International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

# DESIGN AND IMPLEMENTATION OF AN OFFLINE VOICE-CONTROLLED HOME ASSISTANT

**Ajayi Oluwabukola F (PhD)[1], Oyiana Jeremiah Chukwuka[2], Izang Aaron (PhD) [3], Ufomaduh Rodrick Iheomasinachi[4], Eweoya Ibukun[5], Anyaehie Amarachi Adaugo[6], Solanke Oluwole[7], Taiwo Omobolaji Samuel[8].**

[1,2,3,6,7,8]Computer Science Department, Babcock University, Ilishan-Remo, Ogun State, Nigeria.
[3]Information Technology Department, Babcock University, Ilishan-Remo, Ogun State, Nigeria.
[4]Computer Science Department, Babcock University, Ilishan-Remo, Ogun State, Nigeria.
[1]ajayioluwa@babcock.edu.ng, [2]Oyiana7942@student.babcock.edu.ng, [3]izanga@babcock.edu.ng, [4]Ufomaduh9112@student.babcock.edu.ng, [5]eweoyai@babcock.edu.ng, [6]amarachi.anyaehie@yahoo.co.uk, [7]Solanke0069@pg.Babcock.edu.ng, [8]Taiwo2583@student.babcock.edu.ng

**ABSTRACT**
In recent years, voice-controlled home automation has significantly transformed how individuals interact with their living spaces. However, mainstream solutions such as Amazon Alexa, Google Assistant, and Apple Siri rely heavily on cloud-based processing, which introduces serious concerns regarding data privacy, latency, and dependence on constant internet connectivity. This project introduces Home Guardian, an offline-capable voice-controlled home assistant designed to operate entirely without internet access. Utilizing a microcontroller-based system architecture combined with local voice recognition through the AI Thinker VC-02 module, Home Guardian allows users to control various smart home devices—such as lights, smart sockets, and door locks—and play offline music via direct voice commands or a mobile application developed using Kodular. The system communicates over a local Wi-Fi network, enabling real-time, secure interaction between the app and the central unit, even in areas with poor or no internet service. Initial testing shows that localized voice processing improves response speed, enhances privacy, and reduces overall system costs. By eliminating the need for cloud servers, this research presents a privacy-focused, cost-efficient, and adaptable alternative to commercial voice assistants, especially suitable for resource-constrained or privacy-sensitive environments.

**KEYWORDS**
Offline Home Assistant, Local Voice Control, ESP32, AI Thinker VC-02, Smart Home IoT, Kodular App, LAN Communication, Privacy-Centric Automation.

## 1. Introduction

Smart home technologies have increasingly become an integral part of modern living, enhancing convenience, energy efficiency, and home security [1]. At the forefront of this evolution are voice-controlled systems such as Amazon Alexa, Google Assistant, and Apple Siri, which allow users to control appliances, retrieve information, and automate daily tasks using natural language commands. However, these systems typically rely on constant internet connectivity and cloud-based processing, raising concerns about user data privacy, latency, cost, and network dependency [2, 3].

In developing regions or rural areas where internet access is limited or unstable, these cloud-based solutions may not function reliably [4]. Moreover, transmitting sensitive audio data to external servers introduces potential vulnerabilities and privacy concerns. As noted by [2], processing voice commands locally can significantly mitigate these risks and improve user trust in smart assistant systems.

This work, titled Design and Implementation of an Offline Voice-Controlled Home Assistant, addresses these limitations by introducing "Home Guardian," a microcontroller-based offline assistant capable of interpreting voice commands without relying on the internet. The system employs the AI Thinker VC-02 module for local voice recognition, integrated with the ESP32 microcontroller, relays, and sensors to control household appliances and monitor environmental parameters such as temperature, humidity, and atmospheric pressure.

In addition to voice interaction, the assistant features a mobile application developed using Kodular, which allows users to send commands over a local Wi-Fi network. This eliminates cloud dependence, reduces latency, and ensures functionality even in internet-deprived environments. The system offers a cost-effective and secure alternative to commercial assistants while promoting digital inclusivity, especially in underserved communities.

