Project Report on

IOT Smart Home Automation

Submitted to

Savitribai Phule Pune University

In the partial fulfillment of the requirement of the award of the degree of

Bachelor of Business Administration- Computer Application,

TYBBA -CA, Sem: V

Academic Year 2024-25

By

Hemant Rambhau Dhavale

Under the guidance of

Prof. Amol Kale

Through



Alandi (D)- Pune 412105

**Alandi (D) - Pune 412105**

CERTIFICATE

Department of Computer Application

This is to certify that Mr. **Hemant Rambhau Dhavale**, of TYBBA-CA, Sem V, Exam Seat No.\_\_\_\_\_\_\_\_\_\_\_\_\_ has successfully completed project work entitled **IOT Smart Home Automation** in the partial fulfillment of the requirement of the degree of Bachelor of Business Administrator-Computer Application for the Academic Year 2024-25

Prof. Amol Kale Dr. Vikas Mahandule

Project Guide Head of the Department

Internal Examiner External Examiner

**Acknowledgement**

The words are not enough to express my thanks to **Dr. B. B. Waphare**, **Director, MIT Arts Commerce and Science College Alandi (D)** for providing me with the opportunity to avail the excellent facilities and infrastructure of the institute.

It is my proud privilege to express my profound gratitude to **Dr. Vikas Mahandule, HOD, Computer Application Department**, for his astute guidance, constant encouragement and sincere support throughout my academic course.

I thanks to my honest gratitude to **Prof. Amol Kale** for his inspiration, constructive suggestions and affectionate guidance in my project work completion.

Last but not the least, I express my sincere thanks to all my dear friends and family members for their constant motivation, moral support and invariable direction throughout my life.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SR.NO** | **CONTENT** | **PAGE NO** |
| **1** | **Abstract Of Project** | **6-8** |
|  | 1.1 Introduction | 6 |
|  | 1.2 Motivation | 6 |
|  | 1.3 Problem Statement | 6 |
|  | 1.4 Background | 7 |
|  | 1.5 Objective | 7 |
|  | 1.6 Scope | 7 |
|  | 1.7 Literature Review | 8 |
|  | 1.8 Existing System | 8 |
|  | 1.9 Proposed System | 8 |
| **2** | **Requirement Analysis** | **9-10** |
|  | 2.1 Problem Definition | 9 |
|  | 2.2 Requirement Specification | 9 |
|  | 2.3 Feasibility Study | 10 |
|  | 2.4 System Analysis | 10 |
| **3** | **System Design** | **11-21** |
|  | 3.1 Block Diagram | 11 |
|  | 3.2 Class Diagram | 12 |
|  | 3.3 Activity Diagram | 13 |
|  | 3.4 Use Case Diagram | 14 |
|  | 3.5 Object Diagram | 15 |
|  | 3.6 Sequence Diagram | 16 |
|  | 3.7 State Diagram | 17 |
|  | 3.8 Component Diagram | 18 |
|  | 3.9 Context Level Diagram | 19 |
|  | 3.10 Data Flow Diagram | 20 |
|  | 3.11 Entity Relationship Diagram | 21 |
| **4** | **Project Code** | **22-25** |
| **5** | **Output and Reporting Testing** | **26-29** |
| **6** | **Advantages and Disadvantages** | **30-31** |
| **7** | **Future Scope** | **32** |
| **8** | **Conclusion** | **33** |
| **9** | **Final Thought** | **34** |
| **10** | **References and Bibliography** | **35** |

**Abstract**

1. **Introduction**

In recent years, the concept of home automation has gained significant traction, driven by advancements in technology and the increasing demand for convenience and security in our daily lives. Home automation systems enable users to remotely control and monitor various household devices, improving energy efficiency, enhancing comfort, and bolstering security. This project aims to develop a Smart Home Automation System using Arduino, focusing on creating a user-friendly interface that allows for seamless interaction with connected devices.

This project has an Arduino with an ESP8266 Wi-Fi module, sensors to manage a home. It uses a 4-channel relay to control two bulbs and a 4V DC motor. A DHT11 sensor monitors temperature and humidity, while an MQ135 sensor checks air quality, and a PIR sensor detects motion. The data from these sensors is displayed on a 16x2 LCD and sent to a web server via the ESP8266(Wi-Fi). In case of smoke detection, a servo motor opens the house door for emergency access. Users can control relays and view sensor data remotely via a mobile app.

1. **Motivation:**

The increasing demand for smart home systems that provide convenience, energy efficiency, and improved security motivates this project. People desire solutions that allow them to remotely control and automate daily tasks while maintaining the safety of their homes, especially when they are away.

1. **Problem Statement**

The increasing need for energy efficiency, enhanced security, and convenience in daily life drives the demand for home automation. Traditional homes lack the capability for remote control and real-time monitoring of appliances and security systems, leading to inefficiencies and potential safety risks.

1. **Background:**

With the advent of IoT, the traditional concept of home automation has transformed. Wireless control, sensor integration, and cloud connectivity have made it possible to automate homes and monitor them in real time. Arduino, combined with sensors and Wi-Fi modules, is frequently used for developing affordable and flexible home automation systems.

1. **Objective of the Project**

The primary goal of this project is to develop a Smart Home Automation system that allows users to remotely control and monitor household appliances using a smartphone application. The system should also enhance home security and energy efficiency through automated control and real-time feedback.

Key objectives include:

* **Remote Control** of home appliances via a smartphone.
* **Real-time Monitoring** of environmental conditions (temperature, humidity, air quality).
* **Security Alerts** in response to unauthorized movement detection.
* **Emergency Response** in case of smoke detection, providing a mechanism for immediate action.

1. **Scope of the Project**

This project will implement a smart home automation system focusing on the following:

* Controlling home appliances like lights, fans, and motors using a mobile app.
* Monitoring environmental parameters like temperature, humidity, air quality, and motion.
* Providing security alerts in case of unauthorized access and emergency actions during smoke detection.
* Future scalability by adding more devices and functionalities such as voice control and machine learning-based automation.

1. **Literature Review:**

Several studies have explored the use of Arduino and IoT in home automation, demonstrating how remote monitoring and control of home devices are possible using wireless networks. While existing systems provide these capabilities, many are costly or limited in their features. This project builds on these studies to create a more accessible and scalable solution.

1. **Existing System:**

Current home automation systems offer features like remote control and energy monitoring, but many require proprietary hardware, are expensive, or lack customization options. Additionally, user interfaces are often not intuitive, making them difficult for users to operate or expand.

1. **Purpose System**

The proposed system in this project improves on existing solutions by providing a more affordable and flexible Smart Home Automation system. It uses an Arduino microcontroller, ESP8266 Wi-Fi module, and sensors to remotely control appliances and monitor environmental conditions. The system allows for real-time data visualization through a mobile app, and includes security alerts for motion and smoke detection. Unlike other systems, it is designed to be user-friendly, with easy scalability for future upgrades like voice control and machine learning-based automation.

Requirement Analysis

* Problem Definition

The traditional home setup often lacks the capability for remote control and real-time monitoring of appliances and security systems. This leads to inefficiencies in energy consumption and potential security risks. There’s a need for an integrated system that allows homeowners to manage their environment conveniently and securely from a distance.

