A Project Report

on

IOT Based Security System with Unpredictable Pattern

By

Rupesh Thakur – 4420

Under the esteemed guidance of

Ms. NEENU JOHNSON

(Assistant Professor)

Submitted in partial fulfilment of the Requirements for the award of the Degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

SEMESTER V EXAMINATION

DEPARTMENT OF INFORMATION TECHNOLOGY

THAKUR COLLEGE OF SCIENCE AND COMMERCE

(Permanently Affiliated to University of Mumbai)

KANDIVALI (E) -400101, MUMBAI, MAHARASHTRA

A.Y. 2024-25

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled, **"IoT Based Security System with Unpredictable Pattern"**, undertaken at the Thakur College of Science and Commerce by **RUPESH THAKUR** Roll. No: **(4420)** is submitted in partial fulfilment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY SEM V Examination and does not form part of any other course undergone by the candidate.
It is further certified that he/she have completed all the required phases of the project.

Project Guide **HOD**

External Examiner **Internal Examiner**

College Seal

ACKNOWLEDGEMENT

We would like to express our sincere thanks to the notable individuals whose constant assistance, direction, and knowledge have been crucial to the project's success.

We would like to begin by expressing our deepest gratitude to **Dr. Santosh Singh**, Head of Department for his unwavering support and imaginative leadership. Your crucial support and guidance were essential to the success of this project. Your confidence in our skills and dedication to creating an innovative atmosphere have been significant in enabling us to step into this innovative project.

Furthermore, we would like to express our profound gratitude to our esteemed Principal, **Dr. Mrs. Chaitali T Chakraborty**, for her continuous encouragement and support. Your inspirational guidance and strong dedication to success have been an inspiration to all of us.

We also extend our sincere thanks to **Ms. Neenu Johnson,** our guide. Your continuous support and expert advice were key to our project's success. Your capacity to push and motivate us has been essential to setting our goals and refining our strategy. We sincerely appreciate all of your helpful advice and inspiration.

We also like to thank the distinguished faculty members of **Thakur College of Science and Commerce** and the Department of IT. Your knowledge, direction, and mentoring have been extremely helpful in helping us refine our capabilities. Moreover**,** we are also grateful to our families and friends for their unwavering support during this journey.

We acknowledge and value the sacrifices you have made, including the innumerable hours you spent enabling and encouraging us to follow our goals and ensure the success of our project. Your confidence in us has always inspired us, and we are extremely thankful for your affection, compassion, and constant support.

Lastly, we acknowledge the **online resources** that offered vital information and inspiration, and we extend special thanks to the **local farmers** who generously shared their valuable insights with us.

INDEX

1. INTRODUCTION

In today's digital world, the Internet of Things (IoT) has transformed many areas of our lives, including how we approach security and surveillance. Traditional security systems often use separate components that don't communicate with each other in real time, thereby limiting their effectiveness. However, by integrating IoT technologies, security systems can now offer better monitoring, instant data updates, and automated responses, making them more reliable and efficient.

This project presents an IoT-based security and surveillance system using the ESP32 microcontroller as the main brain for handling data and communication between the sensor and cloud (MQTT).

The system includes several sensors:

- The MQ2 sensor to detect flammable gases.
- The MQ3 sensor for LPG leak detection.
- The DHT22 sensor to monitor humidity and temperature.
- The R307 finger-print sensor for Authentication.
- A reed switch to check if a door is open or closed.

Additionally, the ESP32-CAM module is used for live video surveillance, allowing users to visually confirm security events as they happen. Data visualization and control of the system are managed through Node-RED, a tool that helps create easy-to-use dashboards.

To enhance security, a 12V relay and solenoid are used for automated access control, and RFID technology ensures that only authorized individuals can enter. Communication between all these devices is managed by an MQTT broker, which ensures that data is transmitted efficiently and reliably.

By combining these technologies, the system provides a comprehensive security solution that enables real-time monitoring, alerts, and automated responses. This approach not only strengthens security but also helps to advance smart home and industrial safety systems.

1.1: OBJECTIVE & SCOPE OF THE PROJECT

SCOPE

The scope of this Project encompasses the comprehensive design, implementation, and evaluation of an IoT-based security and surveillance system intended for versatile application in residential, commercial, and small-scale industrial settings. The system is designed to seamlessly integrate multiple sensors and devices to monitor environmental parameters and physical access points, ultimately providing a holistic security solution. Key components and functionalities include:

- **Sensor Integration:** The system incorporates specific sensors tailored for different purposes, such as the MQ2 and MQ3 gas sensors for detecting combustible gases and LPG leaks respectively, the DHT22 sensor for monitoring ambient humidity and temperature, and a reed switch for door status detection. This comprehensive sensor integration enhances the system's ability to capture diverse environmental data and detect potential security threats effectively.
- **Real-Time Surveillance:** To facilitate visual monitoring of the premises, the system utilizes the ESP32-CAM module to capture and stream live video feeds in real time. This feature provides users with direct visual access to the secured areas for enhanced situational awareness and security management.
- **Data Processing and Communication:** The system deploys the ESP32 microcontroller to aggregate sensor data, process information, and establish seamless communication with cloud services or local networks. This ensures efficient handling of sensor data and facilitates smooth integration with external systems and services.
- **User Interface:** A user-friendly dashboard is created using Node-RED to enable real-time data visualization, system control, and alert management. The implementation of Node-RED enhances the ease of use and accessibility of the system, allowing users to interact with the data and control the system efficiently.
- **Automation and Alerts:** Automated responses based on sensor inputs are developed, including sending notifications or activating alarms in case of detected anomalies. This proactive approach enhances the system's ability to respond to potential security threats in real time, ensuring swift and appropriate actions are taken when required.
- **Access Control:** The system integrates a 12V relay and solenoid for automated door control, ensuring seamless operation. For secure access, it employs a multifactor authentication (MFA) mechanism, combining RFID technology and the **R307**

fingerprint sensor. The RFID module verifies the user's credentials, while the fingerprint sensor provides biometric authentication, ensuring that only authorized individuals gain entry. This layered security approach enhances overall access control, preventing unauthorized access and improving the reliability of the secured premises.

- **Data Transmission:** The system ensures efficient, reliable, and secure communication between all components using an MQTT broker. This allows for seamless and secure data transmission between different system components, enabling effective coordination and operation of the entire system.

OBJECTIVE

The primary objectives of this research are as follows:

- **Design and Development:** The goal is to design and develop an integrated IoT-based security and surveillance system using the ESP32 platform and specified sensors. The system aims to ensure seamless communication and data synchronization for efficient operation.
- **Real-Time Monitoring:** This objective involves the implementation of real-time data acquisition and video streaming capabilities. The focus is to provide continuous monitoring of environmental conditions and physical access points, enabling quick response to any security threats or irregularities.
- **Data Visualization and Management:** The aim is to develop an intuitive dashboard using Node-RED for real-time visualization of sensor data, system status, and event logs. This will allow for easy interpretation of the collected data for informed decision-making.
- **Alert Mechanism:** The objective is to establish an efficient alert system that promptly notifies users of detected security breaches or hazardous conditions through various communication channels such as SMS, email, and mobile notifications. The aim is to ensure swift and effective response to potential security issues.
- **Automated Access Control:** This objective involves implementing a reliable access control system using 12V relay, solenoid, and RFID technology. The system aims to ensure secure and automated management of entry points, enhancing overall security measures.
- **System Evaluation:** The goal is to assess the performance, reliability, and scalability of the proposed system through comprehensive testing under different scenarios and

conditions. This will enable the identification of any potential weaknesses and the refinement of the system for optimal performance.

- **Enhancement of Security Infrastructure:** The objective is to contribute to the advancement of smart security solutions by integrating multiple sensor modalities and leveraging IoT technologies for enhanced protection and surveillance. This includes the inclusion of advanced technologies for a more robust security infrastructure.

1.2: THEORETICAL BACKGROUND

IoT, or the Internet of Things, encompasses a network of interconnected devices that communicate and exchange data over the internet. These devices are often equipped with sensors, software, and other technologies to collect and share information, enabling them to perform automated tasks and provide valuable insights.

1. Challenges and Considerations

Security Concerns:

- Data Privacy: IoT devices must safeguard sensitive information from unauthorized access and breaches.
- Vulnerability to Attacks: IoT devices are susceptible to cyber-attacks, including hacking, denial-of-service (DoS) attacks, and spoofing.
- Data Integrity: Ensuring the accuracy and reliability of the collected and transmitted data.

Technical Issues:

- Interoperability: Integrating devices from different manufacturers can be challenging due to varying standards and protocols.
- Scalability: Managing and scaling IoT systems can become complex as the number of devices and volume of data increases.
- Latency: Real-time responses are critical for security applications, and network latency can impact performance.

Regulatory and Ethical Issues:

- Compliance: Adhering to regulations and standards related to data protection and privacy (e.g., GDPR, CCPA).
- Ethical Use: Balancing surveillance needs with privacy concerns and ensuring responsible use of monitoring technologies.

2. Future Directions

Advancements:

- AI and Machine Learning: Utilizing AI to enhance IoT-based security systems for improved threat detection, predictive analytics, and automated decision-making.
- Edge Computing: Performing more data processing locally to reduce latency and bandwidth usage.

- Blockchain: Utilizing blockchain technology to enhance security, data integrity, and access control in IoT systems.

3. Applications:
    - Video Surveillance: IoT-enabled cameras with capabilities such as motion detection, facial recognition, and license plate recognition.
    - Intrusion Detection: Implementing sensors and alarms to detect unauthorized access or movements.
    - Environmental Monitoring: Using sensors to detect changes in environmental conditions that may indicate security threats (e.g., smoke detectors, gas leak sensors).
    - Access Control: Deploying systems for managing entry and exit through biometric sensors, smart locks, and other authentication methods.