## 2. Problem Statement

Voice-controlled home assistants have transformed how users interact with their living spaces, offering hands-free control of lighting, appliances, and entertainment systems. However, widely used systems like Amazon Alexa, Google Assistant, and Apple Siri depend heavily on internet connectivity and cloud-based processing. This reliance introduces several key challenges. First, it raises significant privacy concerns, as user commands and personal data are transmitted to and stored on external servers. Second, it creates performance issues in areas with poor or unstable internet access, where these assistants become slow, unresponsive, or entirely non-functional. Third, the high cost of commercial voice assistants and their required hardware places them beyond the reach of many users, especially in low-income or resource-constrained environments.

These limitations highlight the need for an offline, cost-effective, and reliable voice-controlled home assistant that can operate efficiently without internet access. Such a system would ensure greater data privacy, better performance in low-connectivity areas, and broader accessibility for users regardless of their economic or infrastructural limitations.

## 3. Literature Review

### 3.1. Cost-Effective Home Automation System with Offline Mode Controller and Speaker Verification (Voice Commands) for Handicapped [5].

This study presents a home automation system tailored for individuals with disabilities, integrating offline voice command processing and speaker verification to enhance accessibility. The system employs the ESP8266 microcontroller and supports local control without internet dependency. While the system emphasizes accessibility, it lacks comprehensive integration with environmental sensors (e.g., temperature, humidity) and does not address broader smart home functionalities beyond basic device control. Additionally, the scalability and adaptability of the system to diverse home environments are not thoroughly explored.

**3.2. Design of an Intelligent Voice Controlled Home Automation System [6].**

This paper introduces a microcontroller-based system that enables users to control home appliances via voice commands using an Android smartphone. It leverages Bluetooth communication for device interaction, aiming to provide a cost-effective solution for home automation. The reliance on Bluetooth limits the operational range and may not be suitable for larger homes. The system's dependence on an Android application also restricts its accessibility to users of other platforms. Furthermore, the study does not address offline voice processing capabilities, which are crucial for ensuring functionality in areas with limited internet connectivity.

**3.3. Intuitive Control on Electric Devices by Smartphones for Smart Home Environments [7].**

This research proposes a smartphone-based control mechanism for smart home devices, utilizing intuitive gestures and interfaces to enhance user experience. The system focuses on improving the ease of interaction between users and home appliances through mobile applications. The study primarily addresses the user interface aspect and does not delve into the integration of offline voice control or environmental sensing. Additionally, the system's dependence on smartphones and potential lack of support for users with disabilities may limit its inclusivity.

**3.4. Design and Implementation of a Cloud-Enabled Random Neural Network-Based Decentralized Smart Controller with Intelligent Sensor Nodes for HVAC [8].**

This paper presents a decentralized smart controller for HVAC systems, integrating intelligent sensor nodes and leveraging cloud-based random neural networks for efficient energy management. The system aims to optimize heating, ventilation, and air conditioning operations through advanced data processing. The reliance on cloud computing introduces concerns regarding data privacy and system latency. In scenarios with limited or unreliable internet connectivity, the system's performance may degrade. Additionally, the focus on HVAC systems limits the applicability of the findings to broader home automation contexts.

**3.5. Discovery of Resident Behavior Patterns Using Machine Learning Techniques and IoT Paradigm [9].**

This study explores the use of machine learning algorithms, specifically the C4.5 algorithm and Weka API, to identify resident behavior patterns in smart homes. The goal is to enable customized automatic control schemes for home devices based on user habits.
The system's effectiveness is heavily dependent on the availability and quality of data collected from IoT devices. In environments where data collection is limited or inconsistent, the accuracy of behavior pattern recognition may suffer. Moreover, the study does not address the implementation of offline processing, which is vital for ensuring privacy and functionality in areas with limited internet access.

## 4. Methodology

### 4.1. Design Model

This project adopts the Incremental Software Development Life Cycle (SDLC) model to ensure modular development and systematic integration of hardware and software components. The incremental approach allowed for building and testing each module—such as voice recognition, environmental sensing, device control, and mobile app interface—individually before full system integration. This method enabled early detection of errors and continuous feedback incorporation. The development began with core functionalities like offline voice recognition and smart socket control, followed by the integration of environmental sensors and mobile app communication. Final testing included system-wide validation to ensure command accuracy, responsiveness, and offline operability.

