IJMRT: Volume (8), Issue 1, 2026, Pages 59-72
International Journal of Multidisciplinary Research Transactions
(A Peer Reviewed Journal)
www.ijmrt.in

I SSN (Print):2663-2381
ISSN(Online):2663-4007
| Regular Article | Open Acess
Aravandi Publications

Voice-Controlled Smart Home Automation Using Raspberry Pi And IoT

Dharani Sivapriya S[1], Bhuvaneshwari M[2]*, Seetharaman R[3]

[1]Student, Department of Instrumentation and Control Engineering, Saranathan College of Engineering, Tamil Nadu, India.
[2]Student, Department of Instrumentation and Control Engineering, Saranathan College of Engineering, Tamil Nadu, India.
[3]Assistant Professor, Department of Instrumentation and Control Engineering, Saranathan College of Engineering, Tamil Nadu, India.

Corresponding author(s):
Bhuvaneshwari M, Student, Department of Instrumentation and Control Engineering, Saranathan College of Engineering, Tamil Nadu, India.
Email: bhuvanamurali12345@gmail.com

DoI: https://doi.org/10.5281/zenodo.18427902

Citation:

Dharani Sivapriya S, Bhuvaneshwari M, Seetharaman R (2026). Voice-Controlled Smart Home Automation Using Raspberry Pi And IoT. International Journal of Multidisciplinary Research Transactions, 8(1), 59–72. https://doi.org/10.5281/zenodo.18427902

Accepted: 27 January 2026 Available online: 30 January 2026

Abstract

This project presents the design and implementation of a smart home automation system using a Raspberry Pi 3 Model B+ to enhance comfort, safety, and energy efficiency. The system enables both manual and automatic control of home appliances, particularly lighting, through physical switches and a custom-developed mobile application. Communication between the Raspberry Pi and the mobile device is achieved over a local Wi-Fi network using the Raspberry Pi's IP address. Automatic operation is supported using an LDR sensor to control lighting based on ambient light conditions, while voice commands provide hands-free control through the mobile application. Additionally, an MQ-2 gas sensor is integrated to detect gas leakage and trigger a buzzer alert to improve household safety. The entire system is programmed in Python and executed using the Thonny IDE. Experimental results demonstrate reliable real-time control, responsiveness, and effective gas leak detection, making the system a cost-effective and scalable solution for smart home applications.

Keywords: Smart Home, Raspberry Pi, IoT, Voice Control, LDR Sensor, MQ-2 Gas Sensor, Home Automation.

1. Introduction

Smart home automation has emerged as an important application of the Internet of Things (IoT), aiming to improve comfort, energy efficiency, and safety in residential environments. With the rapid advancement of embedded systems and wireless communication technologies, traditional home appliances can now be monitored and controlled remotely through smart devices. This project presents the design and implementation of a Raspberry Pi–based smart home automation system that integrates manual control, mobile application control, voice commands, and automatic operation modes.

The proposed system is developed using a Raspberry Pi 3 Model B+ as the central controller. It enables users to control household lighting through physical switches as well as a custom mobile application connected over a local Wi-Fi network. The system supports multiple modes of operation, including manual ON/OFF control, voice-based control via the mobile application, and an automatic mode where a Light Dependent Resistor (LDR) sensor controls lighting based on ambient light conditions. Additionally, home safety is enhanced using an MQ-2 gas sensor to detect gas leakage, which triggers an alert through a buzzer when hazardous conditions are detected.

The Raspberry Pi is programmed using Python in the Thonny IDE, and real-time communication between the mobile application and the Raspberry Pi is achieved using the device's IP address within the same network. This integrated approach provides a low-cost, scalable, and user-friendly smart home solution that enhances convenience, automation, and safety. The project demonstrates the effective use of IoT technologies in developing intelligent home automation systems suitable for modern smart living environments.

2. Literature Review

1. R. Al-Ali, M. Qasaimeh, M. Al-Mardini, S. Radder, and I. A. Zualkernan, "ZigBee-based irrigation system for smart agriculture," IEEE International Conference on Automation Science and Engineering, 2012, pp. 512–517.

Al-Ali et al. (2012) proposed a ZigBee-based smart irrigation system aimed at improving water efficiency in agricultural fields. The system integrates soil moisture sensors with a ZigBee wireless network to enable real-time monitoring and automated irrigation control. The study demonstrates reduced water consumption and reliable low-power communication, making it suitable for smart agriculture applications.

The idea of using wireless communication (inspired by ZigBee) for remote monitoring and control of field devices. Integration of sensors to collect real-time environmental data for automated decision making. Use of a central controller to process sensor data and control outputs. Emphasis on automation to reduce manual intervention, which aligns with your automatic mode operation. Concept of energy-efficient system design suitable for continuous operation.