1.3: PROBLEM DEFINATION

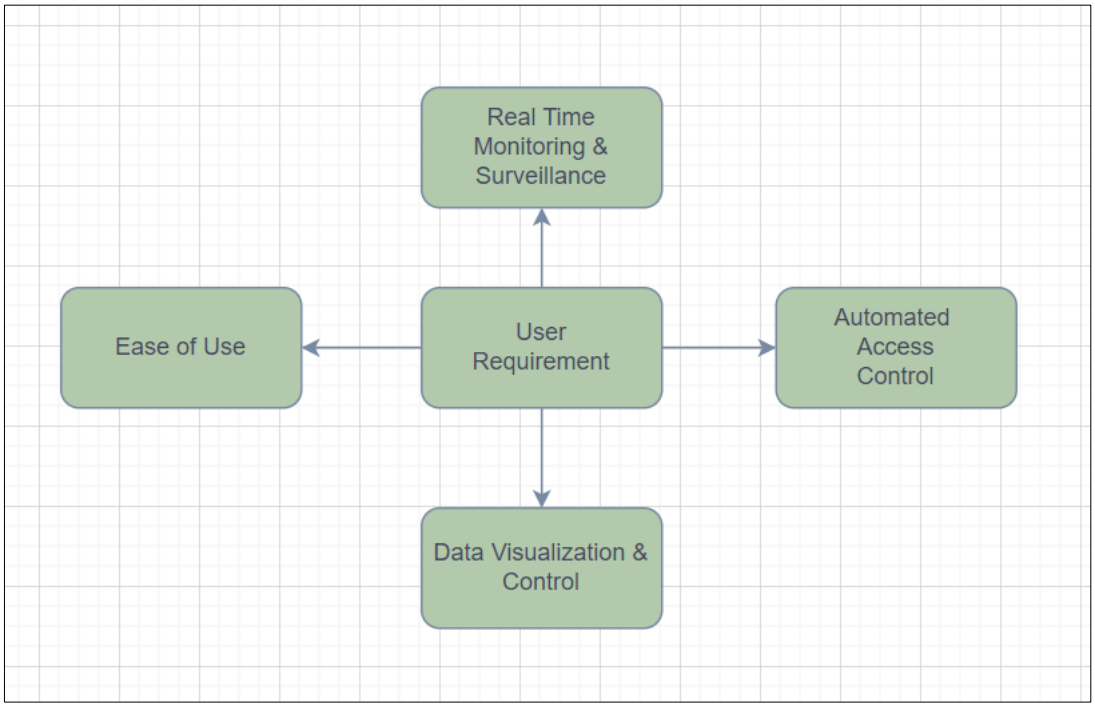
In the sphere of security and surveillance, traditional systems often face significant limitations, including delayed response times, lack of real-time monitoring, and insufficient integration of environmental sensing capabilities. These deficiencies can result in vulnerabilities related to the detection and response to security breaches, hazardous conditions, and unauthorized access. Specifically, the challenges addressed in this research include:

- **Lack of Real-Time Monitoring:** Conventional security systems may not provide instant data on security events or environmental changes, leading to delayed responses to potential threats. This delay can impact the system's effectiveness in identifying and mitigating security risks promptly.
- **Fragmented Sensor Integration:** Existing systems often rely on isolated sensors that do not communicate effectively, resulting in inefficiencies in data processing and threat detection. This fragmented approach can hinder the system's ability to provide a comprehensive and cohesive representation of the security environment.
- **Inadequate Environmental Hazard Detection:** Many security systems focus solely on physical access without monitoring environmental parameters, such as gas leaks, which pose significant safety risks. The lack of comprehensive environmental monitoring can expose occupants to potentially harmful conditions.
- **Limited Data Visualization and User Interaction:** The absence of user-friendly interfaces limits users' ability to monitor system status, analyse data trends, and manage alerts effectively. This can hinder proactive decision-making and timely responses to security incidents.
- **Scalability Issues:** Traditional systems may not easily adapt to varying security needs and expanding infrastructures. This limitation can restrict the system's ability to accommodate changes in security requirements and technological advancements.
- **High Implementation Costs:** The integration of multiple security and environmental sensors can be cost-prohibitive, limiting accessibility for residential and small-scale industrial applications. This financial barrier can impede the widespread adoption of comprehensive security solutions.

Addressing these challenges necessitates the development of an integrated, real-time IoT-based security and surveillance system that leverages advanced microcontrollers and sensors. By utilizing the ESP32 platform alongside MQ2 and MQ3 gas sensors, DHT22 for environmental monitoring, and ESP32-CAM for video surveillance, this research aims to

create a cohesive system that offers comprehensive security coverage. The incorporation of Node-RED for dashboard management further enhances user interaction and data visualization, ensuring timely and informed decision-making in response to security events and environmental hazards.

1.4: USER REQUIREMENTS



1.1: User Requirement Overview

ESP32 Microcontroller:

- **Firmware Development:** Develop and optimize firmware for the ESP32 to handle real-time data collection, processing, and seamless communication with other system components. Consider implementing modular and scalable code architecture to support future updates and additions.
- **Connectivity:** Implement and fine-tune Wi-Fi and Bluetooth capabilities to ensure robust and secure device communication and control. Evaluate power-saving modes and reliable reconnection mechanisms for uninterrupted operation.

Sensor Integration:

- **MQ2 and MQ3 Sensors:** Develop comprehensive code to read, calibrate, and interpret data from the MQ2 and MQ3 gas sensors. Consider implementing adaptive calibration algorithms to account for environmental changes and ensure accurate readings.
- **DHT22 Sensor:** Implement efficient data acquisition methods for precise temperature and humidity readings. Consider integrating error-handling protocols to maintain data accuracy in varying environmental conditions.
- **Reed Switch:** Create robust code to detect and report door status changes in real time. Consider implementing debouncing techniques to filter out noise and ensure accurate door state detection.

ESP32-CAM Module:

- **Video Streaming:** Implement optimized functionality for capturing and streaming live video, prioritizing low latency and high-resolution video transmission. Consider integrating video compression algorithms for data efficiency without compromising quality.
- **Image Processing:** Develop advanced features for image processing, such as motion detection, snapshot capture, and object recognition to enhance the system's surveillance capabilities.

Node-RED Dashboard:

- **Dashboard Design:** Utilize Node-RED to create an intuitive and visually appealing user interface for real-time data visualization and system control. Focus on responsive design and customization options to meet diverse user requirements.
- **Data Flow Management:** Configure robust data flows between sensors, actuators, and the dashboard, ensuring seamless data integration and real-time updates. Consider implementing data validation and error handling mechanisms to maintain data integrity.

Automated Access Control:

- **Relay and Solenoid Control:** Implement robust software for precise control of the 12V relay and solenoid, prioritizing reliability and safety in automated access management scenarios.
- **RFID Integration:** Develop secure and efficient functionality to read and process RFID tags for seamless access control. Prioritize encryption and authentication protocols to ensure secure entry authorization.
- **R307 Fingerprint Sensor Integration:** Develop a secure and efficient system to capture, store, and authenticate fingerprint data for seamless access control. Implement robust encryption and authentication protocols to ensure biometric data security, preventing unauthorized access and enhancing overall system reliability.

MQTT Broker:

- **Data Transmission:** Implement and optimize the MQTT protocol for efficient and reliable data transmission between the ESP32 microcontroller and the Node-RED dashboard. Focus on data prioritization and QoS configurations for varying data types.

- **Subscription Management:** Configure scalable and secure topics and message handling to ensure seamless data flow and minimize latency in the MQTT communication network.

Alert System:

- **Notification Configuration:** Develop versatile software to handle alert notifications via SMS, email, or mobile apps, ensuring timely and reliable alert delivery based on predefined triggers.
- **Threshold Settings:** Allow users to configure flexible thresholds for triggering alerts based on sensor data, prioritizing user customization and adaptability to varying environmental conditions.

System Testing and Validation:

- **Performance Testing:** Conduct rigorous performance testing, including stress testing and edge-case scenarios, to ensure the system meets performance and reliability standards under diverse operational conditions.
- **Usability Testing:** Evaluate the user interface and overall system usability through comprehensive user testing and feedback collection to ensure the system meets user needs and expectations effectively.

1.5: FEASIBILITY STUDY

Technical Feasibility

1. Component Compatibility:
    - ESP32 Microcontroller: due to integrated Wi-Fi and Bluetooth capabilities, high processing power, and versatility. –
    - Sensors: MQ2, MQ3, DHT22, and reed switch are reliable and provide accurate readings for gas detection, temperature, humidity, and door status.
    - ESP32-CAM: Supports real-time video streaming and image capture for live surveillance & easily available.
    - Node-RED: Open-source tool for building user-friendly dashboards and integrating data sources.
    - 12V Relay and Solenoid: easy interfacing with the ESP32.
    - M5 Stack RFID: Accuracy and cost effective.
    - R307 Fingerprint Sensor: Authentication and door control.

2. System Integration: Feasible with mature technologies, available libraries, and examples.
3. Data Management and Communication: MQTT protocol ensures efficient data transmission and real-time updates & open source.

Operations Feasibility

1. System Operations: Continuous environmental monitoring, gas leak detection, live video surveillance, and autonomous operations with automated alerts and responses.
2. User Interaction: Node-RED provides an intuitive web-based dashboard for real-time data viewing, access control, and alerts.
3. Maintenance, Support, and Scalability: Regular maintenance, user training, and support are necessary. The system is designed to be scalable.

Economic Feasibility

1. Cost Analysis: Initial costs are relatively low due to the affordability of IoT components. Operational costs primarily involve maintenance and occasional upgrades.

2. Return on Investment: The system provides enhanced security, reduced risk, and minimal operational costs, leading to a positive ROI.

Schedule Feasibility

1. Development Timeline:
    - Design and Planning: 2-3 weeks
    - Implementation: 1-3 weeks
    - Testing: 1 week
    - Deployment: 1-2 weeks

2. Milestones:
    - Prototype Development: 10 to 15 Days
    - System Integration: 4 Days
    - Testing and Evaluation: 5 Days
    - Final Deployment: 2 Days

1.6: DETAILS OF HARDWARE & SOFTWARE USED

Hardware Used:

1. LM2596 DC-DC Convertor: to convert the 12V to 5V for powering the Controller & Sensors.
2. ESP32 WROOM: Main brain of the system.
3. ESP32 CAM: for video streaming.
4. MQ2: for smoke detection.
5. MQ3: for LPG Leakage detection.
6. DHT22: for humidity & temperature sensing.
7. WS1850S RFID: for Access Control.
8. R307 Fingerprint Sensor: for Access Control.
9. 16 * 2 LCD Display: for displaying the network status, cloud status, time etc
10. I2C Driver for Display
11. 1 Channel Relay: for operating the 12V Solenoid & isolation.
12. 12V Solenoid
13. Reed Switch: for door status
14. Universal PCB Prototype Board
15. Wires
16. JST Connector
17. 2 Pin PTR Connector
18. Buzzer
19. Resistor
20. 12V Adaptor

Software Used:

1. MQTT Broker (test.mosquitto.org): mediator between the Node-RED & ESP32 for data exchange.
2. Node-RED: for designing the Dashboard.
3. MQTT Box: for testing the MQTT Client.
4. JavaScript: for implementing the custom script.

2: SYSTEM ANALYSIS & DESIGN

2.1: DETAILED LIFE CYCLE OF THE PROJECT

Description	Phases	Timeline	remark
Project Requirement Gathering	Controller Selection	2 Weeks	
	Sensor Selection (Gas Sensor, Survilleance, Status Monitoring)		
	Cloud Selection		
Procurement	Purchasing the ESP32, ESP32 CAM, MQ2 & MQ3 Sensor, DHT22, Reed Switch, 1 Channel Relay, 12 v Solenoid, RFID, 16 * 2 LCD Display & I2c Driver	2 Weeks	Some components are purchased online & some are easily available
Getting Familiar With the Technology Used by interfacing the sensor (Hardware + Software)	ESP32 Blinking Program	1 Day	Used Predefined Library
	Interfacing MQ2 & MQ2 sensor	1 Day	
	Interfacing DHT22 Sensor	2 Day	
	Interfacing Reed Switch & Relay	1 Day	
	interfacing 12 v Solenoid	1 Day	
	Interfacing RFID	4 Days	
	Interfacing 16 * 2 Display	2 Days	
	Interfacing ESP32 CAM	5 Days	
	Reading the Reed Switch Status	1 Day	
Node RED Dashboard Desgning	Real Time Survilleance	2 Week	
	Status		
Final Code Merging		1 Week	
Testing		1 Week	
Assembly	Component Soldering on Zero Board	1 Week	
	Packing in the Enclosure	3 Days	

2.1: Detailed life Cycle

1. **Requirement Gathering:**

   Gather and document specific requirements, including hardware components (ESP32, MQ2, MQ3, DHT22, ESP32-CAM, RFID,R307 etc.), software needs (Node-RED, MQTT), and expected system performance.

2. **Component Selection:**

   Finalize the selection of controller, sensors & Cloud Protocol based on the requirements. Ensure that each component is compatible with the ESP32 microcontroller and meets the project's technical needs.