**4.2. Development Tools**

i. **Hardware Components:**

- AI Thinker VC-02 Module**:** Used for offline voice command recognition.
- ESP32 Microcontroller: Acts as the system's main controller, executing commands and managing communication.
- Sensors**:** Includes BME280 for temperature, humidity, and pressure sensing.
- Relay Modules**:** Used to switch connected appliances ON/OFF based on commands.
- Smart Socket Unit: Connected via the local network and controlled remotely by the ESP32.

ii **Software Tools:**

- Kodular: Used to build a mobile app for local control and monitoring, without requiring any programming.
- Arduino IDE: Used to write and upload code to the ESP32 and configure the voice module and sensors.
- Wi-Fi (Local Network): Communication between the mobile app and ESP32 takes place over the same Wi-Fi network.
- Kodular Companion: For real-time mobile app testing.
- Fritzing: Used to create circuit diagrams for hardware implementation.

**4.3 Functional Modules**

i. Offline Voice Recognition Module: The AI Thinker VC-02 recognizes predefined voice commands locally, without any need for cloud access, and sends command signals to the ESP32.

ii. Command Execution and Device Control Module: The ESP32 interprets the commands received and activates the appropriate relays to control devices such as lights, sockets, and door locks.

iii. Environmental Monitoring Module: Sensors connected to the ESP32 measure real-time temperature, humidity, and pressure values. These readings can be accessed through the mobile app.

iv. Mobile Application Module: The app created with Kodular allows users to control devices manually, view sensor readings, and send commands over the local Wi-Fi. It ensures that users can interact with the system even without speaking.

v. Smart Socket Control Module: A secondary ESP-based socket receives ON/OFF commands from the main system wirelessly. The local communication ensures real-time switching without relying on the internet.

vi. Power Supply and Safety Module: The system uses regulated power circuits to convert AC to required DC levels (e.g., 3.3V and 5V). Protective components like resistors and diodes are used to ensure stability and hardware safety.

**4.4 System architectural design**

The architectural design is a simple yet effective structure that illustrates how the offline voice-controlled home assistant operates. It includes a user giving voice commands directly to the hardware component, which processes the input using offline voice recognition and then sends the appropriate control signals to connected appliances. This direct interaction between the user, hardware, and appliances ensures fast response times and independence from internet connectivity, making the system highly reliable for offline use. The hardware also contains sensors to measure environmental conditions like temperature, pressure, and humidity, adding a layer of intelligent monitoring to the system's core functions (Figure 1).

The design also incorporates a mobile application that communicates with the hardware over a local Wi-Fi network. Users can interact with the system via this app either through voice or touch commands, offering an alternative control method. The mobile app both sends commands and receives updates from the hardware, ensuring users can manage and monitor appliances remotely within the network range. This two-way communication between the hardware and the app enhances usability and flexibility, while maintaining the system's offline functionality and privacy.