* **Requirement Specification**

This section defines the hardware and software components necessary to implement the Smart Home Automation system.

**Hardware Requirements:**

* + **Arduino (Microcontroller):** Acts as the central controller for the entire system.
  + **ESP8266 Wi-Fi Module:** Facilitates wireless connectivity to enable remote control and monitoring.
  + **4-Channel Relay Module:** Controls household appliances such as lights and motors.
  + **DHT11 Sensor:** Measures temperature and humidity levels.
  + **MQ-135 Sensor:** Monitors air quality by detecting gases like smoke and carbon dioxide.
  + **PIR Sensor:** Detects motion for security alerts.
  + **Servo Motor:** Operates the door opening mechanism in case of emergencies.
  + **LCD 16x2:** Displays real-time sensor data (temperature, humidity, air quality).
  + **Bulbs and 4V DC Motor:** Serve as controllable devices to demonstrate the automation system.

**Software Requirements:**

* + **Arduino IDE:** To write and upload code to the Arduino board.
  + **ESP8266 Libraries:** For Wi-Fi communication.
  + **RemoteXY or Blynk Platform:** For developing the mobile application interface.
  + **Cloud Services (RemoteXY/Blynk Cloud):** For hosting the web server that communicates with the Arduino and mobile app.
* **Feasibility Study**

# **Technical Feasibility:** The project utilizes widely available components like Arduino, sensors, and Wi-Fi modules. The required technical knowledge involves basic electronics and programming skills, making the project technically feasible.

# **Economic Feasibility:** The cost of components and software tools is relatively low, making the project economically viable. The use of open-source platforms further reduces expenses.

# **Operational Feasibility:** The system is designed to be user-friendly and accessible via a mobile application. This ensures that users can easily control and monitor their home devices, making the project operationally feasible.

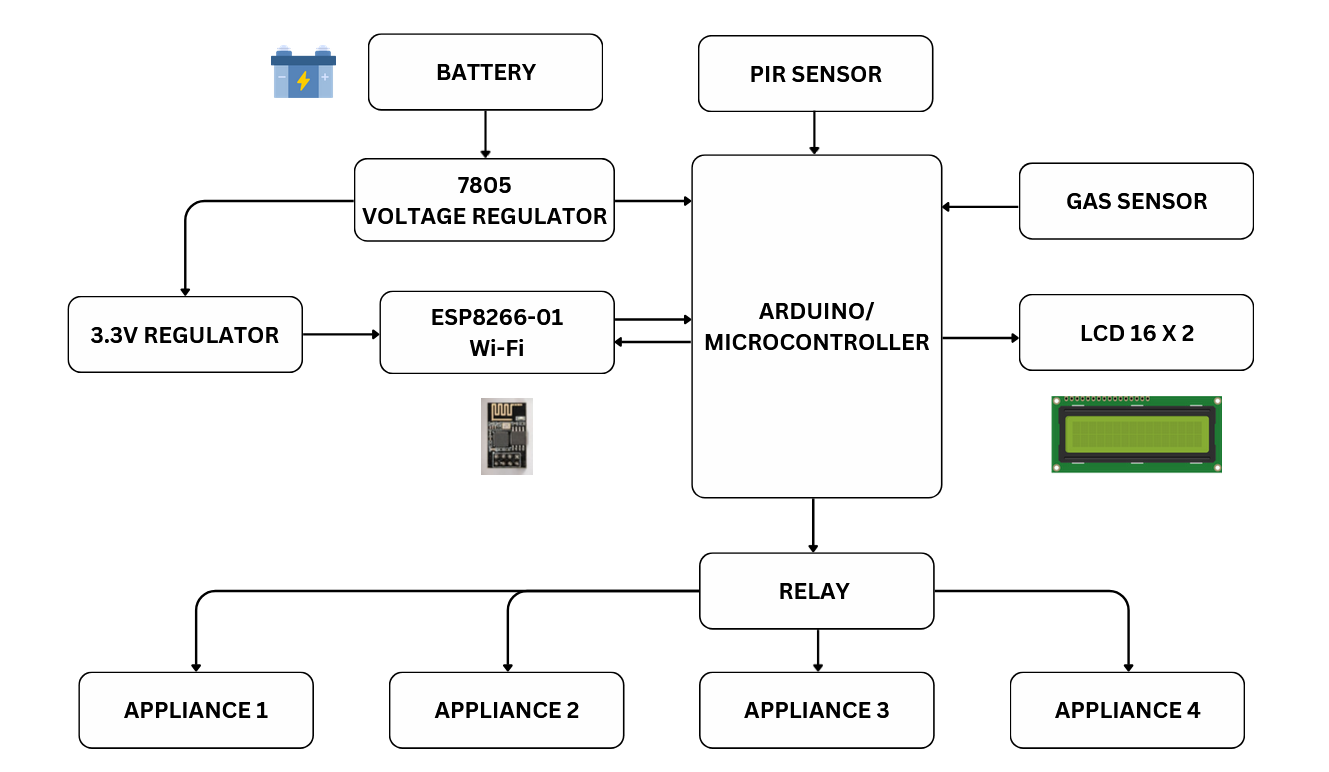
# **Legal Feasibility:** The project adheres to standard regulations and does not infringe on any copyrights or patents. All components and software used are compliant with legal requirements.