2. R. Piyare and M. Tazil, "Bluetooth based home automation system using cell phone," IEEE 15th International Symposium on Consumer Electronics, 2011, pp. 192–195.
Piyare and Tazil (2011) developed a Bluetooth-based home automation system that allows users to control household appliances using a mobile phone. The system employs short-range Bluetooth communication for device control, offering a low-cost and simple automation solution. However, the limited range of Bluetooth restricts scalability and remote access compared to newer IoT-based systems.

The idea of wireless control of home appliances using a mobile phone was adopted as the core concept. Similar to the paper, my project focuses on remote ON/OFF control of devices without physical interaction. The use of a mobile application as a user interface to control appliances was inspired by this work. The concept of low-cost and user-friendly home automation was also applied. Additionally, the paper's emphasis on real-time control and convenience influenced the design of my smart home system. These ideas were implemented using modern hardware and software platforms in my project.

3. R. Antonucci, A. Conti, and D. Masotti, "Raspberry Pi-based smart home," International Conference on Consumer Electronics (ICCE), IEEE, 2016, pp. 390–391.
Antonucci et al. (2016) presented a Raspberry Pi–based smart home system that serves as a low-cost and flexible home automation platform. The system integrates sensors and actuators with network connectivity to enable monitoring and remote control of home appliances. The study

highlights the suitability of Raspberry Pi for prototyping scalable and customizable smart home applications.

The project adopts the use of Raspberry Pi as the central control unit for home automation, similar to the architecture proposed in the paper. It follows the idea of controlling home appliances through software-based commands instead of manual operation. The concept of remote control using a mobile application is applied for switching devices ON and OFF. Sensor-based automation, as discussed in the paper, is implemented using an LDR sensor for automatic light control. The system supports manual and automatic modes, inspired by the flexible control approach in the reference work. Overall, the paper influenced the low-cost, scalable, and user-friendly smart home design used in this project.

4. K. Gill, S. H. Yang, F. Yao, and X. Lu, "A ZigBee-based home automation system," IEEE Transactions on Consumer Electronics, vol. 55, no. 2, pp. 422–430, May 2009.
Gill et al. (2009) proposed a ZigBee-based home automation system focused on low-power and reliable wireless communication between home devices. The system uses a centralized controller to manage sensors and actuators, enabling efficient monitoring and control of household appliances. The study demonstrates ZigBee's suitability for scalable and energy-efficient smart home environments.

The idea of wireless communication for controlling home appliances, which inspired remote ON/OFF control in your system. Use of a central controller to manage multiple devices, similar to how Raspberry Pi acts as the main control unit. Concept of user-friendly control through a mobile interface, which relates to your mobile app control. Automation logic where sensors trigger actions automatically, similar to your LDR-based auto mode. Focus on energy-efficient and low-power operation, relevant to smart home design. Overall IoT-based smart home architecture, which forms the foundation of your home automation project.

5. M. Collotta and G. Pau, "A novel energy management approach for smart homes using Bluetooth Low Energy," IEEE Journal on Selected Areas in Communications, vol. 33, no. 12, pp. 2988–2996, Dec. 2015.
Collotta and Pau (2015) introduced a novel energy management approach for smart homes using Bluetooth Low Energy (BLE) technology. The system enables efficient monitoring and control of household energy consumption with minimal power overhead. Results show that BLE-based

communication improves energy efficiency while maintaining reliable connectivity for smart home applications.

The idea of smart energy management by controlling home appliances only when required was adopted. Similar to the paper's approach, my project focuses on efficient device control to reduce power consumption. The concept of wireless communication for home automation inspired the use of remote control via a mobile application. Automation logic such as automatic ON/OFF based on sensor input aligns with their intelligent decision-making method. Centralized control of devices reflects the paper's smart home management framework. Overall, the paper guided the design of an energy-efficient, user-friendly smart home system using modern communication technologies.

3. Problem Statement

Traditional home control systems lack flexibility, automation, and remote accessibility, making it difficult to efficiently manage household devices and energy consumption. Existing methods often rely on manual operation, leading to higher energy usage and reduced comfort. There is a need for a reliable, scalable, and automated solution that can intelligently control and monitor home appliances. Implementing a PLC-based home automation system addresses these issues by enabling centralized, programmable, and remote management of household operations.

4. System Design and Methodology

The proposed smart home automation system is designed using a centralized control approach based on a Raspberry Pi 3 Model B+, which acts as the main processing and communication unit. The system integrates multiple sensors, actuators, and a mobile application to enable real-time monitoring, automation, and remote control of household appliances.