3. **Development and Prototyping**

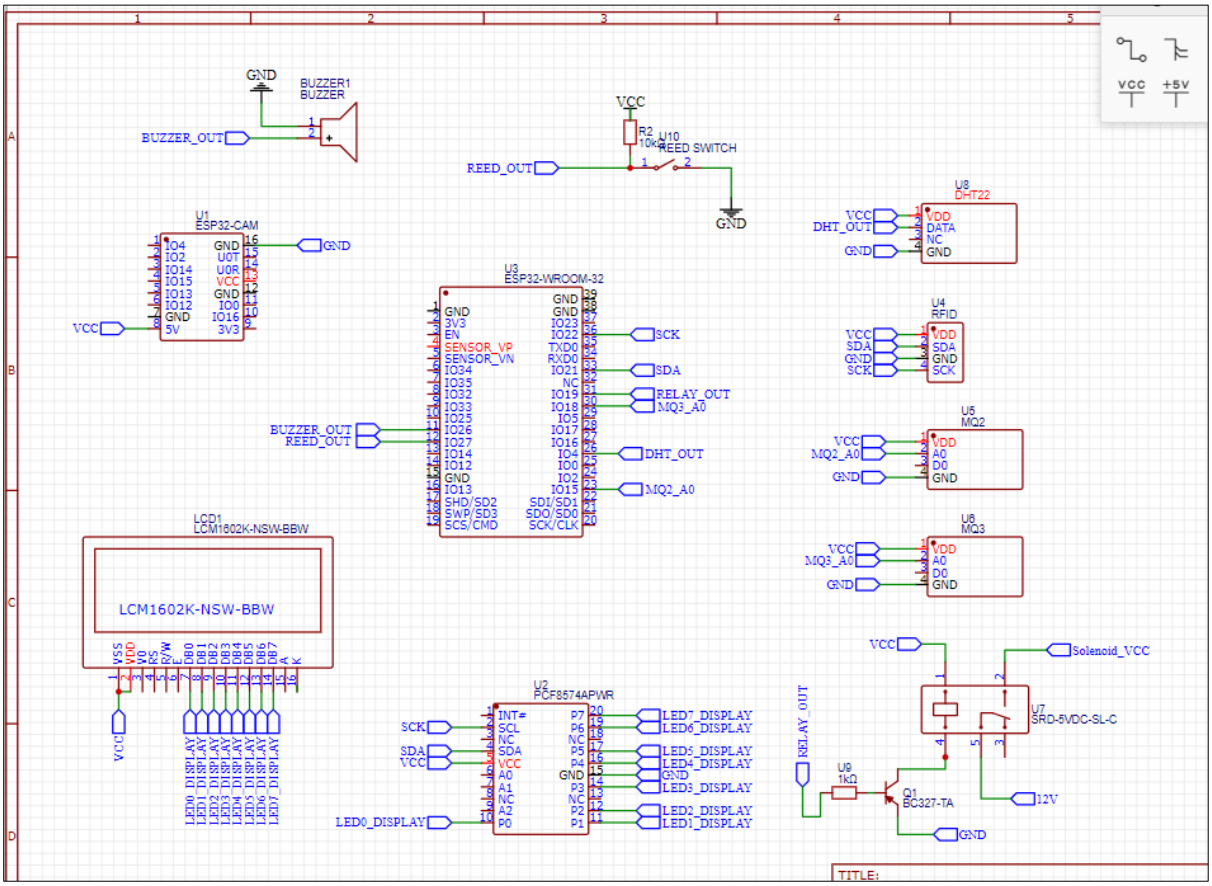
   - Hardware Setup: Assemble the hardware components, including wiring the MQ2 and MQ3 sensors, DHT22 sensor, ESP32-CAM, reed switch, 12V relay, solenoid, and RFID module with the ESP32 microcontroller.

- Firmware Development: Write and upload firmware to the ESP32 to handle sensor data collection, MQTT communication, and control operations.
- Node-RED Dashboard: Develop the Node-RED dashboard to display real-time sensor data, video feeds from the ESP32-CAM, and provide control interfaces for access control.
- Integration: Integrate all hardware and software components to ensure they work seamlessly together. This includes setting up the MQTT broker, configuring the ESP32 to publish/subscribe to relevant topics, and ensuring Node-RED correctly processes and displays data.

4. **Testing and Validation:**

   Test each sensor individually to ensure they are functioning correctly. Verify that the ESP32 is correctly reading sensor data and controlling actuators.

2.2: CIRCUIT DIAGRAM



2.2: Circuit Diagram

Pinout Details:

DESRIPTION	SENSOR OUT	ESP32 PIN
DHT22	OUT	GPIO 4
MQ2	A0	GPIO 15
MQ3	A0	GPIO 18
Reed Switch	REED_OUT	GPIO 27
Buzzer	POSITIVE TERMINAL	GPIO 26
Relay	IN1	GPIO 19
I2C Driver	SDA	GPIO 21
	SCL	GPIO 22
RFID	SDA	GPIO 21
	SCL	GPIO 22
R307	RX	GPIO 16
	TX	GPIO 17
VCC	VCC	VCC
GND	GND	GND

2.3: Interfacing Pin Details

2.3: COMPONENT LEVEL DESCRIPTION AND SPECIFICATION

1. ESP32 WROOM

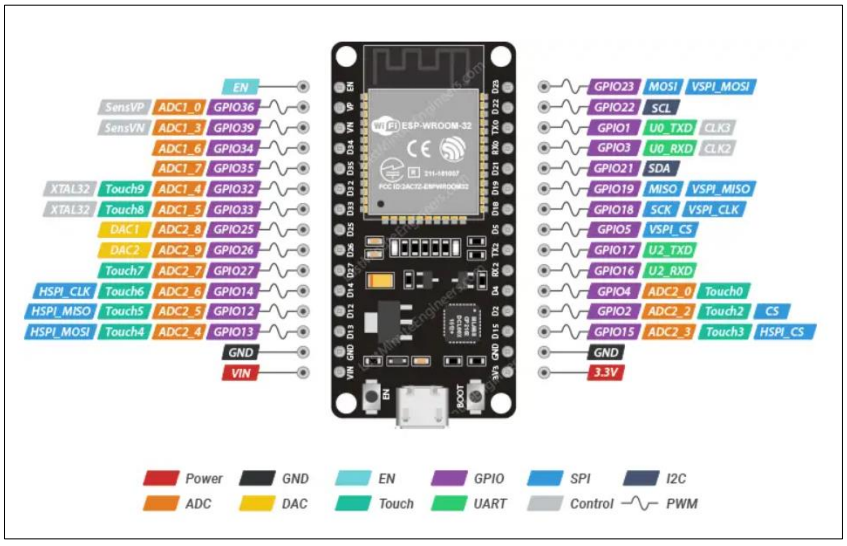
The ESP32 is a versatile microcontroller with integrated Wi-Fi and Bluetooth, making it ideal for IoT applications. It serves as the central processing unit for this security system, managing data from various sensors, controlling solenoid, and communicating with the Node-RED dashboard.

ESP32 Peripherals and I/O:

Although the ESP32 has 48 GPIO pins in total, only 25 of them are broken out to the pin headers on both sides of the development board. These pins can be assigned a variety of peripheral duties, including:

15 ADC channels	15 channels of 12-bit SAR ADC with selectable ranges of 0-1V, 0-1.4V, 0-2V, or 0-4V
2 UART interfaces	2 UART interfaces with flow control and IrDA support
25 PWM outputs	25 PWM pins to control things like motor speed or LED brightness
2 DAC channels	Two 8-bit DACs to generate true analog voltages
SPI, I2C and I2S interface	Three SPI and one I2C interfaces for connecting various sensors and peripherals, as well as two I2S interfaces for adding sound to your project
9 Touch Pads	9 GPIOs with capacitive touch sensing

Pinout:



2.4: Pinout of ESP32

Features:

- Dual-core Ten silica LX6 microprocessor.
- 2.4 GHz Wi-Fi and Bluetooth 4.2 BLE.
- Multiple GPIO pins for sensor and actuator interfacing.
- Low power consumption with multiple power modes.

2. LM2596 DC-DC Convertor:

DC-DC Buck Converter Step Down Module LM2596 Power Supply is a step-down(buck) switching regulator, capable of driving a 3-A load with excellent line and load regulation. These devices are available in fixed output voltages of 3.3 V, 5 V, 12 V, and an adjustable output version.

The LM2596 series operates at a switching frequency of 150kHz, thus allowing smaller sized filter components than what would be required with lower frequency switching regulators.

2.5: LM2596

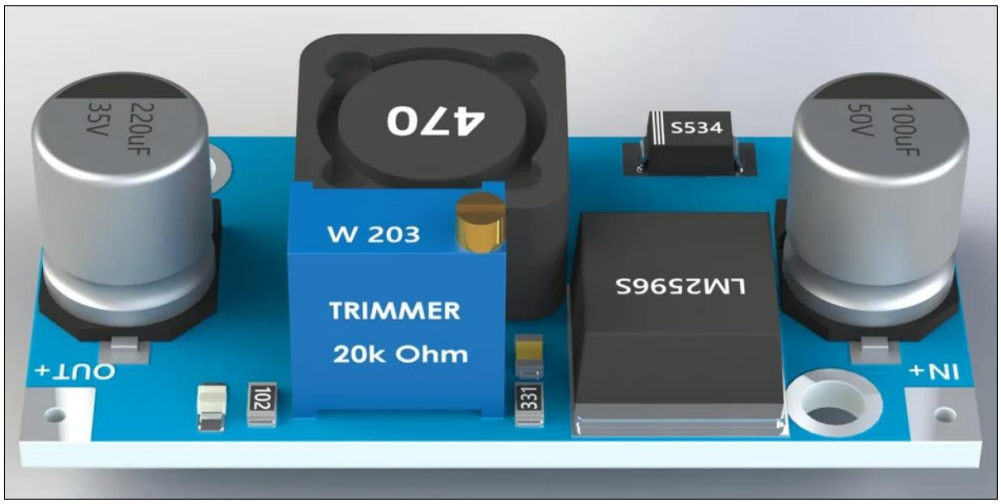
Pinout:

IN+ Here we connect the **red wire** from the battery (or the power source), this is VCC or VIN (4.5V - 40V)

IN- Here we connect the **black wire** from the battery (or the power source), this is ground, GND or V--

OUT+ Here we connect the positive voltage of the power distribution circuit or a component powered

OUT- Here we connect the ground of the power distribution circuit or a component powered



2.6: LM2596 Voltage Adjustment

This is a buck converter meaning that it will take higher voltage and convert it into lower voltage. To adjust the voltage, we have to do couple of steps.

1. Connect the converter with the battery or other power source. Know how much voltage you have inputted in the converter.
2. Set the multi-meter to read the voltage and connect the output of the converter to it. Now you can already see the voltage on the output.
3. Adjust the trimmer (here 20k Ohm) with a tiny screwdriver until the voltage is set to the desired output. Feel free to turn the trimmer in both directions to get the feeling how to work with it. Sometimes when you use the converter for the first time you will have to rotate the trimmer screw 5-10 full circles to get it working. Play with it until you get the feeling.
4. Now that the voltage is appropriately adjusted, instead of the multi-meter connect the device/module you want to power.