# **System Analysis**

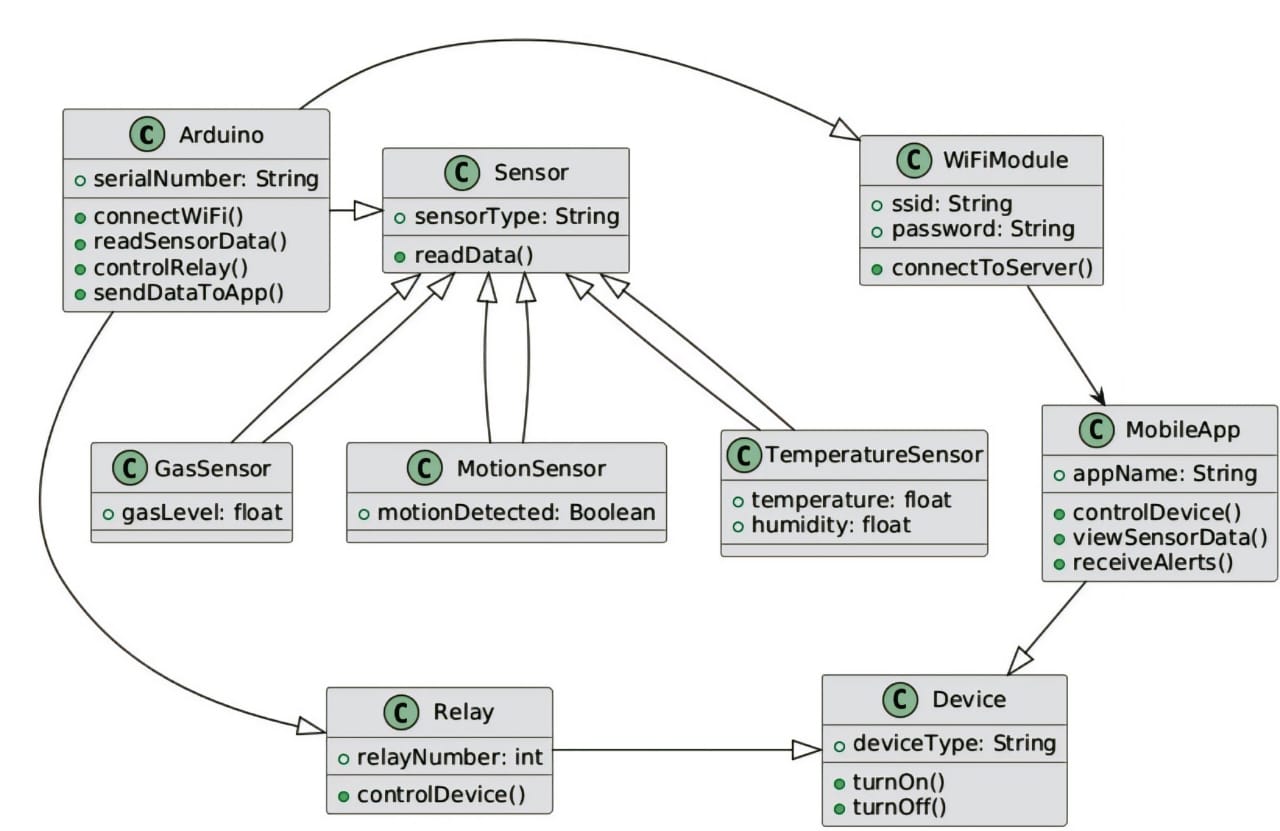
* + - * **Requirement Analysis:** Understanding the needs of homeowners for remote control and monitoring, energy efficiency, and enhanced security. Identifying the specific sensors and actuators required for the system.
      * **System Design:** Creating a detailed design of the system architecture, including block diagrams and circuit schematics. Planning the integration of sensors, relays, the Wi-Fi module, and the Arduino.
      * **Deployment:** Setting up the hardware components and configuring the mobile application. Ensuring seamless communication between the hardware and the app through the internet.
      * **Testing:** Conducting thorough testing to verify sensor accuracy, relay functionality, and app communication. Ensuring the system responds correctly to user commands and provides real-time updates.
      * **Maintenance:** Establishing a plan for regular updates and maintenance to ensure long-term reliability and performance of the system.

