages of our project are:

1. Multiple Control Methods: Users can control devices effortlessly using voice commands or a mobile app, offering flexible interaction.
2. Accessibility: Ideal for elderly and disabled users with hands-free voice control, enhancing independence.
3. Cost-Effective: Built with affordable components like ESP32 and open-source platforms, making it budget-friendly.
4. Remote Monitoring: Manage appliances from anywhere using cloud services, providing peace of mind and control.

X. LIMITATIONS

Some of the limitations are:

1. Internet Dependency: Relies on a stable Wi-Fi connection for cloud-based voice and app controls.
2. Limited Device Count: The basic configuration manages only up to 4 devices, requiring expansion for more.
3. Latency: Cloud-based voice commands (e.g., via IFTTT) may experience minor delay

DOI: 10.17148/IJIREEICE.2025.13810

4. Security Risks: Cloud communication requires proper security measures to mitigate potential vulnerabilities.
5. Voice Assistant Restrictions: Control is limited to a set of predefined commands and phrases.

XI. FUTURE SCOPE

The future of ESP32-based smart home automation is promising, with intelligent, autonomous, and interconnected living environments. This includes AI and Machine Learning integration, enabling predictive automation, personalized user experiences, and increased security. Matter protocol adoption, 5G integration, and edge computing will improve connectivity and interoperability. Sensor integration will enable environmental control, and user interaction will improve with natural voice interactions, gesture detection, and Augmented Reality. Future developments will focus on advanced energy management, sustainability, and security, with biometric access control and blockchain exploration. These advancements aim to transform smart homes into intuitive, efficient, and secure spaces.

XII. CONCLUSION

The ESP32-based smart home automation system stands out as a very promising and versatile solution for modern smart living, thanks to its user-centred design. This system offers a versatile and user-friendly foundation for smart home technology. By seamlessly merging voice control with mobile application commands, it dramatically improves accessibility, allowing users to engage with their homes with unrivalled comfort and convenience. One of the system's primary strengths is real-time synchronisation across all control modalities, which ensures constant updates, increasing reliability and user confidence. Furthermore, the system is designed to be both cost-effective and highly scalable, allowing for future extension to handle additional devices and capabilities. Its versatility makes it suited for both personal and home use. Its versatility makes it appropriate for both personal home use and educational purposes. Looking ahead, with further development and potential enhancements such as the integration of advanced sensors for environmental monitoring or improved compatibility with various smart home ecosystems, this system shows significant promise for more sophisticated and comprehensive smart home deployments.

REFERENCES

[1]. V. R. Hegde, A. V. Girish, Smart Home Automation Using Wi-Fi and Voice Control, J. Electrical Eng. and Technology, 2021.
[2]. M. H. Patel, V. K. Rathod, Home Automation Using ESP32 and Voice Recognition, IJECIERD, 2020.
[3]. S. Mohan, M. Kumar, S. S. S. Reddy, IoT-Based Smart Home Automation with Voice Recognition, ICCES, 2020.
[4]. N. S. Sreeja et al., Hybrid Control Home Automation using IoT and Voice Commands, IJAREEIE, 2019.
[5]. J. A. D. Fernandes et al., Voice-Controlled Home Automation System, IEEE LATAM Transactions, 2018.
[6]. S. Joshi, K. Das, S. Shukla, Voice-Activated Home Automation with Cloud and Manual Override, ICCT, 2017.
[7]. Shailendra S. Gokhale, P. S. G. Dhamdhere, Smart Home Automation using IoT, IEEE ICCES, 2016.

.