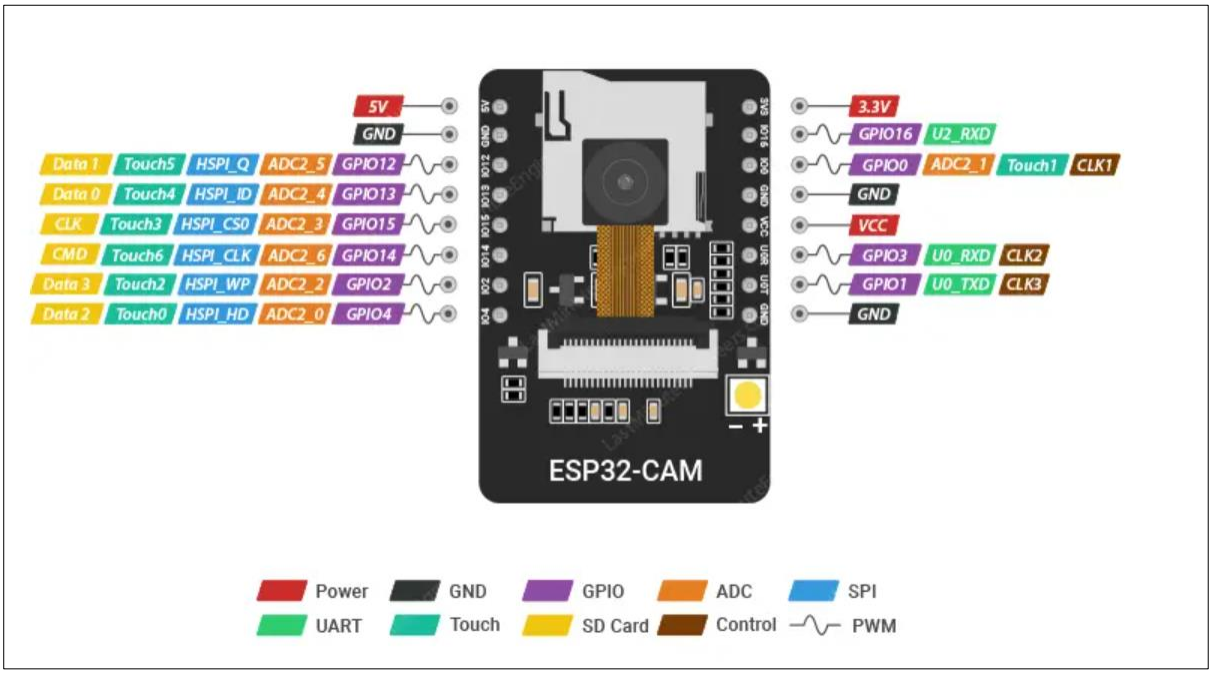
Specifications:

Output current	Rated current is 2A, maximum 3A(Additional heatsink is required)
Switching Frequency	150KHz
Operating temperature	Industrial grade (-40 to +85)
Conversion efficiency	92%(highest)
Load Regulation	± 0.5%
Voltage Regulation	± 0.5%
Dynamic Response speed	5% 200uS
Dimension	45*20*14mm(L*W*H)

3. ESP32 CAM

ESP32-CAM is a powerful device with built-in camera and Wi-Fi support which make it suitable for most IoT Project where Video Streaming is required. Unfortunately, the ESP32-CAM has fewer I/O pins, some of which are shared with the SD card and thus cannot be used when the card is present, making it difficult to design a project around it.

Pinout:



2.7: ESP32 CAM

Features:

- 2MP OV2640 camera module.
- Supports image and video capture.
- Wi-Fi connectivity for real-time streaming.
- Compact and low power consumption.

4. MQ2 Sensor

The MQ2 sensor is one of the most widely used in the MQ sensor series. It is a MOS (Metal Oxide Semiconductor) sensor. Metal oxide sensors are also known as Chemiresistors because sensing is based on the change in resistance of the sensing material when exposed to gasses.

The MQ2 gas sensor operates on 5V DC and consumes approximately 800mW. It can detect LPG, Smoke, Alcohol, Propane, Hydrogen, Methane and Carbon Monoxide concentrations ranging from 200 to 10000 ppm.

Pinout:

2.8: MQ2 Pinout

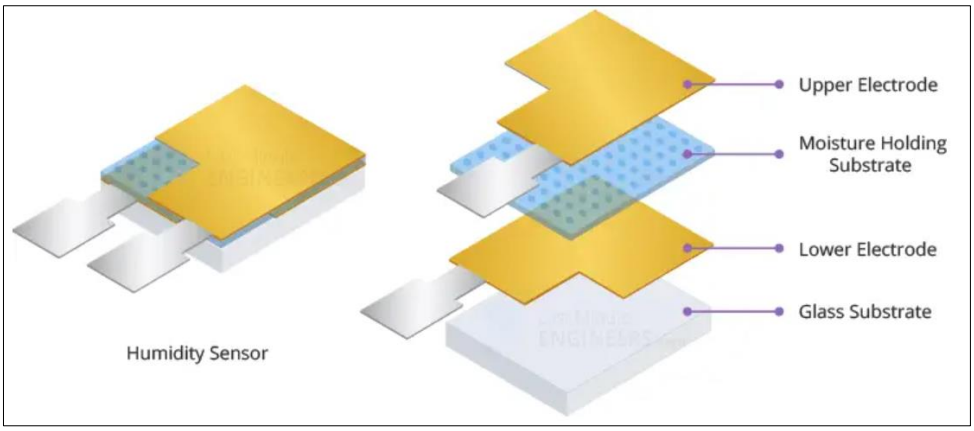
MQ2 Features:

- Detects LPG, i-butane, propane, methane, alcohol, hydrogen, and smoke.
- Sensitive to combustible gases and smoke.
- Analog and digital output.

5. DHT22

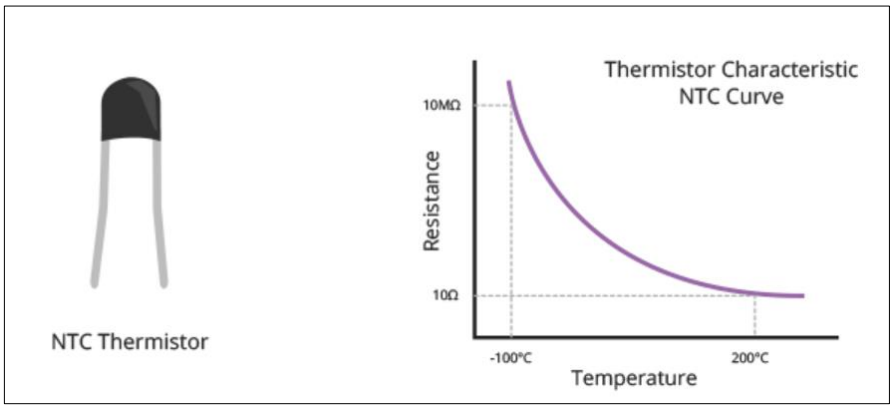
The DHT22 is a digital sensor that measures temperature and humidity. It is known for its accuracy and reliability in various environmental monitoring applications.

The humidity sensing component has two electrodes with a moisture-holding substrate (usually a salt or conductive plastic polymer) in between. As the humidity rises, the substrate absorbs water vapor, resulting in the release of ions and a decrease in the resistance between the two electrodes. This change in resistance is proportional to the humidity, which can be measured to estimate relative humidity.



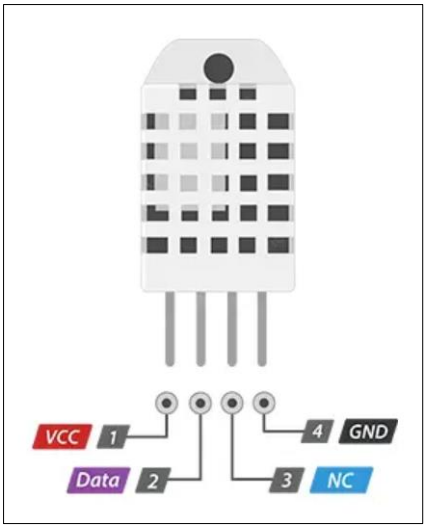
2.9: Internal Structure of Humidity Sensor

The sensor also includes a NTC thermistor for measuring temperature. A thermistor is a type of resistor whose resistance varies with temperature.



2.10: Temperature Map

Pinout:



2.11: DHT22 Pinout

Features:

- Measures temperature from -40 to 80°C with ±0.5°C accuracy.
- Measures humidity from 0 to 100% with ±2-5% accuracy.
- Digital output via a single-wire protocol.

6. WS1850S RFID

RFID operates in the 13.56MHz frequency band and uses the modulation and demodulation principle to interact with the proximity RF card. This unit can realize the function of the card reading and writing device, to identify and record multiple card information, to encode and authority a RF card.

It is suitable for the applications such as access control system, punching system, warehouse goods storage and community vehicle access registration.

I2C address is 0x28.

Pinout:

2.12: RFID Pinout

Features:

- 13.56MHz Operation frequency
- I2C data rate: Fast mode: up to 400 Kbit/s; High-speed mode: up to 3400 Kbit/s
- Transceiver Buffer: 64 bytes
- Supported protocol: ISO14443A, MIFARE and NTAG
- Operate temperature: -20℃-85℃
- How long data be saved for: > 10 years

- Reading and writing distance: < 20 mm
- Program Platform: Arduino, UIFlow (Blockly, Python)
- Two Lego installation holes

7. R307 fingerprint Sensor

The R307 fingerprint sensor is a biometric module designed for secure access control applications. It captures, processes, and stores fingerprint data while ensuring high recognition accuracy and security. The module integrates an optical fingerprint reader and onboard processing capabilities to match fingerprints with stored templates, enabling reliable authentication. It is widely used in access control systems, attendance tracking, and security-sensitive applications. The R307 sensor operates with a serial communication interface (UART) for seamless integration with microcontrollers like Arduino and ESP32.