System Design

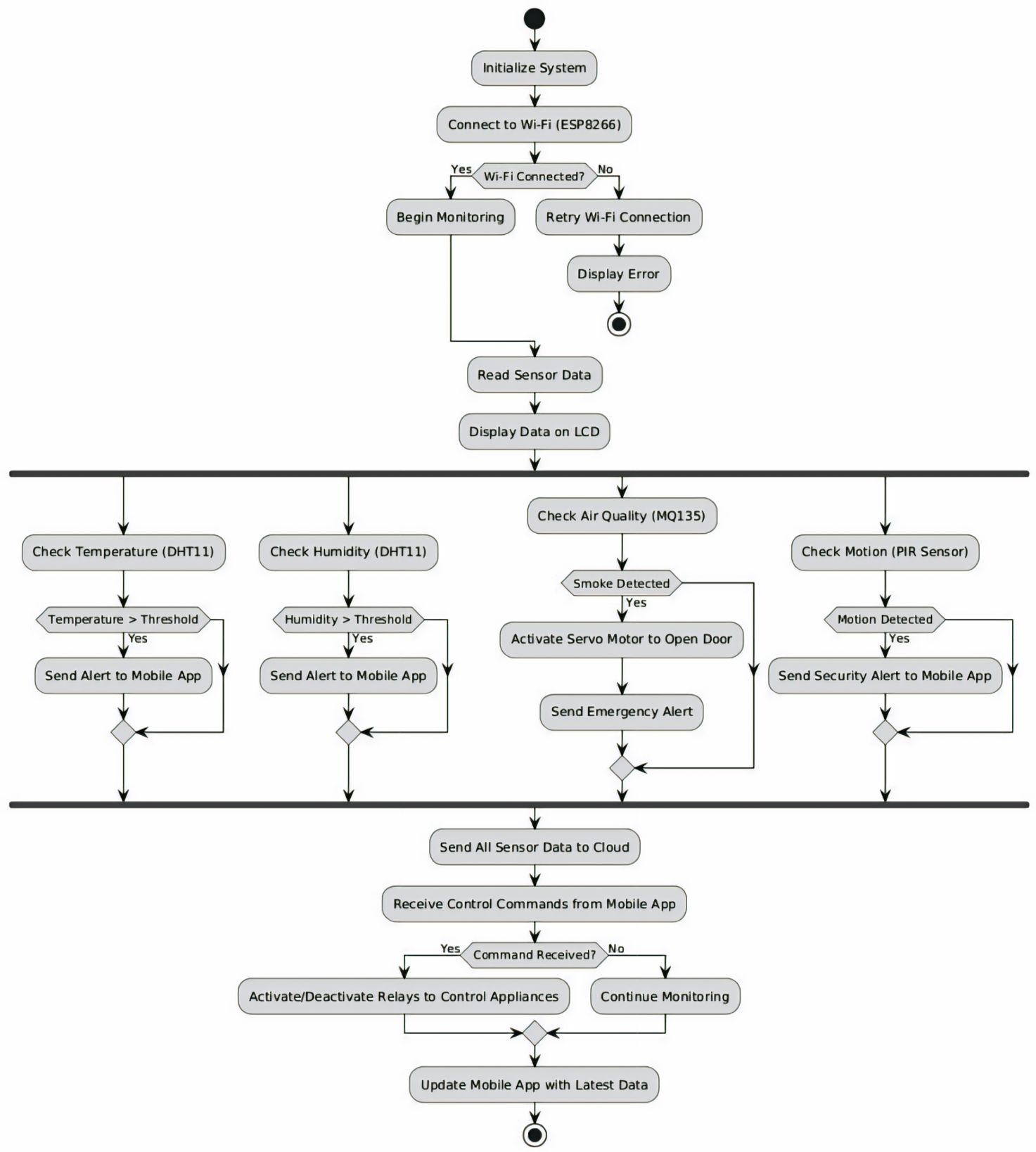
Block Diagram



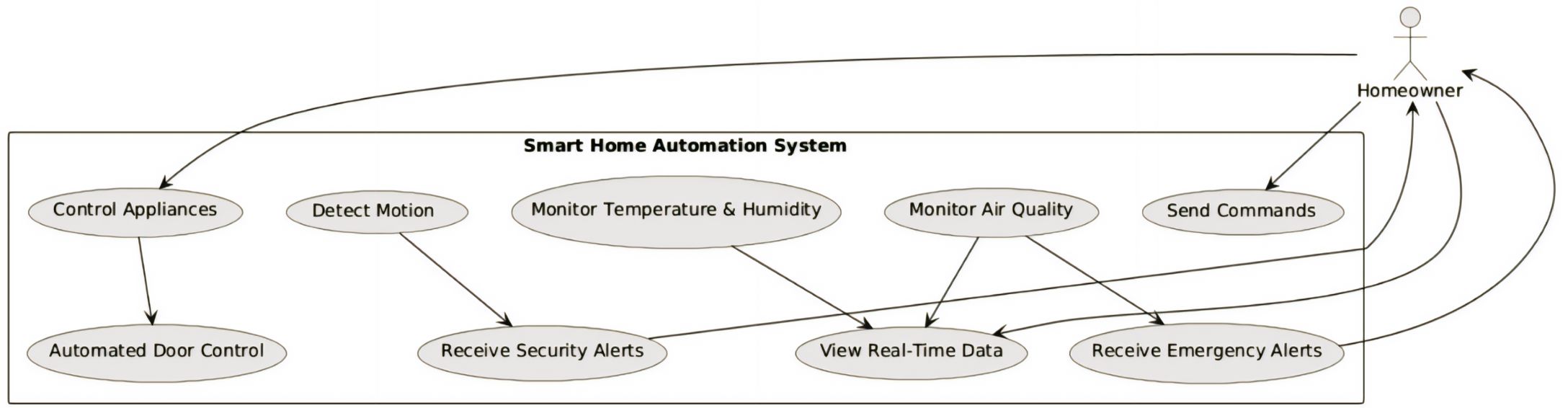
Class Diagram



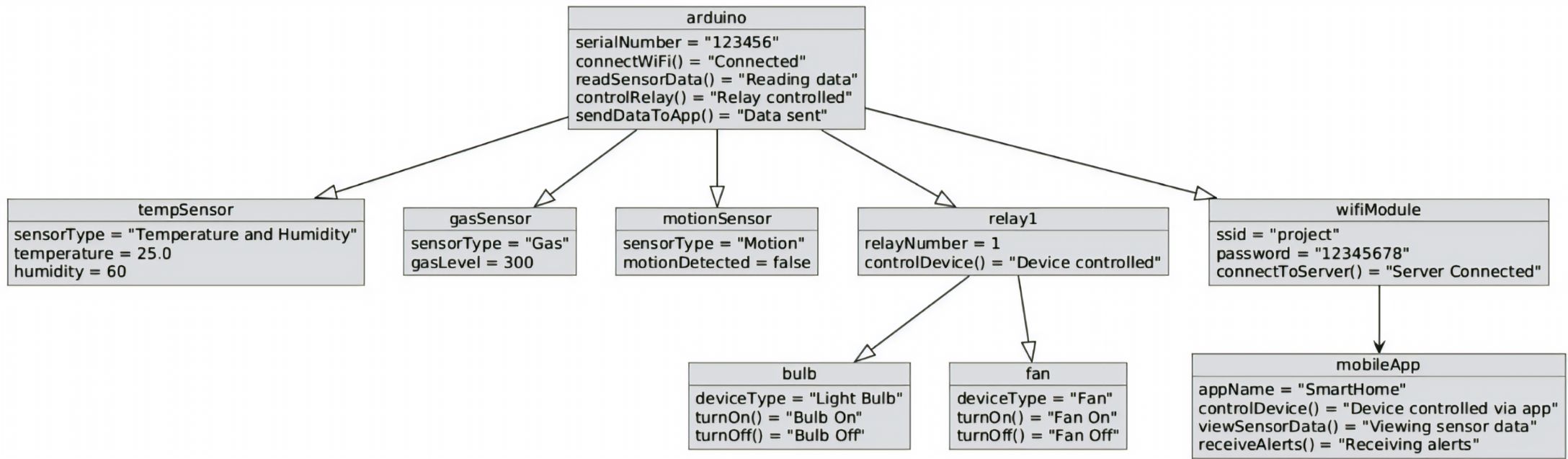
Activity Diagram



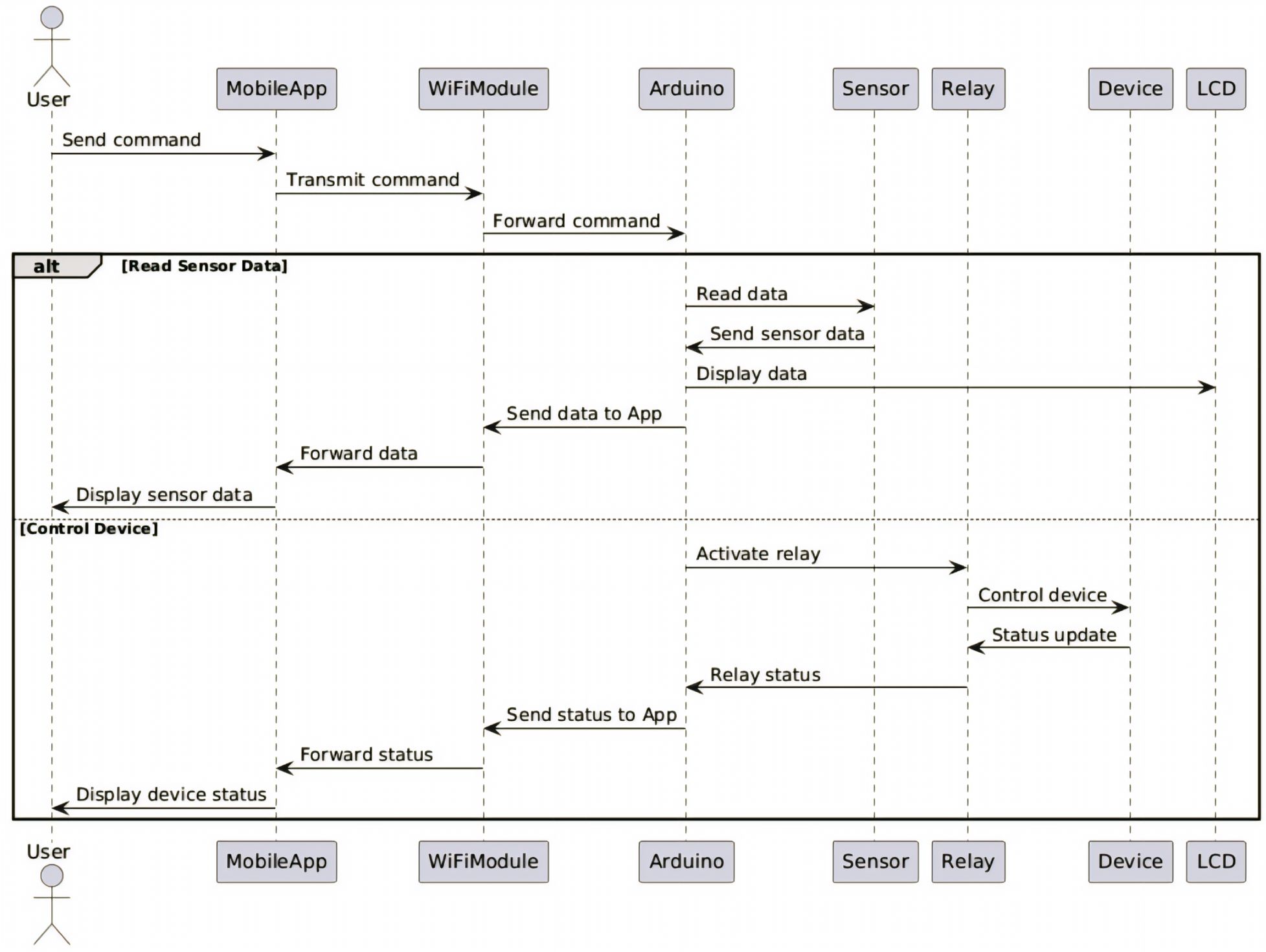