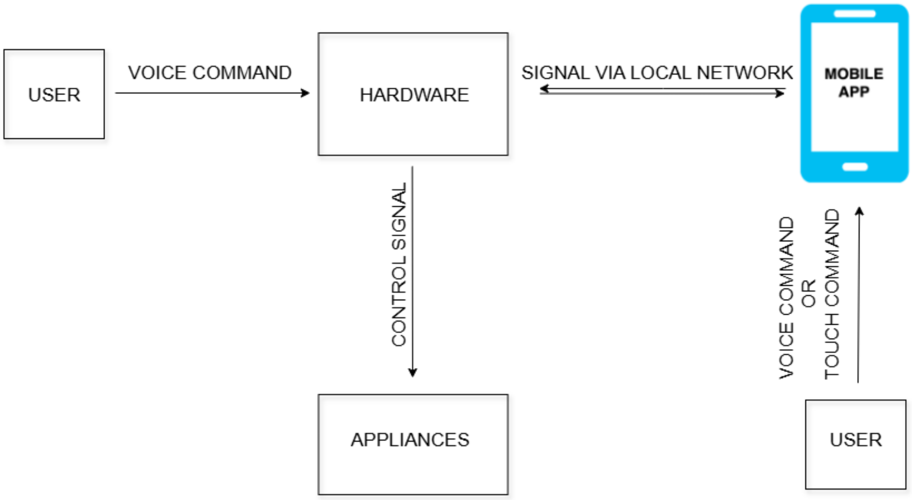


Figure 1: System Architecture Diagram

### 4.5. Flowcharts, Circuit Diagram and Block Diagram

#### 4.5.1. Flowcharts

A flowchart is a visual representation of the sequence of steps and decisions needed to perform a process. Each step in the sequence is noted within a diagram shape. Steps are linked by connecting lines and directional arrows. It is simply a graphical representation of steps. Figures 2,3,4 shows the flowcharts of main system, Mobile App and Smart Socket respectively.