The Raspberry Pi is connected to the local Wi-Fi network and communicates with a custom-developed mobile application using its assigned IP address. All devices, including the mobile phone and Raspberry Pi, operate within the same network to ensure secure and low-latency communication.

5. Hardware Design

5.1. Raspberry PI 3 Model B+

The Raspberry Pi 3 Model B+ serves as the core controller of the system. It processes sensor data, executes automation logic, and controls output devices through GPIO pins. Python programming is used to implement the control logic, and the code is executed using the Thonny IDE.

5.2. Relay Module

A relay module is used to safely interface high-voltage household appliances with the low-voltage GPIO pins of the Raspberry Pi. The relay acts as an electrically isolated switch, enabling ON/OFF control of devices such as lights and buzzers. This ensures user safety and protects the Raspberry Pi from high current and voltage fluctuations.

5.3. LDR Sensor

A Light Dependent Resistor (LDR) is used for automatic lighting control. The resistance of the LDR varies according to ambient light intensity. When the surrounding light level falls below a predefined threshold, the Raspberry Pi automatically activates the relay to turn ON the light. When sufficient light is detected, the light is turned OFF to conserve energy.

5.4. MQ-2 Gas Sensor

The MQ-2 gas sensor is employed to detect the presence of combustible gases such as LPG and smoke. When gas concentration exceeds the safe limit, the Raspberry Pi triggers a buzzer through the relay module to provide an immediate alert, enhancing household safety.

5.5. Battery Backup System

A battery backup unit is incorporated to ensure uninterrupted operation during power failures. The battery supplies power to the Raspberry Pi, sensors, and relay module, allowing continuous monitoring and alert generation even when the main power supply is unavailable. This improves system reliability and safety, especially for gas leakage detection.