Use Case Diagram



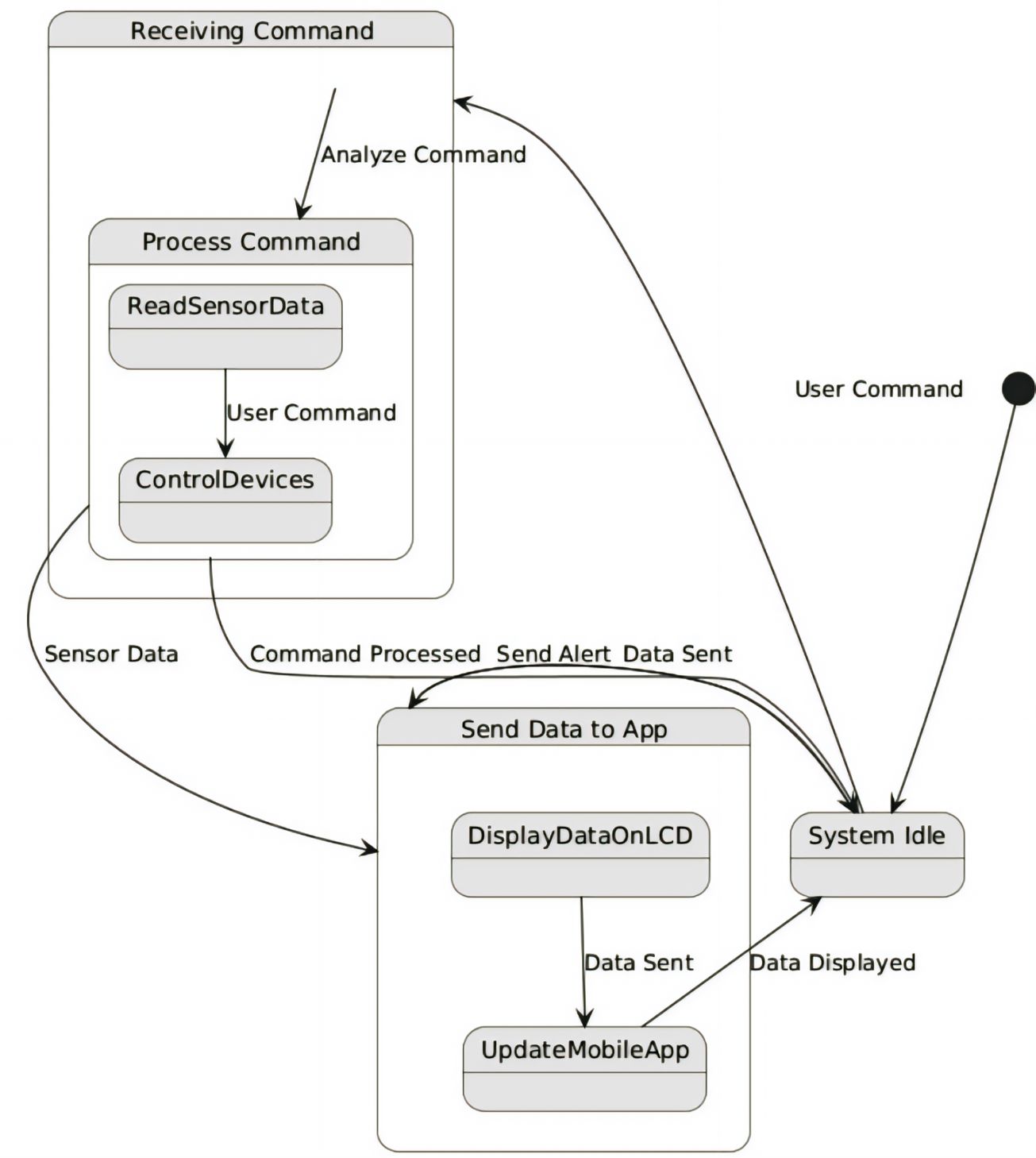
Object Diagram



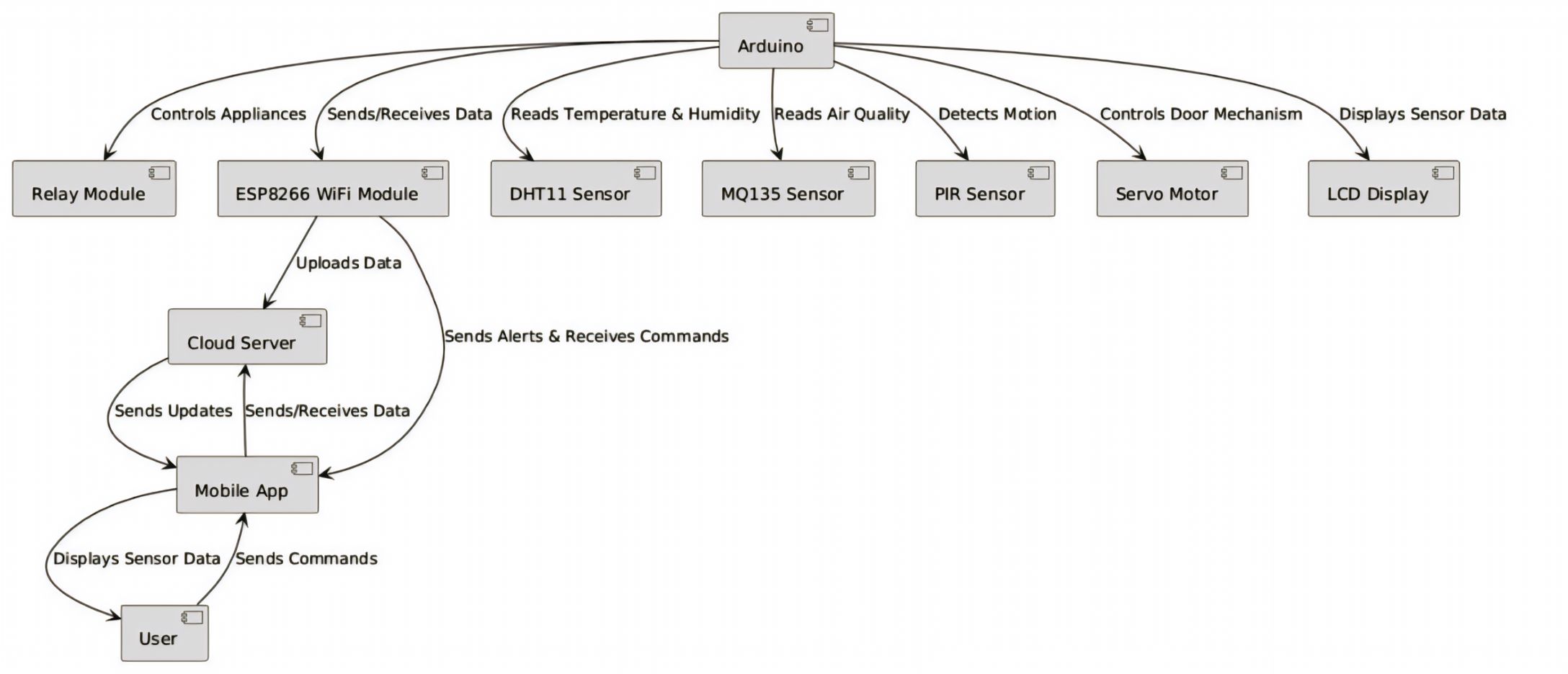
Sequence Diagram



State Diagram



Component Diagram



Context Level Diagram (CLD):