Pinout:

2.13: R307 Pinout

Features:

- 13.56MHz Operation frequency
- Optical Fingerprint Recognition for secure and precise authentication
- Image Resolution: 500 DPI
- Storage Capacity: Up to 1000 fingerprint templates
- Communication Interface: UART (Baud rate: 9600–115200 bps)
- Verification Speed: < 1 second
- False Acceptance Rate (FAR): < 0.001%
- False Rejection Rate (FRR): < 0.1%

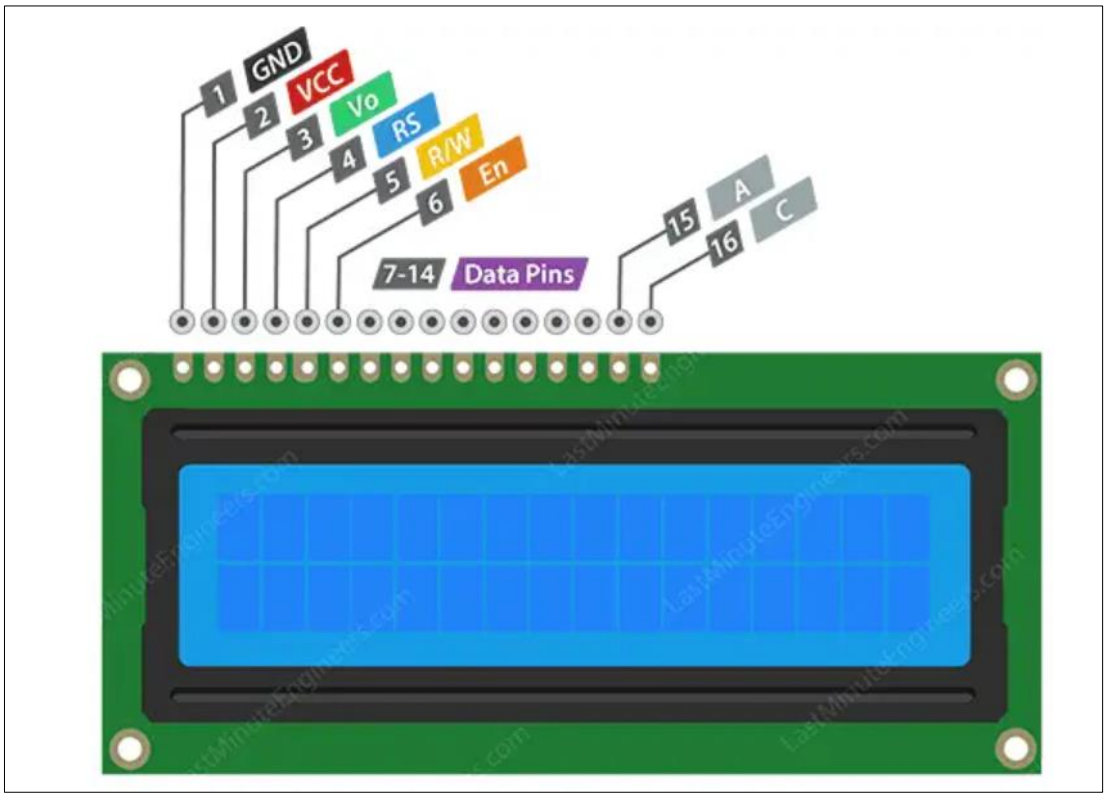
- Operating Voltage: 3.6V–6V
- Operating Temperature: -20℃ to 60℃

8. 16 * 2 LCD Display

LCD, or Liquid Crystal Display, is a type of display that uses liquid crystals to show characters. When activated by an electric current, these liquid crystals become opaque, blocking the backlight that is located behind the screen. As a result, that area will be darker than the rest. By activating the liquid crystal layer in specific pixels, characters can be generated.

these LCDs are ideal for displaying only characters. A 16×2-character LCD, for example, can display 32 ASCII characters across two rows.

Pinout:



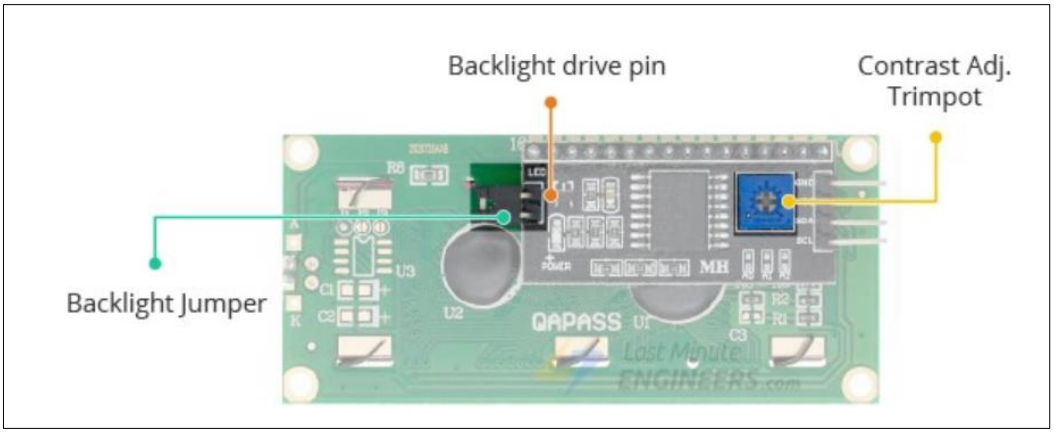
2.13: 16*2 LCD Display

Specifications:

Characters	16
Character Color	White
Backlight	Blue
Input Voltage (V)	5
Length (mm):	80
Width (mm):	36
Height (mm):	14.5

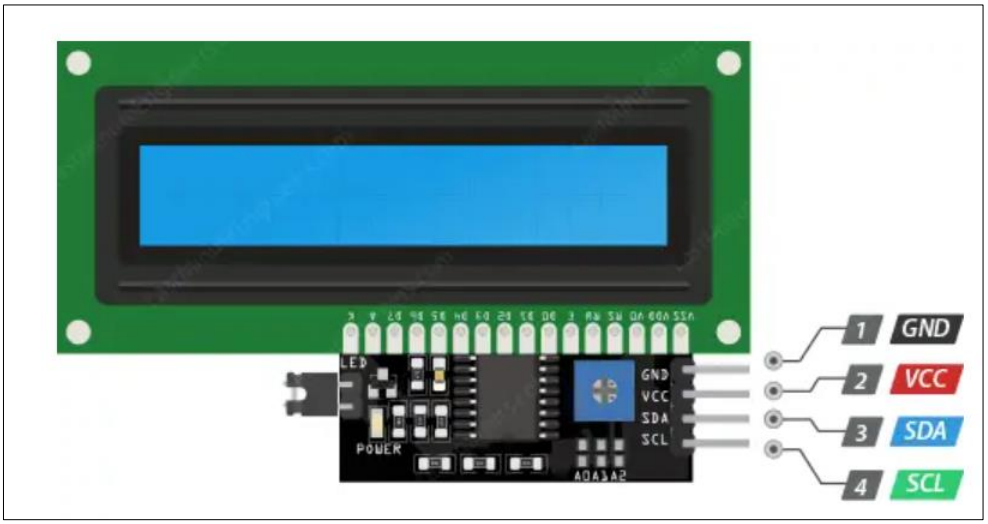
9. I2C Driver

The adapter is an 8-bit I/O expander chip – PCF8574. This chip converts the I2C data from an Arduino into the parallel data required for an LCD display. The board also includes a tiny trim pot for making precise adjustments to the display's contrast.



2.14: I2C Driver Setting

Pinout:

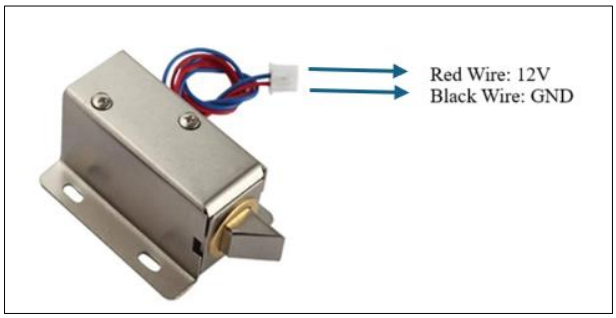


2.15: I2C Driver Pinout

10. 12V Solenoid

The solenoid door lock is an electromechanical locking device that controls access to secured areas. It is activated by the ESP32 based on user authentication.

Pinout:



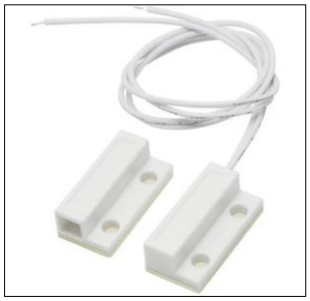
2.16: 12V Solenoid Pinout

Features:

- Iron Body Material
- Rustproof, durable, safe, and convenient to use.
- Suction which tightly stuck the iron, thus locking the door.
- Applicable for being installed in the escape door or fire door electronic controlled system.
- Adopts the principle of magnetism, when the current is through the silicon, the electromagnetic lock will achieve a strong.

11. Reed Switch

In-home alarm systems you need many sensors like **proximity sensors**, Heat detection sensors, and Magnetic door sensors to determine the safety status of the home. In these sensors door sensors are important as these determine whether the gate is open or closed based on that we can detect the intrusion.



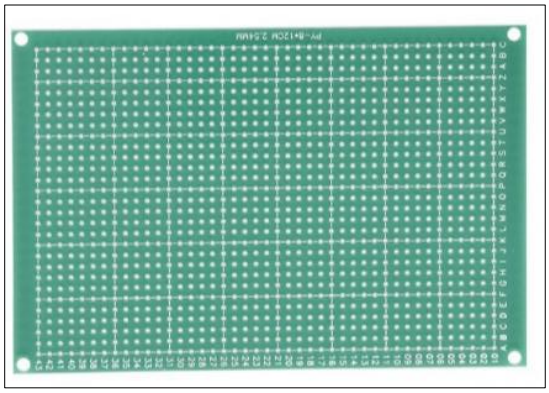
2.17: Reed Switch

Features:

1. Easy to install
2. Strong concealment
3. The magnetic sensor alarm, controls the switch of the circuit via the built-in magnet
4. Alarm when someone intrudes into your places
5. Can be used in places like apartments, hotels, offices, etc.
6. Ideal for residential or commercial use.
7. Designed to do embedded in the door or window frame.

12. Universal PCB Prototype Board

It is used for assembling the component for final look. Universal PCB Prototype Board Single-Side High-quality universal prototyping board with standard 2.5mm (0.1 inches) spacing through holes.



2.18: Universal PCB Board

Features:

1. High quality of base material
2. It has a thickness of the Solder resist layer.
3. copper clad made tolerant as per international Standards

13. Buzzer

It is great to add Audio Alert to your electronic designs. It operates on a 5V supply, uses a coil element to generate an audible tone.

Pinout:

2.19: Buzzer Pinout

14. 12V Adaptor

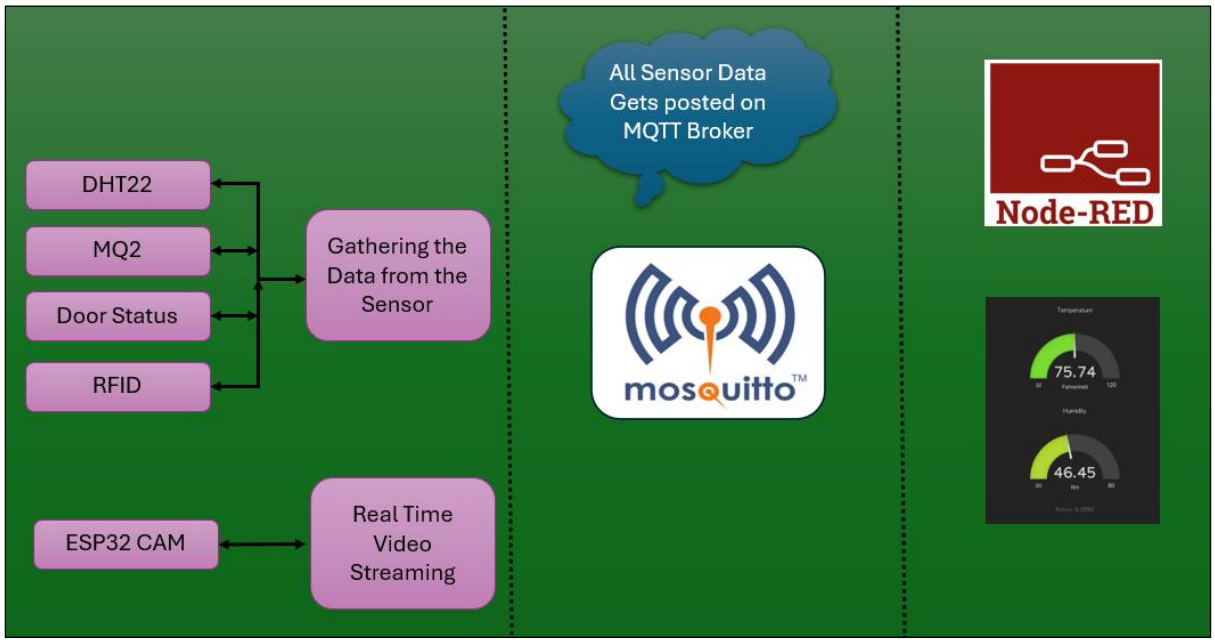
The SMPS Power Adaptor – 12V/2A (Power supply) is a Switched mode power supply (SMPS). This is the advanced power supply and better than conventional ones. If you want low losses and stable output also you don't want to make a rectifier, so go for SMPS.