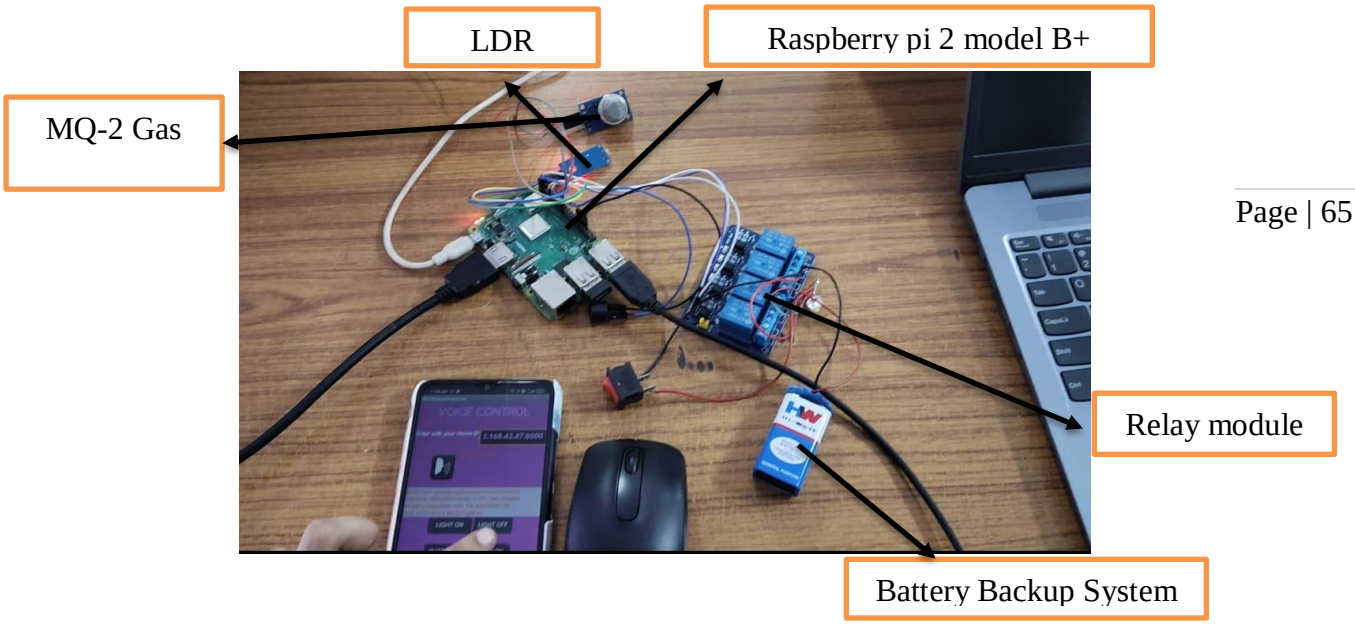


Figure.1. Hardware Design

6. Software Implementation

The software architecture of the proposed smart home automation system is developed using the Python programming language and runs on the Raspberry Pi 3 Model B+. The Raspberry Pi Operating System provides a stable environment for executing the control logic, while the Thonny IDE is used for coding, debugging, and execution. The software is responsible for handling sensor data acquisition, decision-making logic, relay control, and communication with the mobile application. GPIO libraries are utilized to configure and control the input and output pins connected to sensors, relays, and the buzzer. Network-based communication is implemented using IP addressing, enabling the mobile application and Raspberry Pi to interact within the same Wi-Fi network. Additionally, voice commands received through the mobile application are processed and mapped to predefined control actions, allowing hands-free operation of appliances. The modular structure of the software ensures scalability, making it easy to integrate additional sensors or devices in future enhancements.

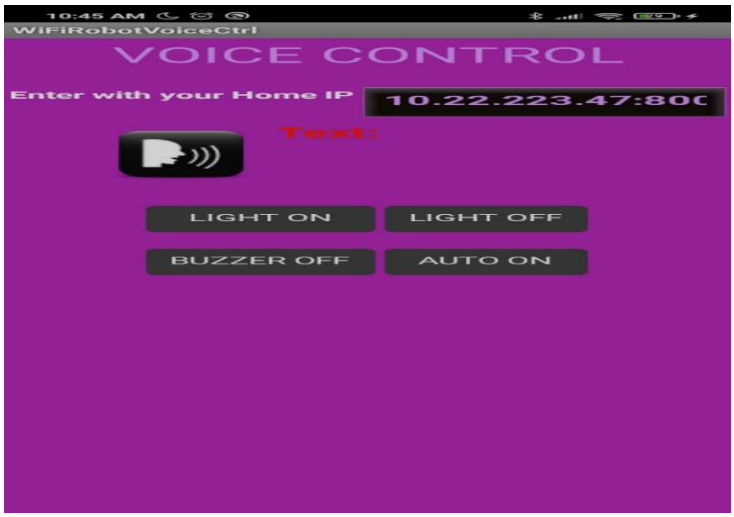


Figure.2. Software Implementation

7. Methodology

The methodology of the proposed system follows a structured approach to ensure reliable automation and safety monitoring. Initially, the Raspberry Pi initializes all GPIO pins and establishes a secure Wi-Fi connection. Sensor calibration is performed to define threshold values for the LDR and MQ-2 gas sensor. The system continuously reads real-time sensor data and compares it with predefined reference levels. In automatic mode, the LDR sensor data is used to control lighting based on ambient light conditions, thereby reducing unnecessary power consumption. Simultaneously, the MQ-2 gas sensor monitors the presence of combustible gases, and when the gas concentration exceeds the safe threshold, the system immediately activates the buzzer through the relay module to alert occupants. In manual mode, user commands received from the mobile application or voice input override the automated logic, allowing direct control of appliances. The inclusion of a battery backup ensures uninterrupted execution of the methodology during power outages, maintaining system functionality and safety monitoring at all times.

8. Flow of Operation

The flow of operation begins when power is supplied to the system through the main supply or battery backup. Upon startup, the Raspberry Pi loads the control program and establishes communication with the mobile application using its assigned IP address. The system then enters a continuous monitoring loop where sensor data from the LDR and MQ-2 sensor is periodically collected. If automatic mode is enabled, the system analyses the LDR sensor values to determine whether the lighting should be turned ON or OFF via the relay. In parallel, gas

sensor readings are constantly evaluated, and any detection of gas leakage results in immediate buzzer activation, irrespective of the operating mode. When a user sends a command through the mobile application or voice control interface, the system validates the request and performs the corresponding relay action. This process continues in real time, ensuring responsive control, energy efficiency, and enhanced safety. The flow is designed to be fault-tolerant, with the battery backup maintaining continuous operation during power failures. A battery backup unit is incorporated to ensure uninterrupted operation during power failures. The battery supplies power to the Raspberry Pi, sensors, and relay module, allowing continuous monitoring and alert generation even when the main power supply is unavailable. This improves system reliability and safety, especially for gas leakage detection.

9. Implementation Process

The implementation of the proposed smart home automation system begins with the hardware setup, where the Raspberry Pi 3 Model B+ is configured as the central controller. The Raspberry Pi OS is installed, and essential Python libraries such as GPIO are set up using the Thonny IDE. Sensors including the LDR and MQ-2 gas sensor are interfaced with the GPIO pins, while the relay module is connected to control household appliances and the buzzer. Proper power connections are ensured through the battery backup system to maintain uninterrupted operation during power failures.