**0.0**

e **Smart Home Automation**

User send Command

Send Device Status Report

Budget Details

Send Security Status updates

Receive Status & Notification

**Homeowner**

**Smart Device**

**Security System**

**Owner**

**Smart Device**

**Security System**

Receive On/Off

Control command

Receive Security Alert

Entity Relationship Diagram (ERD)

Own

1

M

1

Smart Device

Owner

1

Receive

Manage

M

M

Execute

Security System

Command

M

1

**Project Code**

/////////////////////////////////////////////

// RemoteXY include library //

//////////////////////////////////////////////

// you can enable debug logging to Serial at 115200

//#define REMOTEXY\_\_DEBUGLOG

// RemoteXY select connection mode and include library

#define REMOTEXY\_MODE\_\_ESP8266\_HARDSERIAL\_CLOUD

// RemoteXY connection settings

#define REMOTEXY\_SERIAL Serial

#define REMOTEXY\_SERIAL\_SPEED 115200

#define REMOTEXY\_WIFI\_SSID "project"

#define REMOTEXY\_WIFI\_PASSWORD "12345678"

#define REMOTEXY\_CLOUD\_SERVER "cloud.remotexy.com"

#define REMOTEXY\_CLOUD\_PORT 6376

#define REMOTEXY\_CLOUD\_TOKEN "b4ff66f8abbb3089194be9cc731fab6d"

#include <RemoteXY.h>

// RemoteXY GUI configuration

#pragma pack(push, 1)

uint8\_t RemoteXY\_CONF[] = // 86 bytes

{ 255,3,0,13,0,79,0,17,0,0,0,31,1,126,200,1,1,5,0,10,

91,14,28,28,48,4,26,31,79,78,0,31,79,70,70,0,69,84,68,20,

20,0,1,67,16,74,40,10,4,2,26,11,10,16,15,26,26,48,4,26,

31,79,78,0,31,79,70,70,0,10,54,15,26,26,48,4,26,31,79,78,

0,31,79,70,70,0 };

// this structure defines all the variables and events of your control interface

struct {

// input variables

uint8\_t pushSwitch\_1; // =1 if state is ON, else =0

uint8\_t pushSwitch\_01; // =1 if state is ON, else =0

uint8\_t pushSwitch\_02; // =1 if state is ON, else =0

// output variables

int16\_t sound\_1; // =0 no sound, else ID of sound, =1001 for example, look sound list in app

char text\_1[11]; // string UTF8 end zero

// other variable

uint8\_t connect\_flag; // =1 if wire connected, else =0

} RemoteXY;

#pragma pack(pop)

#include <Servo.h>

Servo myservo; // create servo object to control a servo

// twelve servo objects can be created on most boards

int pos = 0; // variable to store the servo position

/////////////////////////////////////////////

// END RemoteXY include //

/////////////////////////////////////////////

#include <LiquidCrystal.h>

LiquidCrystal lcd(7,8,9,10,11,12);

#include <stdlib.h>

#include <dht.h>

#define dht\_dpin A0

int i, j, k;

dht DHT;

void setup()

{

lcd.begin(16,2);

lcd.setCursor(0,0);

lcd.print("DHT Gas Sensor");

lcd.setCursor(0,1);

lcd.print(" ");

delay(2000);

lcd.clear();

RemoteXY\_Init ();

myservo.attach(A3);

myservo.write(pos);

pinMode(2, OUTPUT);

digitalWrite(2, HIGH);

pinMode(3, OUTPUT);

digitalWrite(3, HIGH);

pinMode(4, OUTPUT);

digitalWrite(4, HIGH);

pinMode(A2, INPUT);

pinMode(A1, INPUT);

// TODO you setup code

}

void loop()

{

RemoteXY\_Handler ();

DHT.read11(dht\_dpin);

i=DHT.humidity;

j =DHT.temperature;

k= analogRead(A1);

lcd.setCursor(0 ,0);

lcd.print("T: ");

lcd.print(j);

// Serial.println(j);

lcd.print(" C ");

lcd.setCursor(8 ,0);

lcd.print("H: ");

lcd.print(i);

lcd.print(" % ");

lcd.setCursor(0 ,1);

lcd.print("G: ");

lcd.print(k);

lcd.print(" ");

if(k>600)

{

myservo.write(90);

delay(3000);

myservo.write(0);

}

if(j >40 || i>70 || k>600)

{

strcpy (RemoteXY.text\_1, "Alert!! ");

}

else

{

strcpy (RemoteXY.text\_1, " ");

}

if (digitalRead(A2) == HIGH) {

RemoteXY.sound\_1 = 1001;

}

else {

RemoteXY.sound\_1 = 0;

}

if (RemoteXY.pushSwitch\_01!=0) {

/\* button pressed \*/

digitalWrite(2, LOW);

}

else {

/\* button not pressed \*/

digitalWrite(2, HIGH);

}

if (RemoteXY.pushSwitch\_02!=0) {

/\* button pressed \*/

digitalWrite(3, LOW);