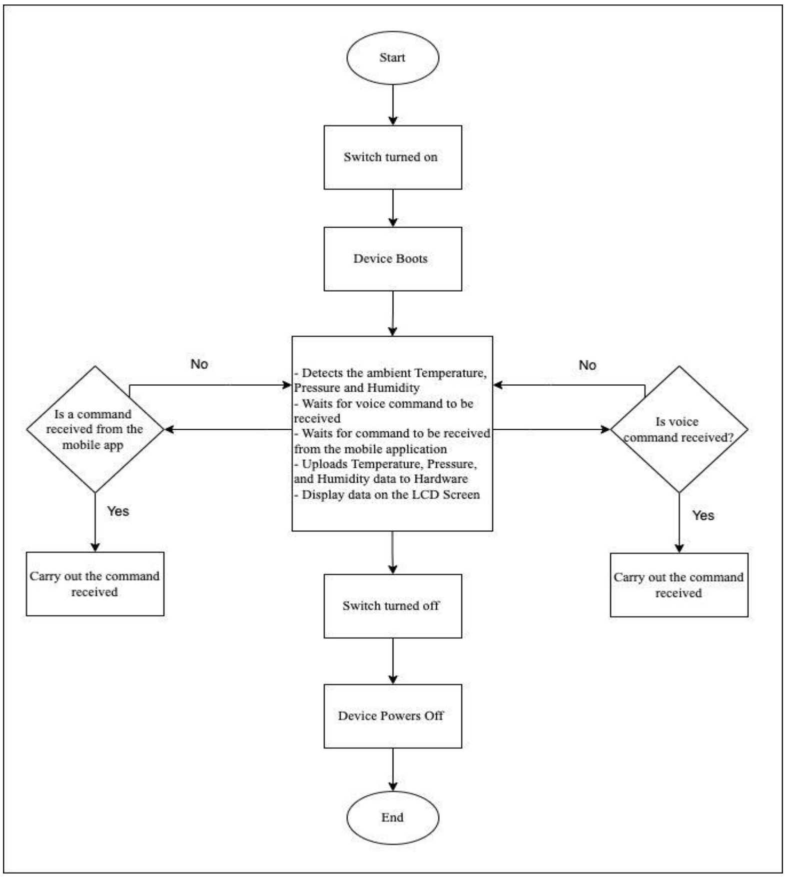


Figure 2: Flowchart Diagram for the main system

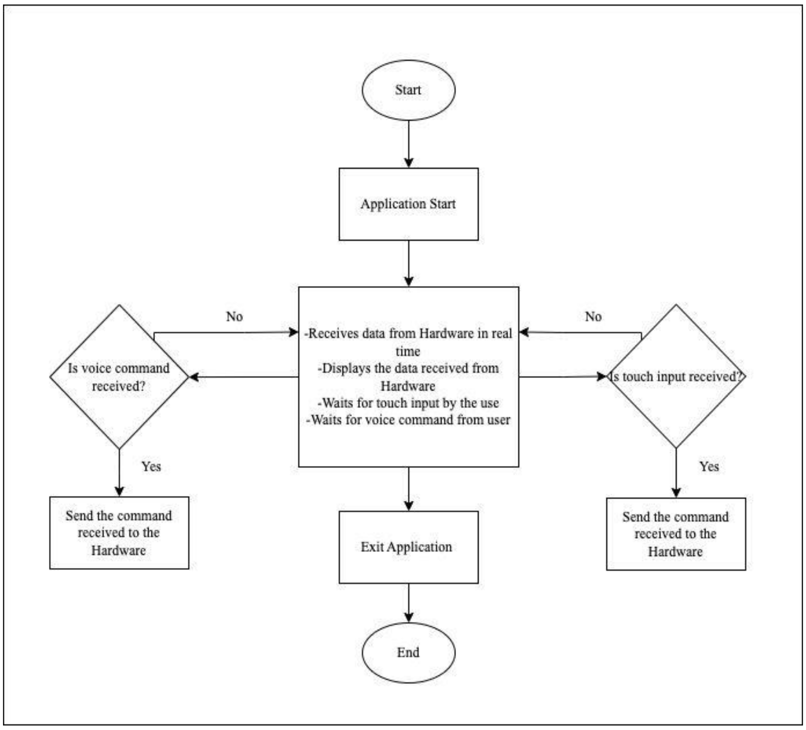


Figure 3: Flowchart Diagram for the Mobile App

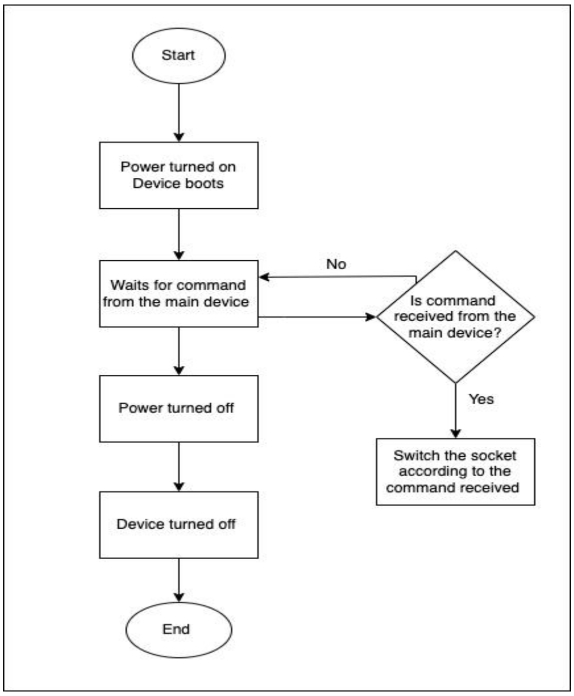


Figure 4: Flowchart Diagram for the Smart Socket

### 4.5.2. Circuit Diagram

The circuit diagram of the voice-controlled home assistant is a vital schematic that illustrates the electrical connections and components (Figure 5) that enable seamless home automation and device control. It shows how the ESP32 microcontroller, sensors, relay modules, and LCD screen work together to interpret voice commands and provide real-time feedback on environmental conditions. This diagram highlights the flow of signals from the user's voice command to the microcontroller, which processes the command and controls various devices through relays.