In the next step, sensor data acquisition is implemented through Python programming. The Raspberry Pi continuously reads input signals from the LDR and MQ-2 gas sensor. Threshold values are predefined in the software to determine ambient light conditions and gas concentration levels. Based on these readings, the control logic evaluates whether to activate or deactivate connected devices, enabling automatic lighting control and gas leakage detection without manual intervention.

The third stage involves actuator control and alert generation. When low light intensity is detected, the Raspberry Pi triggers the relay module to switch ON the light, and it switches OFF the light when sufficient illumination is available. Similarly, if the MQ-2 sensor detects gas levels beyond the safety limit, the relay module activates the buzzer to provide an immediate alert. The relay ensures electrical isolation between high-voltage appliances and the low-voltage Raspberry Pi, maintaining operational safety.

Finally, network communication and voice-based control are integrated into the system. The Raspberry Pi connects to the local Wi-Fi network using IP-based communication, enabling interaction with a mobile application. Voice commands received through the application are processed by the software and mapped to predefined actions such as turning appliances ON or OFF. The modular software architecture allows easy expansion, ensuring that additional sensors or smart devices can be incorporated in future upgrades without major modifications.

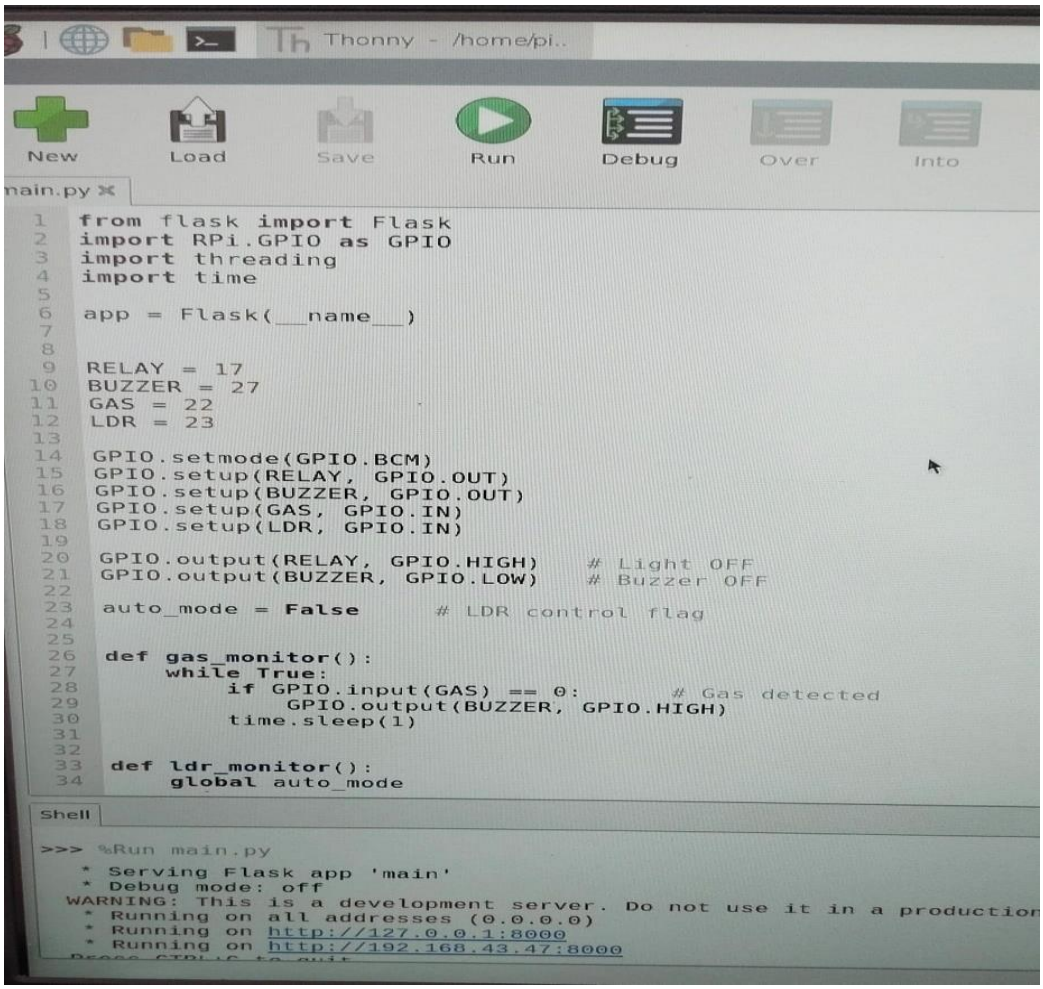