}

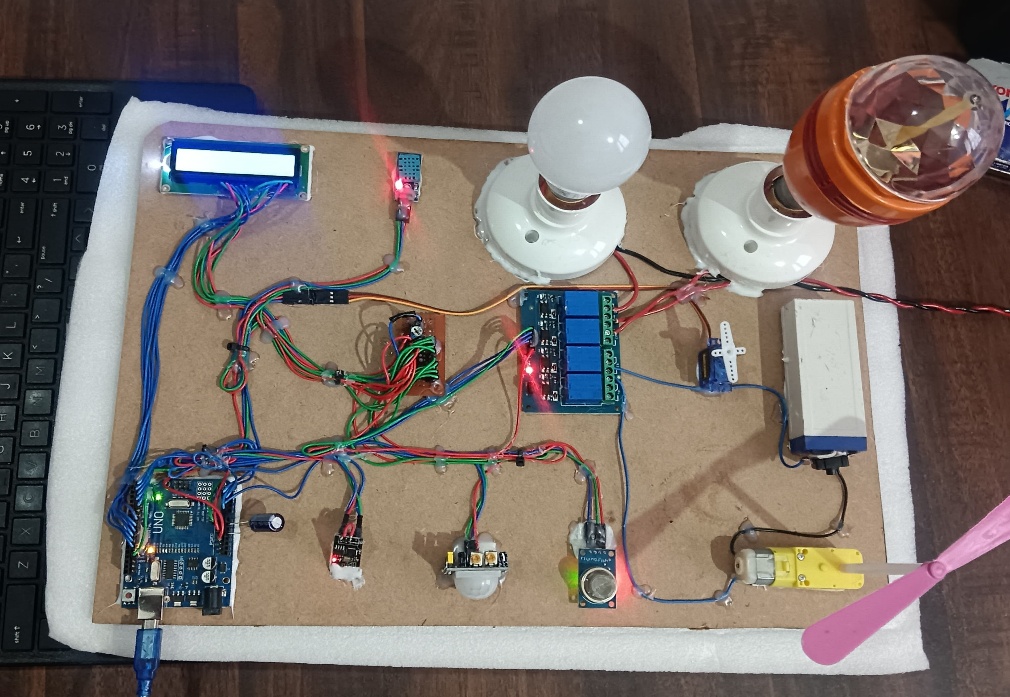
else {

/\* button not pressed \*/

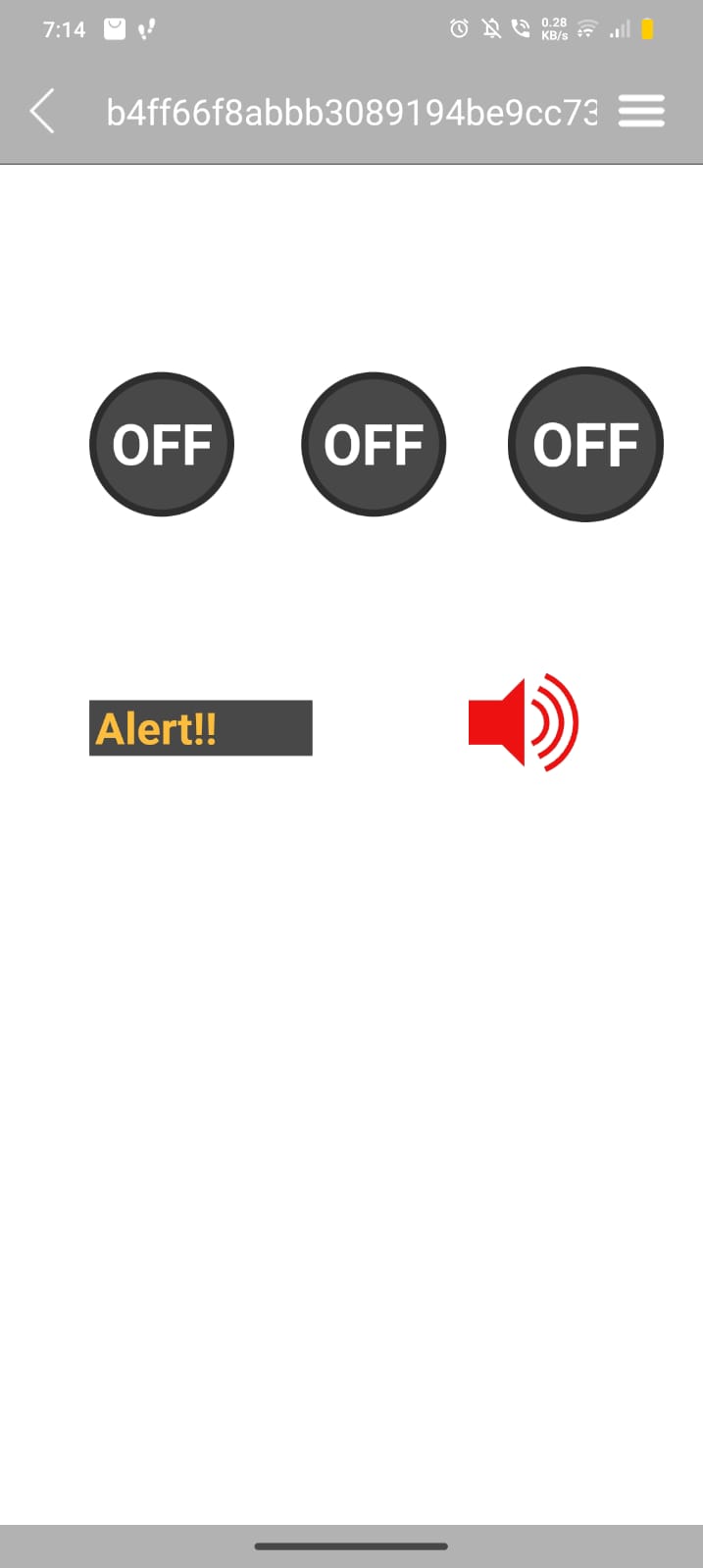
digitalWrite(3, HIGH);

**Project Interface**

****

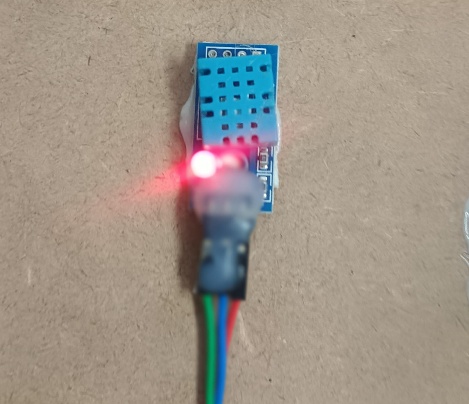
****

**User App Interface**

****

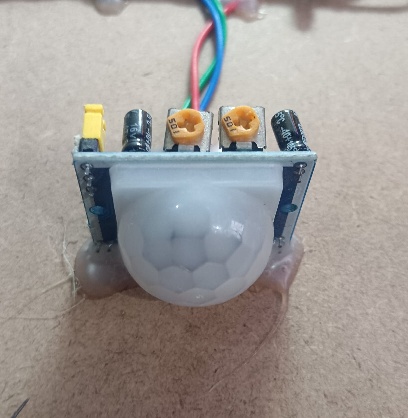
**Sensor Data Display**

****

** **

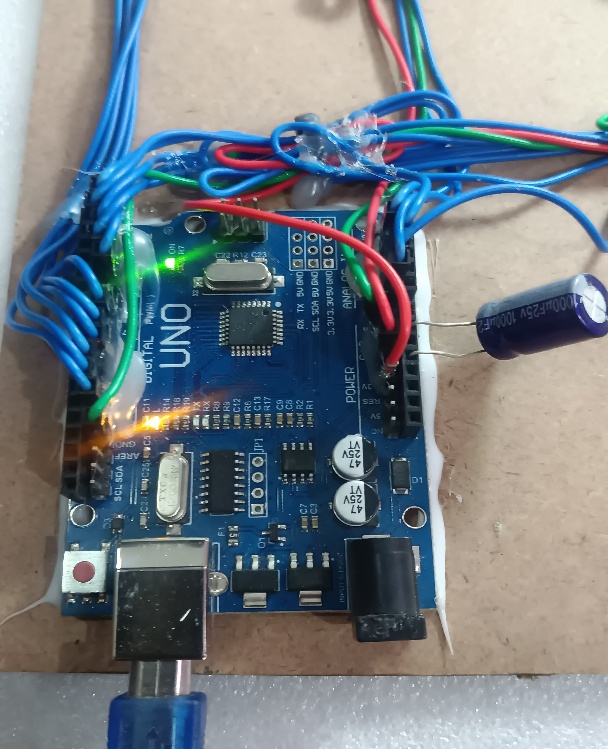
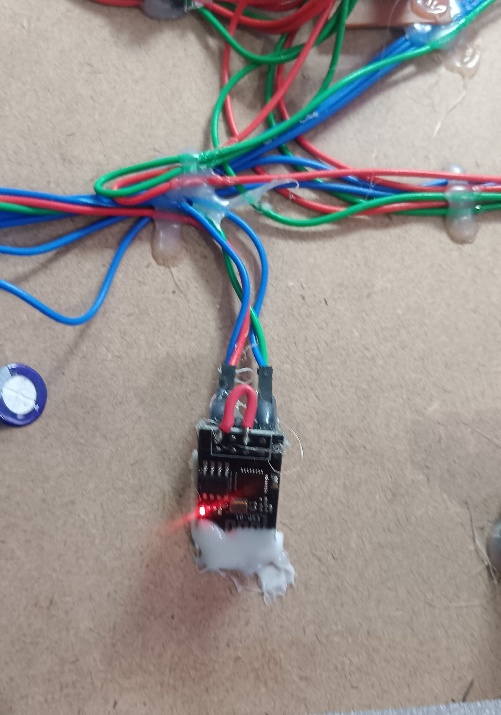
**DH11 Sensor (Temperature & MQ135 Sensor (Gas Sensor)**