2.20: 12V Adaptor

Features:

1. PWM design, make sure the stability and high efficiency of power supply.
2. Anti-jamming, passed EMC test, wave less than 20MV.
3. Efficiency above 80%, exotherm less than 25 degree
4. Function: with overvoltage protection, short-circuit protection, overload protection
5. Constant voltage output, assume stable power supply for LED lightings to reach long lifetime and reduce led light decay.

2.4: ARCHITECTURE DESIGN



2.21: Architecture Diagram

As shown on the above architecture design, the design involves four basics step. They are as follows:

Step 1: Once the device gets bootup first it will initialize the required driver.

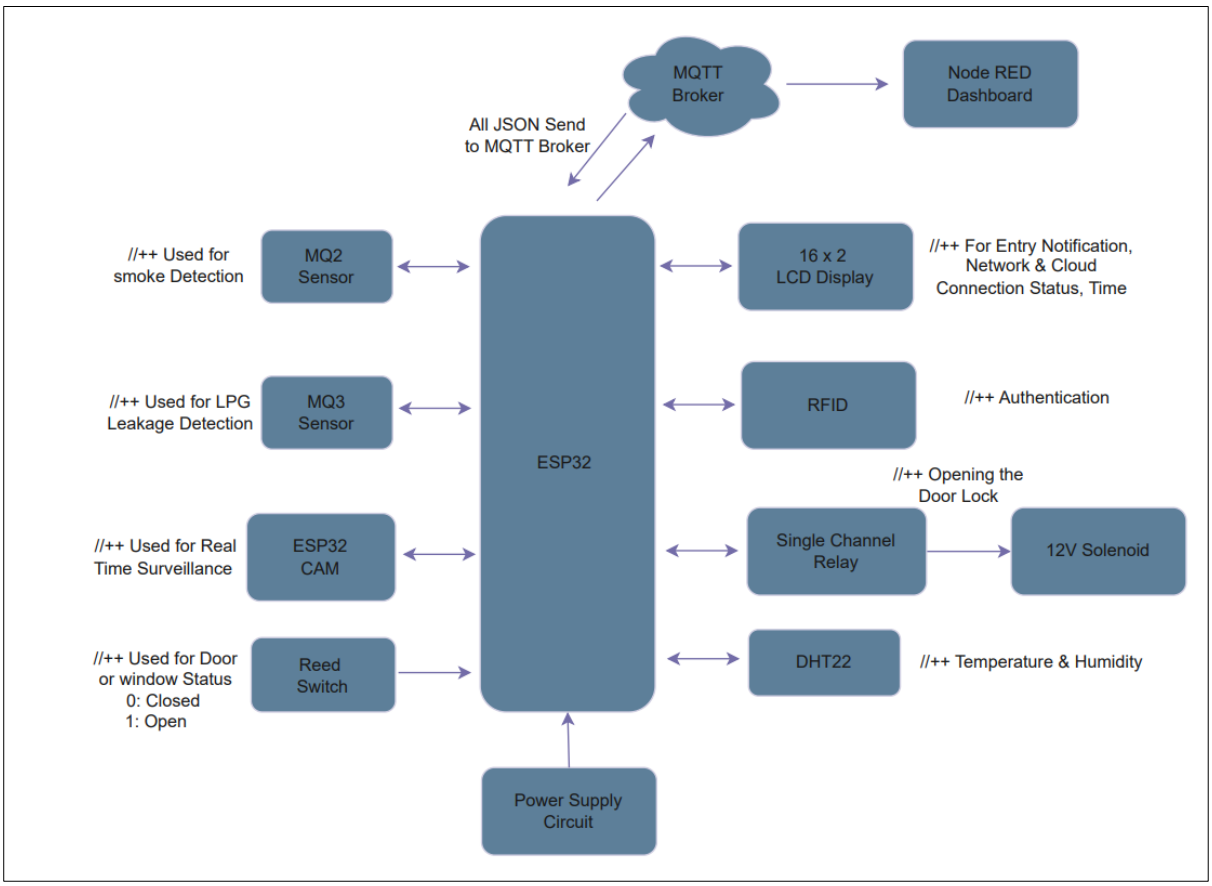
Step 2: After Driver initialization it will establish the connection with Wi-Fi & MQTT Broker (Mosquito)

Step 3: Now it will gather the data from the All the sensor like Temperature & Humidity (DHT22), Gas Sensor (MQ2), Door Status (Reed Switch), Access Control (RFID) & Live video streaming for surveillance through ESP32 Cam.

Step 4: ESP32 will send the data to MQTT Broker (TCP IP Protocol).

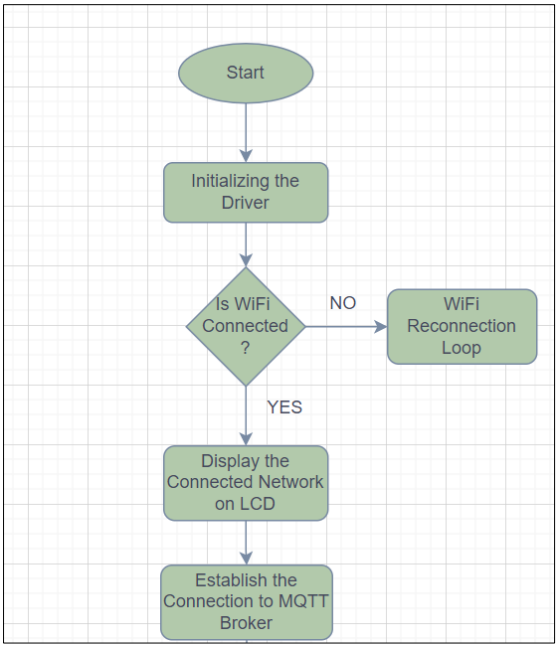
Step 5: All the data are received from MQTT Broker and map to the Node-RED server where dashboard is implemented. Now the user has the access of video streaming and Real time status.