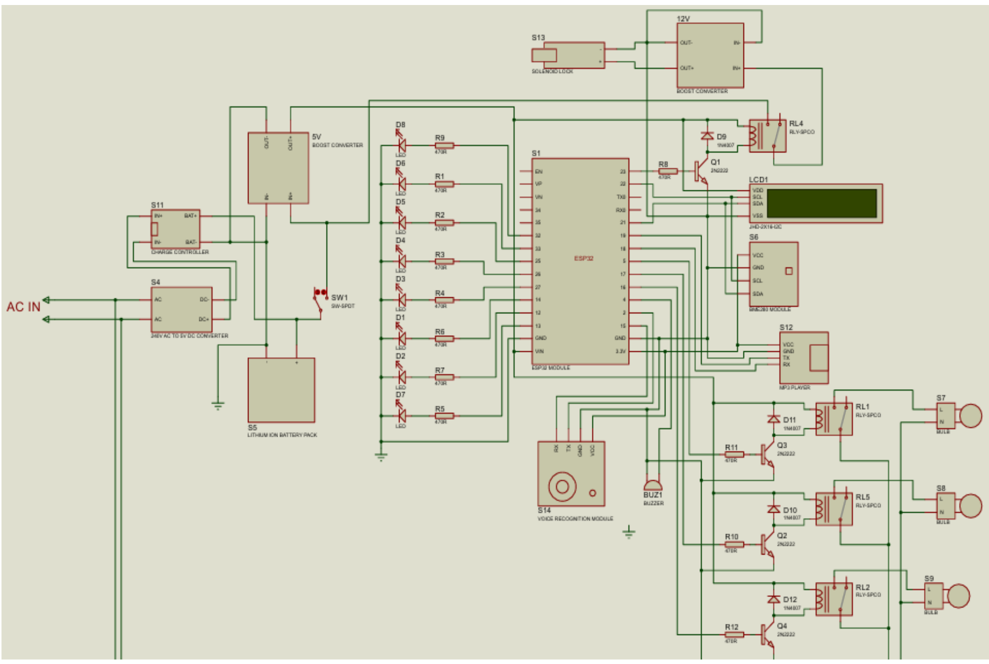


Figure 5: Circuit Diagram for the Main System

### 4.5.3. Block Diagram

A block diagram is a simplified visual representation of a system that shows its main components or functions and how they interact with each other. Rather than showing detailed circuits or code, it uses labeled blocks connected by arrows or lines to represent the flow of information, power, or control between different parts of the system.

The block diagram (figure 6) illustrates how major components like the voice recognition module, ESP32 microcontroller, sensors, relay modules, and mobile app all connect and communicate. Each block represents a functional unit of the system, such as how voice commands are processed, how sensor data is read, and how outputs like lights and sockets are controlled. This diagram helps give a clear overview of the system's operation and highlights the offline, local-network-based interaction between the hardware and software components.

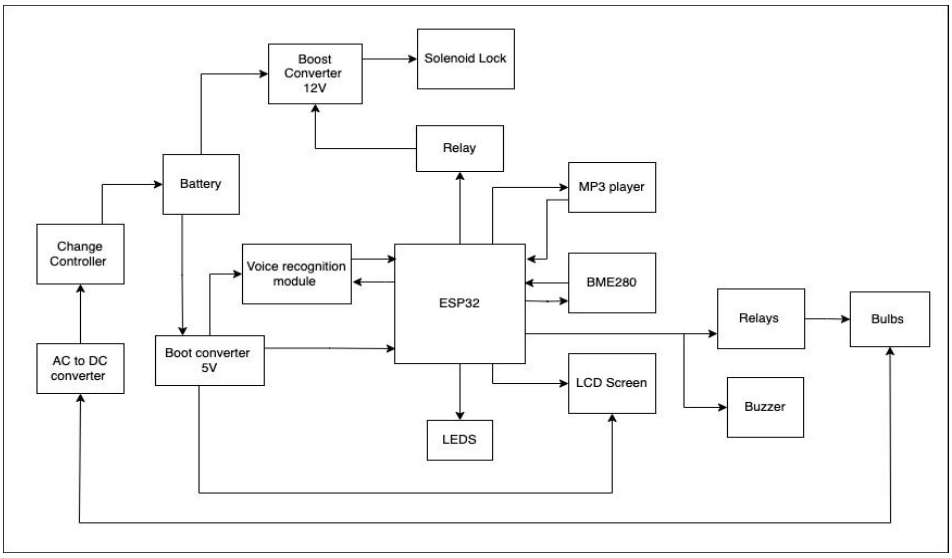


Figure 6: Operational Block Diagram for the Main System

## 5. Result

The implementation of the offline voice-controlled home assistant, Home Guardian, resulted in a functional and responsive system capable of executing various household control tasks without internet dependence. The assistant successfully interpreted and responded to voice commands to perform actions such as switching lights on/off, controlling smart sockets, unlocking/locking doors, and playing locally stored music. In addition to voice control, the companion mobile application built with Kodular provided an intuitive interface for users to issue the same commands over a local Wi-Fi network, ensuring accessibility even when voice control is not practical.

Testing demonstrated that the system maintained fast response times, typically under one second per command, and operated reliably in a standalone environment. The use of onboard sensors for temperature, humidity, and pressure provided accurate real-time environmental data, which could be monitored via the main control unit. Importantly, all operations remained secure and private, as data processing occurred entirely on the device without cloud interaction. These results validate the feasibility and effectiveness of local, offline smart home systems for users concerned with privacy, cost, and network limitations.