**Humidity detect)**

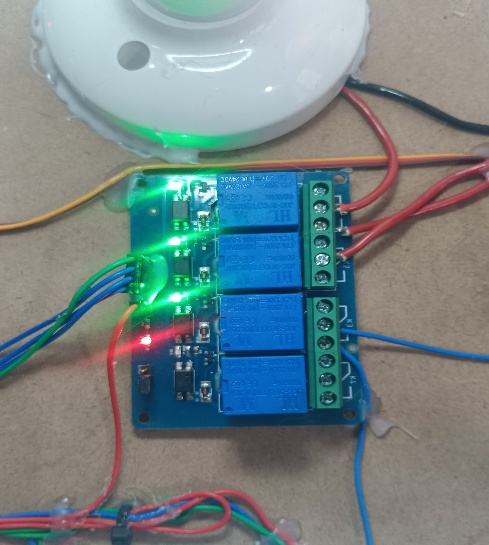
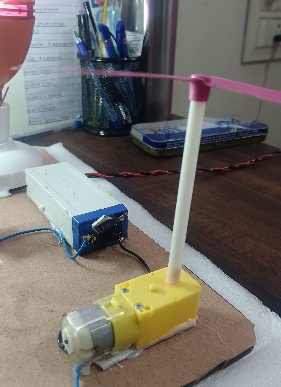
** **

**PIR Sensor ( Detect movement Gas Regulator (For emergency)**

**send alert on user interface (App)**

** **

**Arduino ESP8266 WIFI Module**

**  **

**Relay Module R1 & R2 ( Light ) R3 ( Fan )**

**Advantages**

1. **Convenience**:

The system allows users to control devices remotely through a mobile app, providing convenience in managing home appliances without physically interacting with them.

1. **Energy Efficiency:**

With automation rules and smart sensors like temperature/humidity sensors, the system can optimize energy usage by turning off devices when not needed or adjusting them based on environmental conditions.

1. **Improved Security:**

Security features like motion sensors (PIR), door sensors, and real-time monitoring provide alerts for unusual activity, enhancing home security.

1. **Customizability:**

Users can set up automation rules to trigger devices based on specific conditions, offering personalized control over the home environment.

1. **Real-time Monitoring:**

The system continuously monitors environmental factors like temperature, humidity, and motion, enabling better management of home conditions and the ability to track potential issues instantly.

1. **Remote Access:**

Users can manage and monitor the home automation system remotely, receiving alerts and status updates in real time.

1. **Integration with Voice Assistants:**

Integration with platforms like Google Assistant and Alexa offers an additional layer of control, making it even more user-friendly.

1. **Maintenance Alerts:**

The system can generate maintenance requests based on sensor readings or system malfunctions, enabling proactive maintenance.

**Disadvantages**

1. High Initial Setup Cost:

Installing IoT-based smart home systems can be expensive due to the cost of sensors, smart devices, controllers, and networking equipment.

1. **Complex Installation:**

While the system is effective, it requires a fair level of technical expertise to set up, especially with respect to integrating different sensors, smart devices, and configuring automation rules.

1. **Privacy and Security Concerns:**

IoT devices are vulnerable to hacking or data breaches. Unauthorized access to a smart home system could lead to serious security concerns, such as the control of home devices being taken over by outsiders.

1. **Reliance on Internet:**

The system depends heavily on a stable Wi-Fi connection. If the network fails or is compromised, the functionality of the home automation system will be significantly impacted.

1. **Maintenance and Support:**

Ongoing maintenance is required to ensure that sensors, devices, and systems are updated, functional, and secure. Without proper support, the system may degrade over time.

1. **Potential Compatibility Issues:**

Not all smart devices and sensors are compatible with each other, which could lead to limitations when trying to add or update devices in the system.

1. **Power Dependency:**

The system relies on power for devices, sensors, and Wi-Fi. In the event of a power outage, the entire automation system may cease to function unless there is a backup power source.

1. **Learning Curve:**

For users unfamiliar with IoT and smart technology, there may be a significant learning curve in understanding how to operate and customize the system effectively.

### Future Scope

* **Enhanced Security:** Implementing biometric authentication, advanced security analytics, and possibly blockchain for better protection.
* **AI Integration:** Adding AI for predictive automation and smarter controls, like voice and gesture recognition.
* **Energy Efficiency:** Integrating with smart grids and renewable energy systems to optimize energy usage.
* **Healthcare Integration:** Connecting with health-monitoring devices for wellness tracking and assisted living.
* **5G and Edge Computing:** Leveraging faster 5G networks for real-time control and enhanced device connectivity.
* **Scalability:** Expanding from single-home automation to multi-home systems for smart communities.

### Conclusion

The Smart Home Automation System using Arduino presents a robust solution for modern living, combining convenience, energy efficiency, and enhanced security. By leveraging Arduino technology and the Internet of Things (IoT), this project showcases how everyday tasks can be automated, making home management simpler and more efficient.

The **conclusion** of your IoT Smart Home Automation project emphasizes the significant impact of integrating smart devices into home environments. By automating daily tasks such as controlling lights, temperature, and security systems, the project enhances convenience, security, and energy efficiency. The system provides users with centralized control through a mobile application, offering real-time monitoring and remote access. As IoT technology advances, this project serves as a foundation for future smart home solutions, paving the way for more intelligent, connected, and sustainable living environments.

**Final Thoughts:**

The IoT Smart Home Automation project represents a significant step toward creating a more connected and efficient living environment. By leveraging modern IoT technologies, this project not only simplifies everyday tasks but also enhances security and energy management. The user-friendly interface allows homeowners to monitor and control their devices remotely, providing peace of mind and convenience.

As the demand for smart home solutions continues to grow, this project can serve as a basis for further innovation and development. Future enhancements could include integrating more advanced sensors, improving data analytics for personalized experiences, and enhancing interoperability between different devices and platforms. Overall, the IoT Smart Home Automation project showcases the potential of technology to transform our homes into smarter, safer, and more efficient spaces.

**References**

* RemoteXY Documentation: RemoteXY Website
* Arduino Libraries: Arduino Website
* IoT-Based Smart Home Automation Video: YouTube
* Sensors and Components Datasheets: (Provide specific datasheets for DHT11, MQ135, PIR sensor, etc.)
* Blynk and MIT App Inventor documentation for mobile application development.

**Biblography**

* Hackster.io. (2023). How to Build a Smart Home with IoT.
* Instructables. (2023). Smart Home Automation Using IoT.