Figure.3. Program

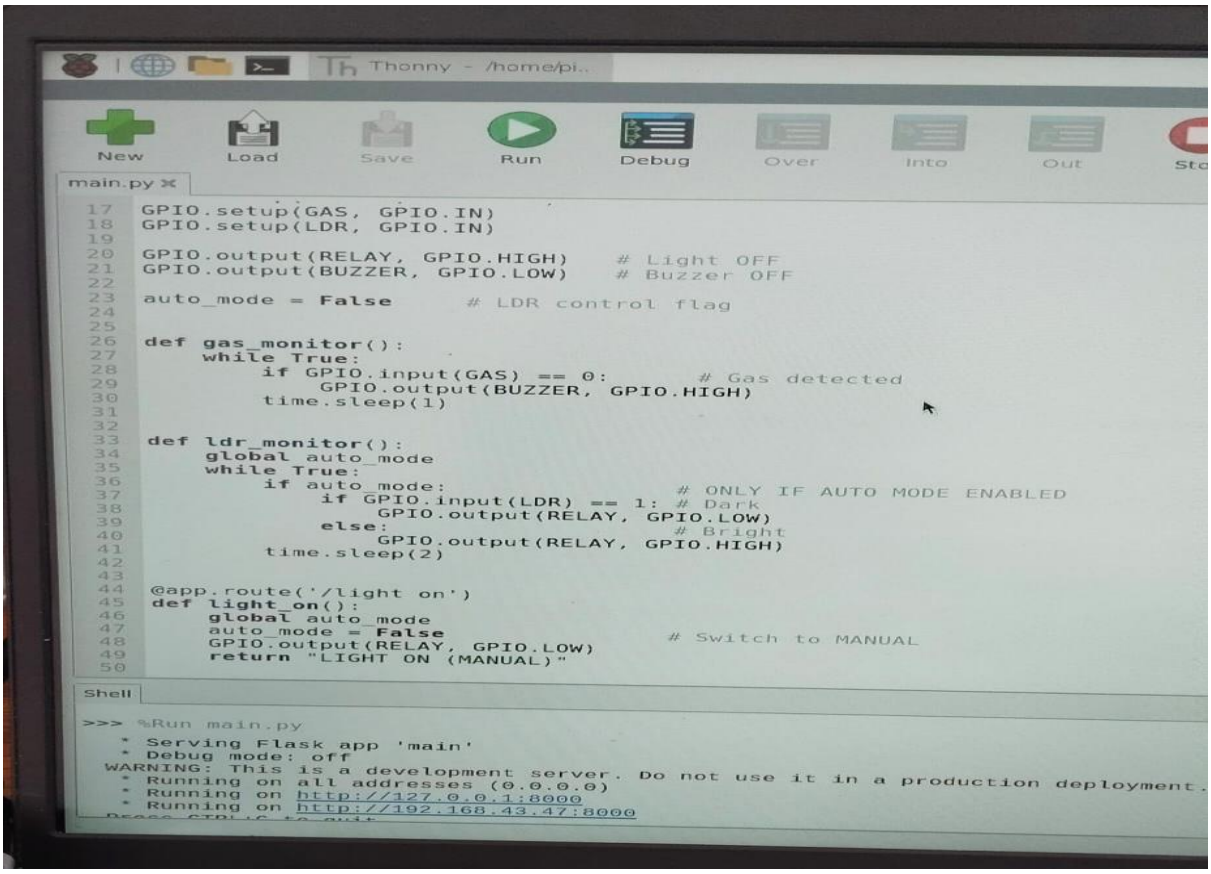


Figure.4. Program

10. Results

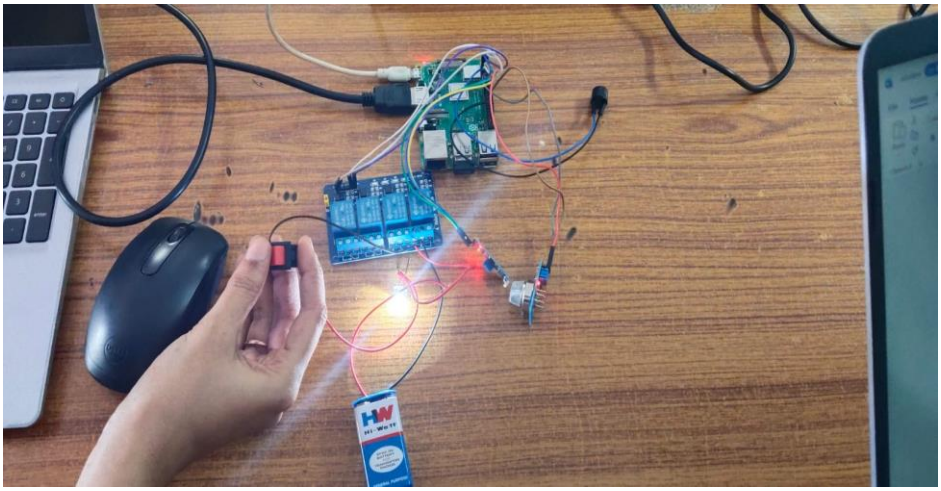
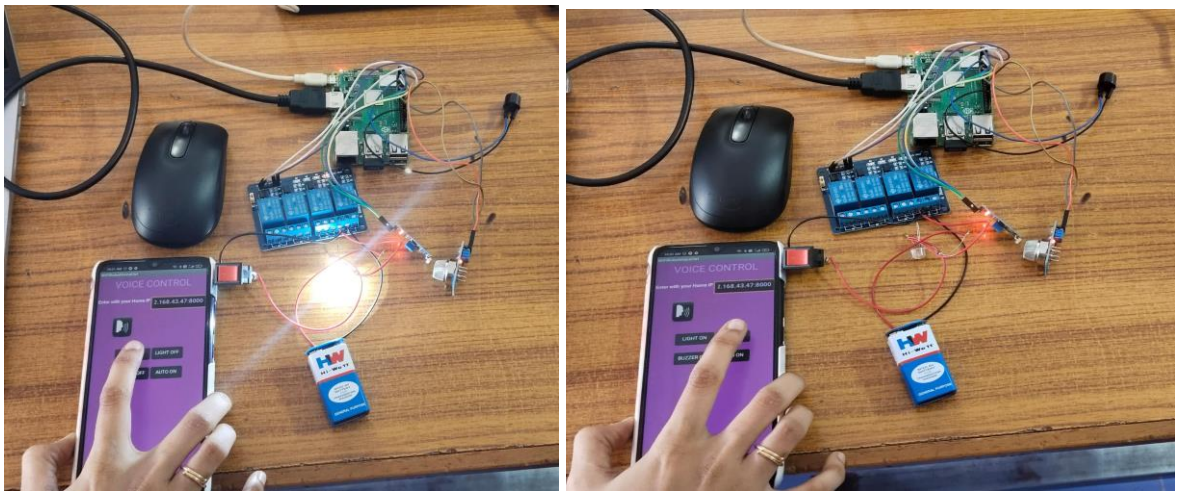


Figure.5. Output through switch

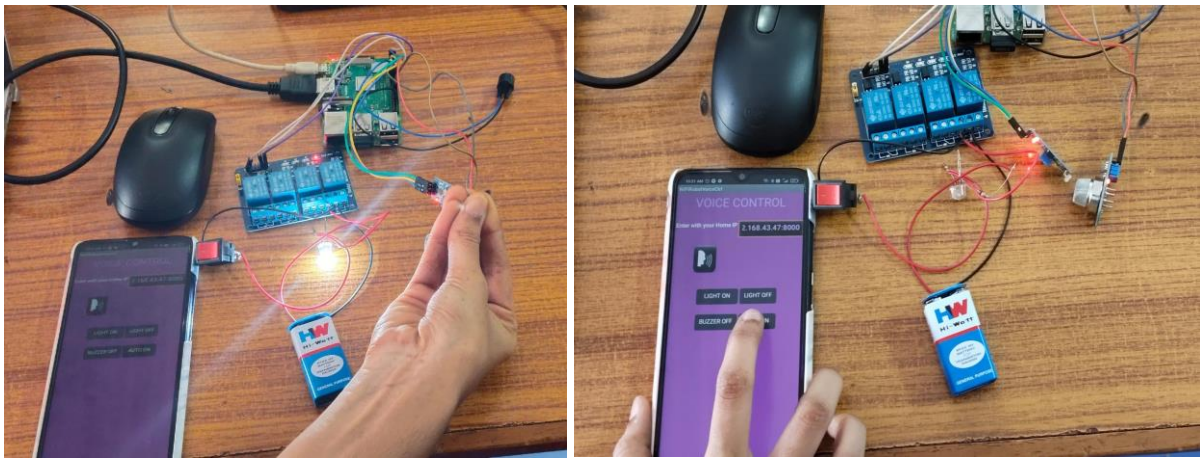
The output is obtained through manual switch control. When the switch is pressed, the input signal is sent to the controller, which activates the relay module. Based on the switch state (ON/OFF), the connected load is turned ON or OFF accordingly.



(a). Light ON **(b). Light OFF**

Figure.6. Output Through Mobile Application

The output is obtained through the mobile application by touching the Light ON/OFF button. When the button is pressed, the command is sent to the Raspberry Pi, which switches the load accordingly.