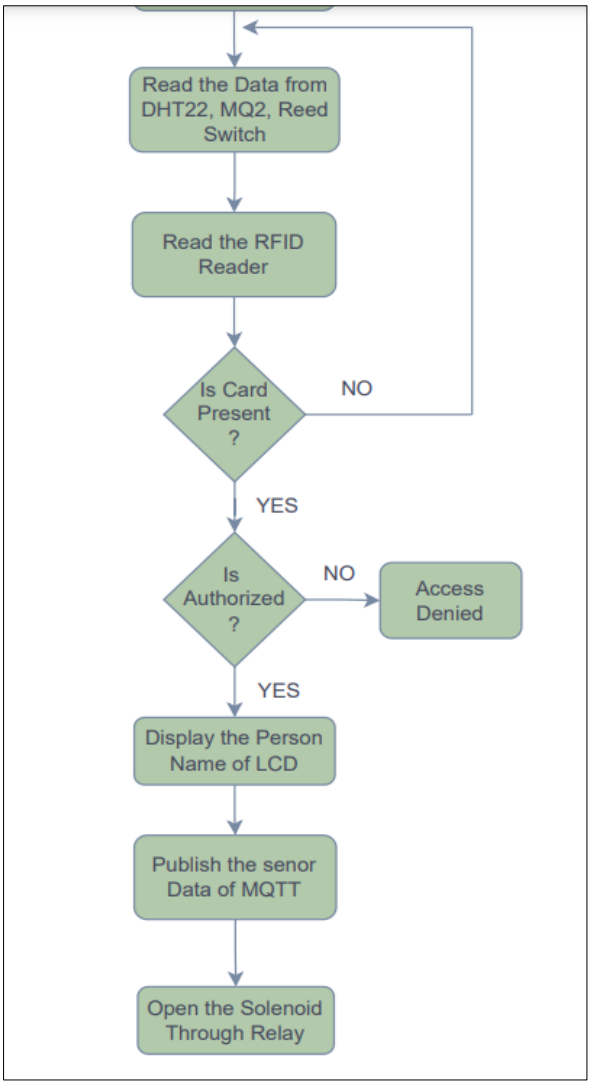
2.5: BLOCK DIAGRAM



2.22: Block Diagram

The Working of the Project has been explained through the below Flow Diagram

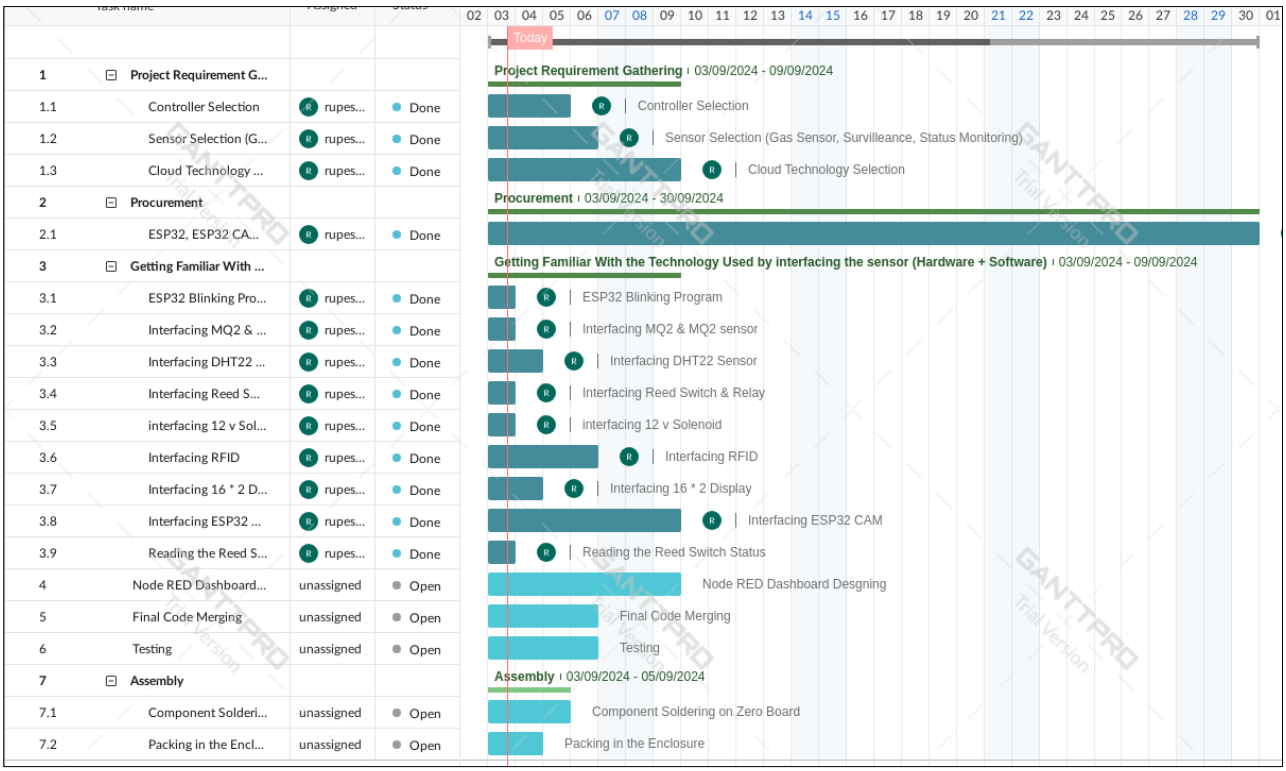




2.23: Working Flow

3: SYSTEM PLANNING

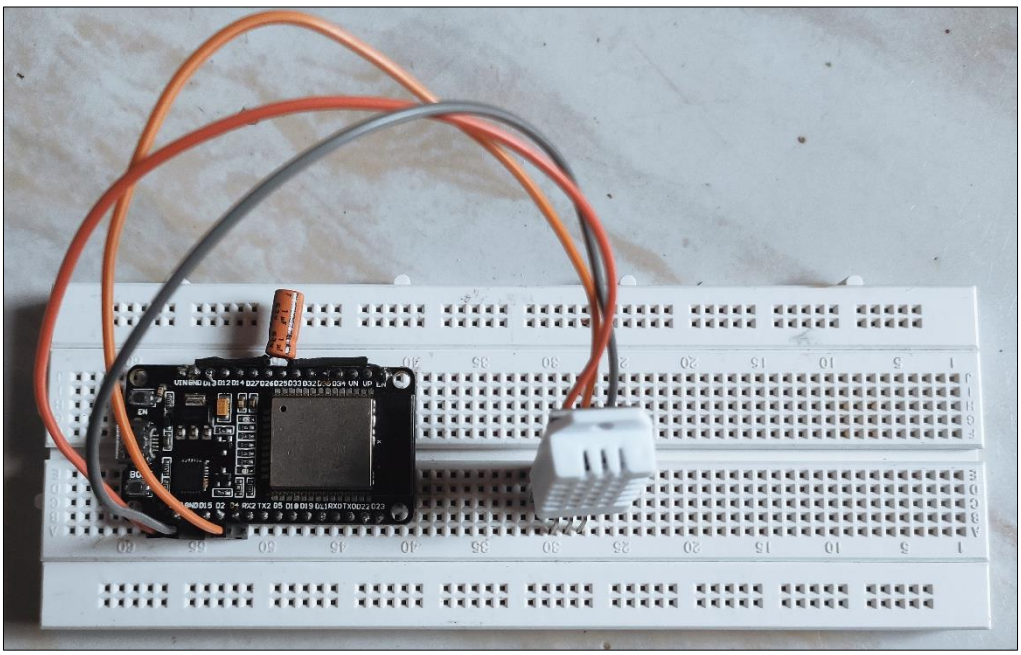
3.1: GANTT DIAGRAM



3.1: planning Gantt Chart

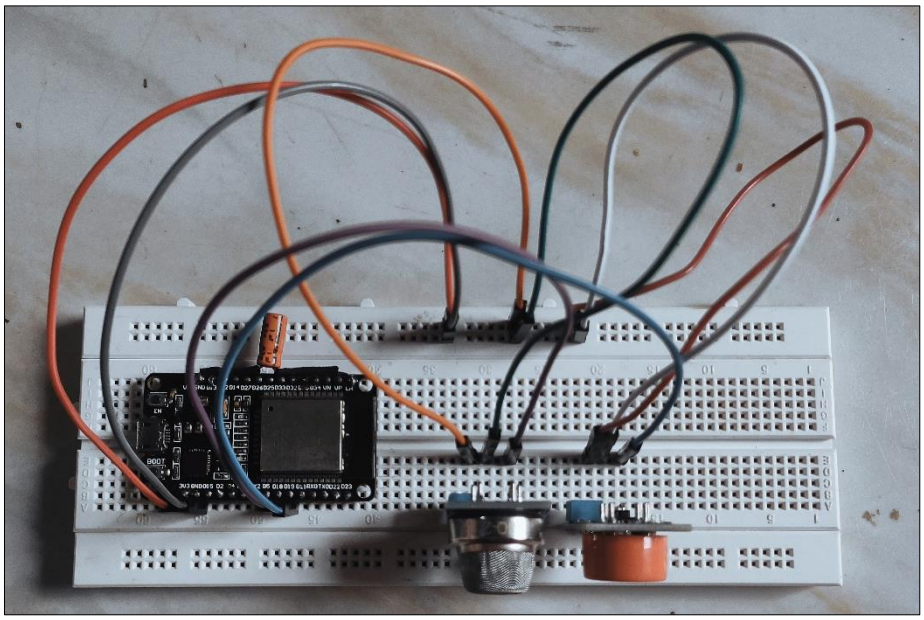
4: SYSTEM IMPLEMENTATION

Activity 1: Interfacing the DHT22 Sensor.



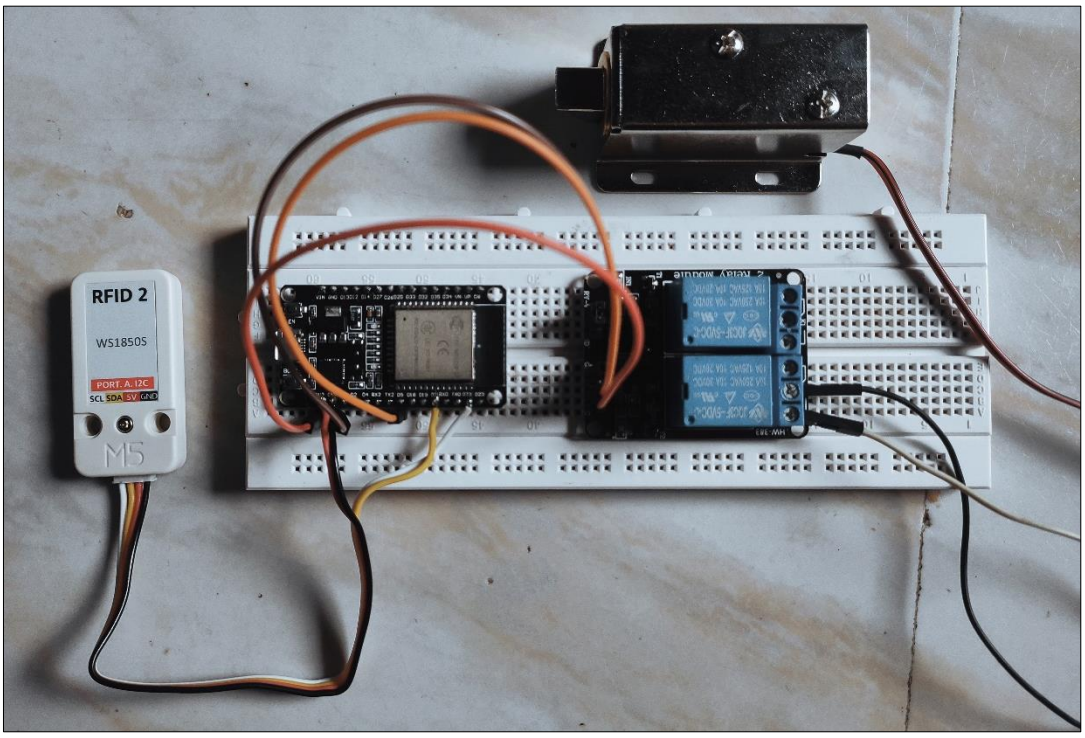
3.2: DHT22 Interfacing Diagram

Activity 2: Interfacing the MQ2 & MQ3 Sensor.



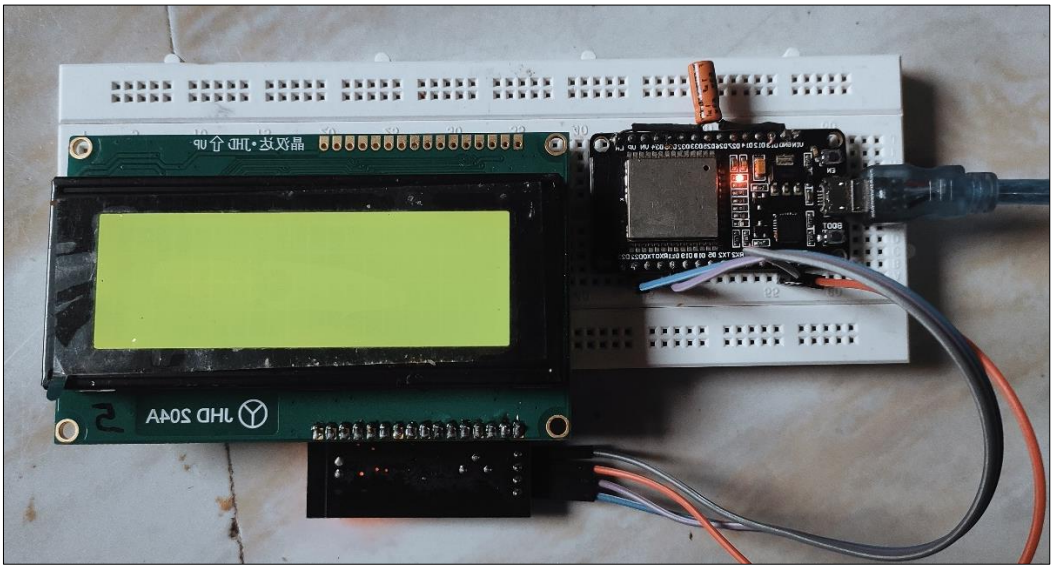
3.3: Gas Sensor Interfacing Diagram

Activity 3: Interfacing the RFID & Solenoid.



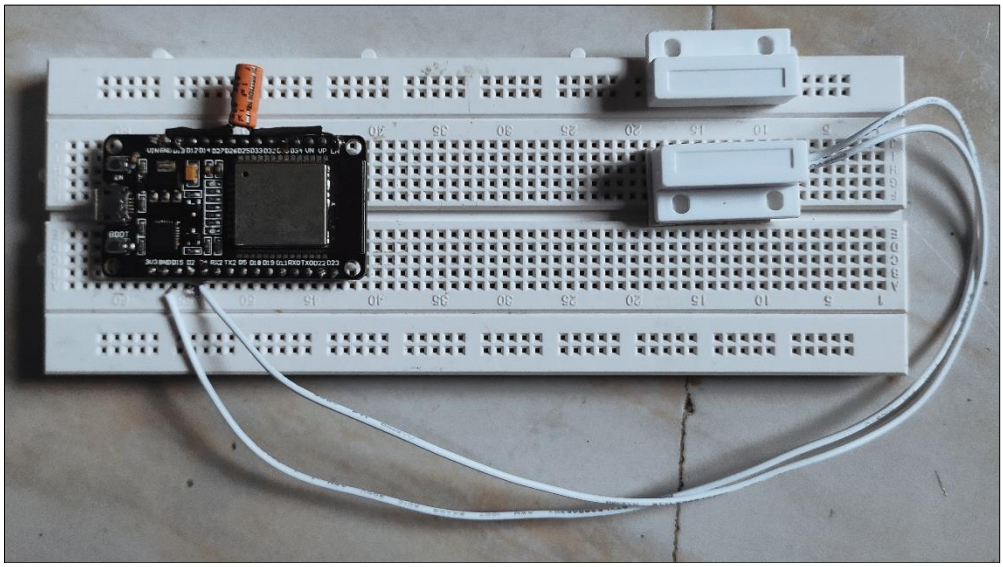
3.4: RFID & Solenoid Interfacing Diagram

Activity 4: Interfacing the 16*2 LCD Display.