### 5.1 The Main System

The architectural design centers on a fully offline voice-controlled assistant that connects a user, hardware control unit, appliances, and a mobile app over a local network. The user can issue voice commands directly to the hardware (through its onboard microphone and voice-recognition module) or via the mobile app's "Voice" interface. Similarly, touch commands on the app's "Switch" panel allow manual control. All app–hardware communication traverses your private Wi-Fi network, so no external internet or cloud services are ever involved.

At the heart of the hardware box sits the ESP32 microcontroller, which orchestrates every action. When a voice sample arrives—either captured by the built-in VC-02 module or relayed from the app via Wi-Fi—the ESP32's firmware decodes the command (e.g. "Turn on light 2" or "Lock door"). It then drives the corresponding GPIO pin high or low, signaling a transistor driver that energizes the appropriate 12 V relay. Each relay, in turn, switches AC power to one of your smart sockets, lights, or the door-lock solenoid.

Feedback loops ensure the user always knows the system's state: the ESP32 updates the LCD screen inside the hardware box with current temperature, humidity, or device status, and simultaneously pushes sensor readings and on/off states back to the app. This two-way sync lets the app's toggles reflect real-time states—even if commands come directly via microphone to the hardware—so the "Switch" screen never shows stale information.

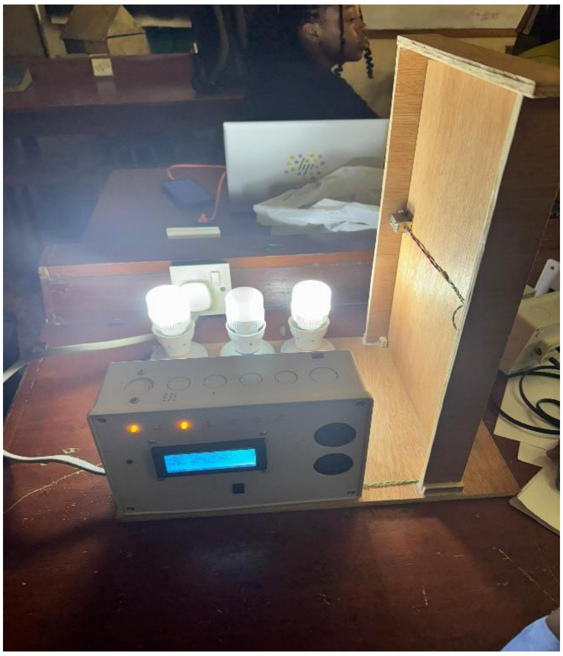

Figure 7: Front View of the System Turned On

### 5.2 The Mobile App

The Home Guardian mobile app, built with Kodular, provides a straightforward interface for controlling your offline voice assistant. On launch, the app presents two main screens: Voice and Switch. In the Voice screen (Figure 8), users tap a microphone icon to record a spoken command. The app then sends that audio over the local Wi-Fi network to the ESP32 controller, which processes the command—such as "turn on socket 1"—and activates the appropriate relay.

The Switch screen (figure 9) offers manual control via on-screen toggles for each device (e.g., smart sockets, lights, door lock). Tapping a toggle sends a simple on/off signal to the ESP32, ensuring instant response without requiring voice input. Behind the scenes, the app maintains a persistent connection to the ESP32 over your home's WLAN, so commands—whether voice or touch—are carried out in real time without any internet dependency.