(a). Automatic Mode **(b). Manual Mode**

Figure.7. Output Through Auto Mode (LDR)

When the Auto ON option is enabled in the mobile application, the system switches to automatic mode. In this mode, the LDR sensor continuously monitors ambient light intensity. When the light level falls below the predefined threshold, the system automatically turns ON the light. As the ambient light increases and reaches or exceeds the threshold value, the system turns OFF the light, ensuring efficient and intelligent lighting control.

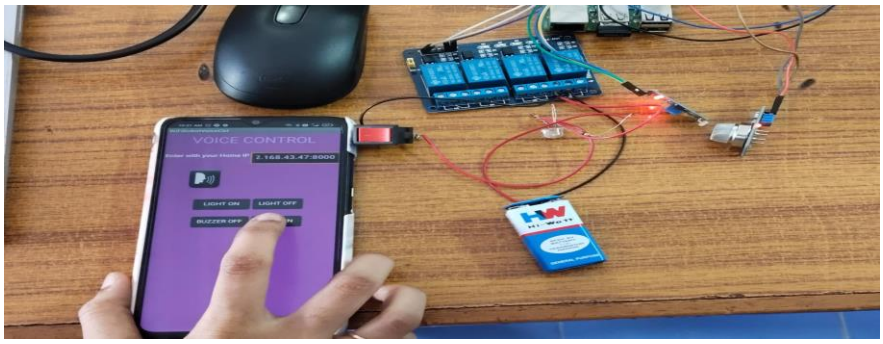


Figure.8. Output for Gas Detection

When the MQ-2 gas sensor detects gas leakage, it immediately triggers the buzzer to provide an audible alarm for safety. The buzzer alerts the user about the presence of hazardous gas in the environment. Through the mobile application, the user can remotely acknowledge the alert and turn OFF the buzzer by touching the Buzzer OFF option. This ensures quick response and convenient control during emergency situations.

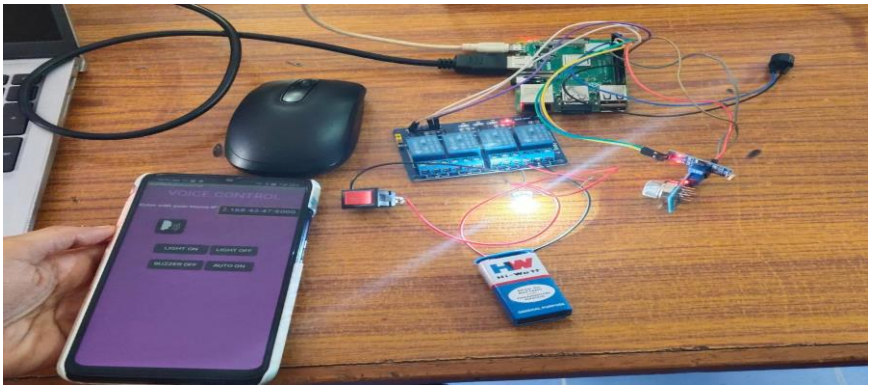


Figure.9. Output Through Voice Control

The system supports voice control through the mobile application. When the user says “Light ON”, the app recognizes the voice command and sends it to the controller, which turns the light ON. Similarly, when “Light OFF” is spoken, the light is switched OFF. The same operation can also be performed by manually touching the ON/OFF buttons in the mobile app. In addition, a physical switch is provided, allowing the user to control the light directly, ensuring reliable operation in both manual and voice-controlled modes.

Acknowledgement
The authors have no acknowledgements to declare.

Funding
This study has not received any funding from any institution/agency.

Conflict of Interest/Competing Interests
No conflict of interest.

Data Availability
The raw data supporting the findings of this research paper will be made available by the authors upon a reasonable request.

REFERENCES

[1]. A. R. Al-Ali, M. Qasaimeh, M. Al-Mardini, S. Radder, and I. A. Zualkernan, "ZigBee-based irrigation system for smart agriculture," IEEE International Conference on Automation Science and Engineering, 2012, pp. 512–517.
[2]. R. Piyare and M. Tazil, "Bluetooth based home automation system using cell phone," IEEE 15th International Symposium on Consumer Electronics, 2011, pp. 192–195.
[3]. R. Antonucci, A. Conti, and D. Masotti, "Raspberry Pi-based smart home," International Conference on Consumer Electronics (ICCE), IEEE, 2016, pp. 390–391.
[4]. K. Gill, S. H. Yang, F. Yao, and X. Lu, "A ZigBee-based home automation system," IEEE Transactions on Consumer Electronics, vol. 55, no. 2, pp. 422–430, May 2009.
[5]. M. Collotta and G. Pau, "A novel energy management approach for smart homes using Bluetooth Low Energy," IEEE Journal on Selected Areas in Communications, vol. 33, no. 12, pp. 2988–2996, Dec. 2015.