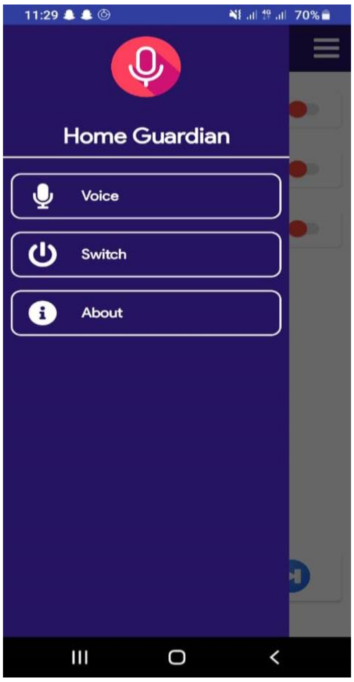


Figure 8: Voice Control Interface

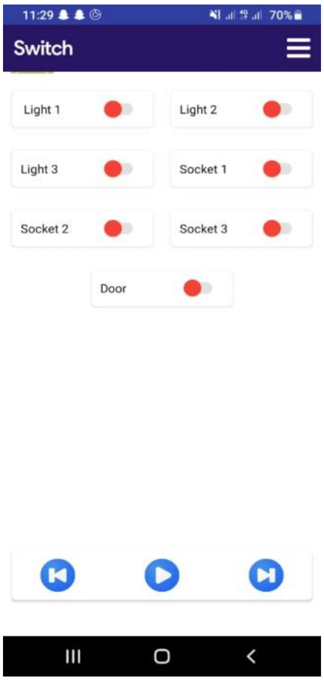


Figure 9: Switch Control Interface

## 6. Conclusion

This study successfully designed and implemented an offline voice-controlled home assistant aimed at improving local smart home automation without reliance on internet connectivity. The system responds to voice commands to perform essential tasks such as lighting control, appliance management, environmental sensing (temperature, humidity, and pressure), and local audio playback. Built using a modular development approach, the assistant integrates key components including voice command recognition, sensor data acquisition, and real-time device control through a secure local Wi-Fi network. The companion mobile application further enhances usability, allowing users to issue commands remotely within the local network. Overall, the project demonstrates the feasibility of cost-effective, private, and reliable smart home solutions tailored to offline environments. The success of the system lays a foundation for future expansion, including support for additional devices, improved command handling, and enhanced user personalization.

## References

1. Alam, M. R., Reaz, M. B. I., & Ali, M. A. M. (2012). A review of smart homes—Past, present, and future. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(6), 1190–1203. https://doi.org/10.1109/TSMCC.2012.2189204
2. Chollet, G., Sansen, H., Tevissen, Y., Boudy, J., Hariz, M., Lohr, C., & Yassa, F. (2024). Privacy preserving personal assistant with on-device diarization and spoken dialogue system for home and beyond. arXiv preprint, arXiv:2401.01146. https://doi.org/10.48550/arXiv.2401.01146

3. Bermuth, D., Poeppel, A., & Reif, W. (2022). Jaco: An offline running privacy-aware voice assistant. arXiv preprint, arXiv:2209.07775. https://doi.org/10.48550/arXiv.2209.07775
4. Falaki, H., Mahajan, R., Kandula, S., Lymberopoulos, D., Govindan, R., & Estrin, D. (2013). Diversity in smartphone usage. Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services, 179–194. https://doi.org/10.1145/1814433.1814453
5. Ali, A. (2023). Cost-effective home automation system with offline mode controller and speaker verification (voice commands) for handicapped. International Journal of Computer Science and Network Security, 23(6), 150–165.
6. Sen, S., Chakrabarty, S., Toshniwal, R., & Bhaumik, A. (2015). Design of an intelligent voice controlled home automation system. International Journal of Computer Applications, 121(15), 39–42.
7. Pan, M.-S., & Chen, C.-J. (2016). Intuitive control on electric devices by smartphones for smart home environments. IEEE Sensors Journal, 16(11), 4281–4294. https://doi.org/10.1109/JSEN.2016.2531162
8. Javed, A., Larijani, H., Ahmadinia, A., Emmanuel, R., Mannion, M., & Gibson, D. (2017). Design and implementation of a cloud-enabled random neural network–based decentralized smart controller with intelligent sensor nodes for HVAC. IEEE Internet of Things Journal, 4(2), 393–403. https://doi.org/10.1109/JIOT.2017.2669482
9. Reyes-Campos, J. J., Olmedo-Aguirre, O. G., Machorro-Cano, J. L. I., Sánchez-Cervantes, R., & Rodríguez-Mazahua, L. (2021). Discovery of resident behavior patterns using machine learning techniques and IoT paradigm. Mathematics, 9(3), 219. https://doi.org/10.3390/math9030219