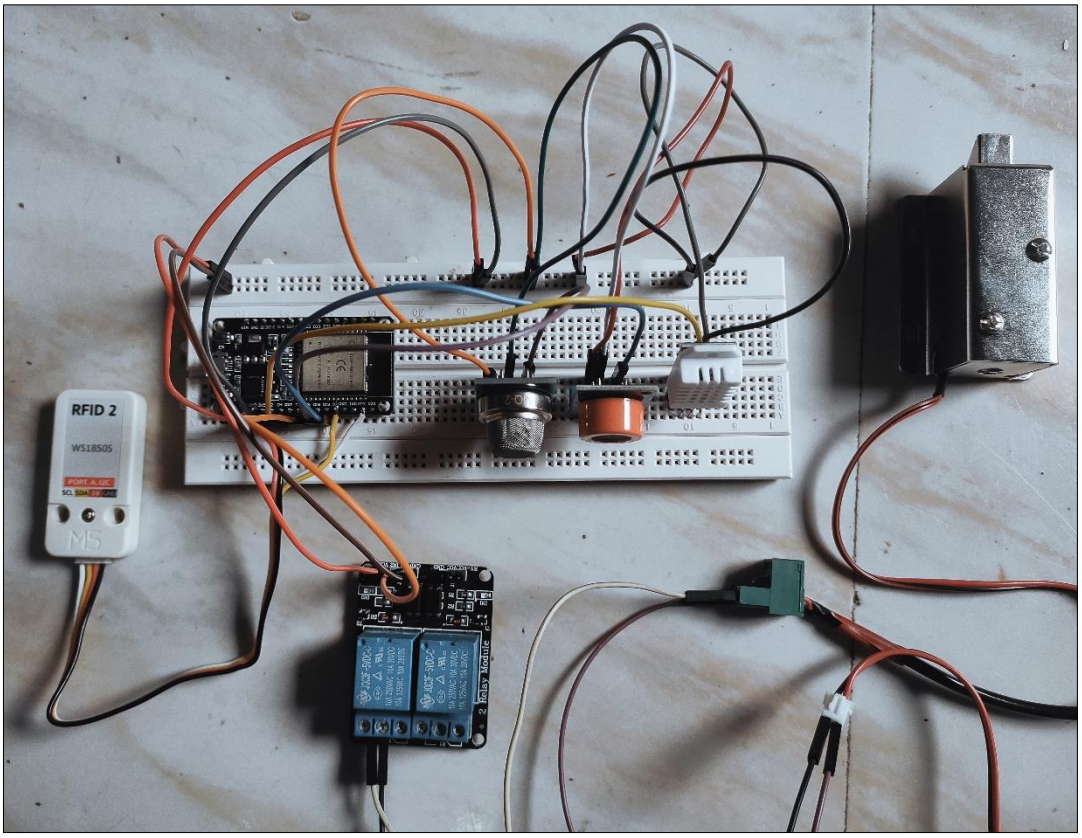
3.5: 16*2 LCD Display Interfacing Diagram Through I2C Driver

Activity 5: Interfacing the Reed Switch.



3.6: Reed Switch Interfacing Diagram

Activity 6: Final Testing with All the sensor.



3.7: Final Interfacing

5. COST BENEFIT ANALYSIS AND SOFTWARE PARAMETER ESTIMATION

Budget Report:

Component	Price
LM2596 DC-DC Convertor	50
ESP32 WROOM	450
ESP32 CAM	379
MQ2	115
DHT22	168
WS1850S RFID	450
16*2 LCD Display	87
I2C Driver	50
1 Channel Relay	79
12V Solenoid Lock	350
Reed Switch	43
Universal PCB Prototype Board	100
Buzzer	5
12V Adaptor	250
Resistor	10
Enclosure	500
Total	3086

4.1: Price of each components

As per our market study through the electronics product site such as Robu, evelta, amazon etc. with this much functionality the product almost cost above 10,000 RS to 25,000 RS.

Software Parameter Estimation:

- MQTT Broker
- AWS Server for hosting Node-RED

As our data is not big, we can use EC2 medium instance for hosting the Node-RED on AWS which nearly cost $0.126/day.

For MQTT Broker we can use free version of Hive MQ Broker.

Free Version Suport:

- Adding 100 Device
- Cloud storage 10GB

Effort Calculation (Man Days):

Step1: Deciding the Project Types

As Project contain both Hardware & Software, having the real time constraint so that the it falls under Embedded Project Type.

a = 3.6 (constant for embedded)

b = 1.20 (constant for embedded)

Step2: Generating the COCOMO Formula based on Project Type

Using the COCOMO Formula for the Embedded Category

Effort = a x $(KLOC)^b$

Substituting values:

Effort = 3.6 x $(KLOC)^{1.20}$

Step3: Calculating the Number of Line of Code

Number of Line of Code = Approx 1000 Lines

Step4: Calculating the Effort

Effort = 3.6 x $(1)^{1.20}$ = Approx (3.6 Persons-months)

Step5: Budget Calculation

Assuming a monthly Student salary of $**100**

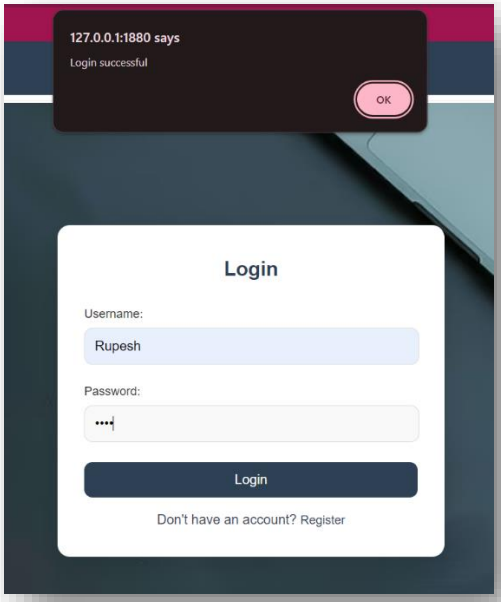
Cost = 100 x 3.6 = $360

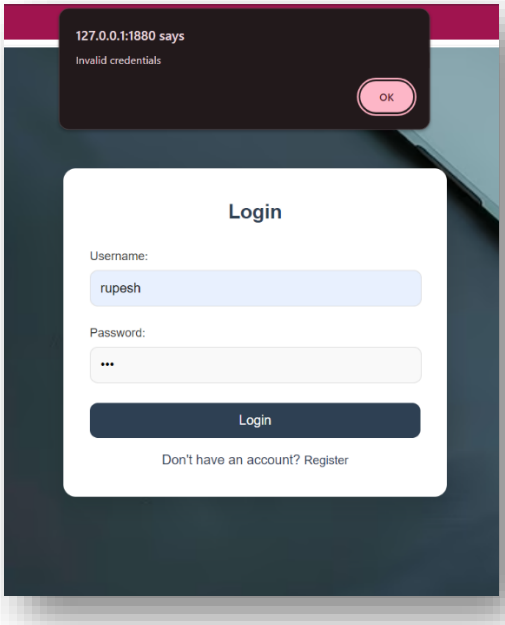
Total Costing:

Component Cost + Software Parameter + Effort = $36.70 + $3.78 + $360 = $404.18.

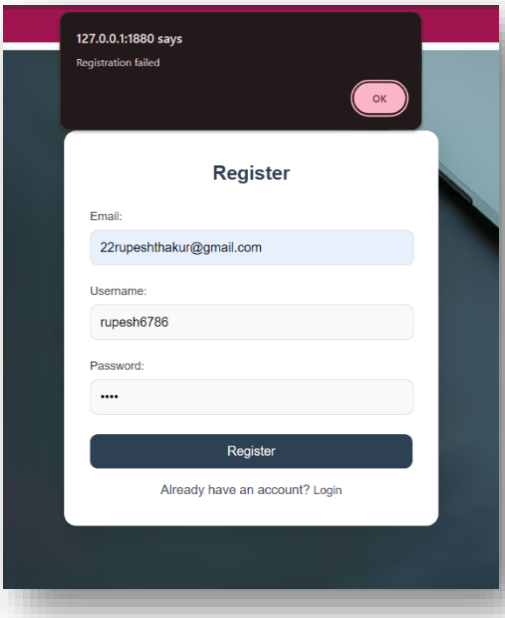
6. SYSTEM TESTING

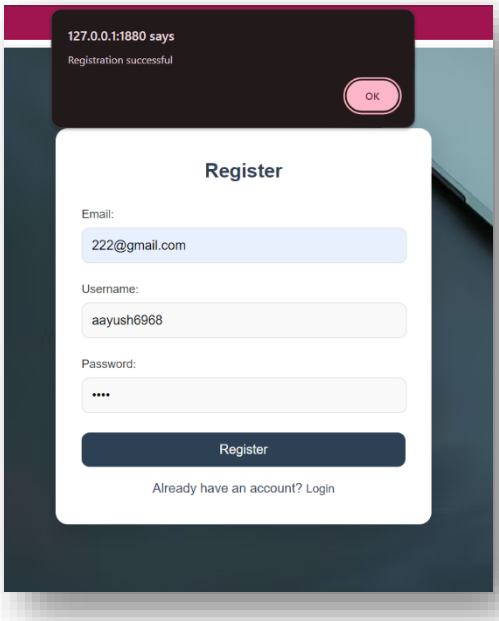
Test Case 1: User Login Functionality



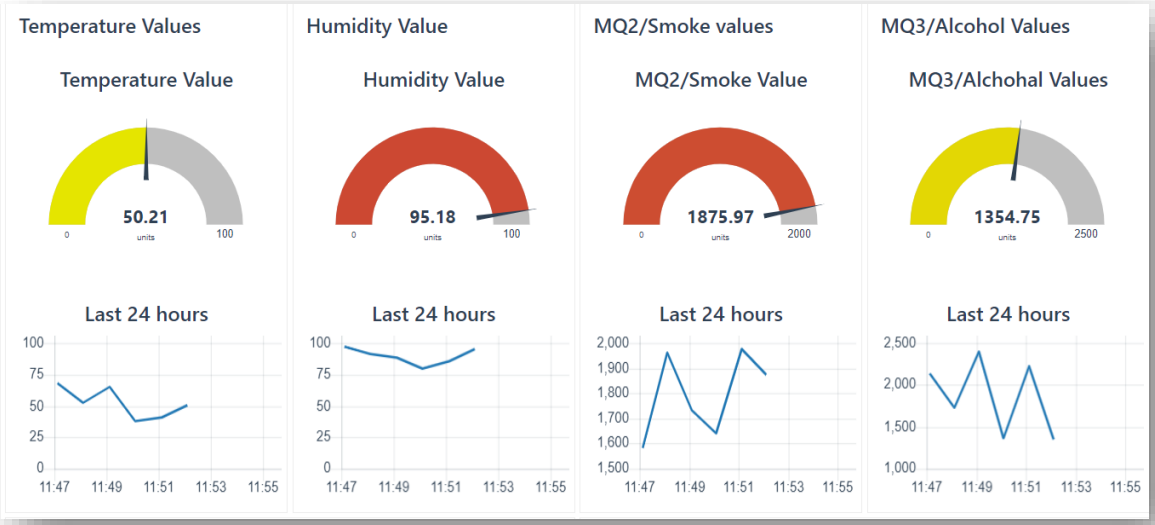


Test Case 2: User Registration Functionality

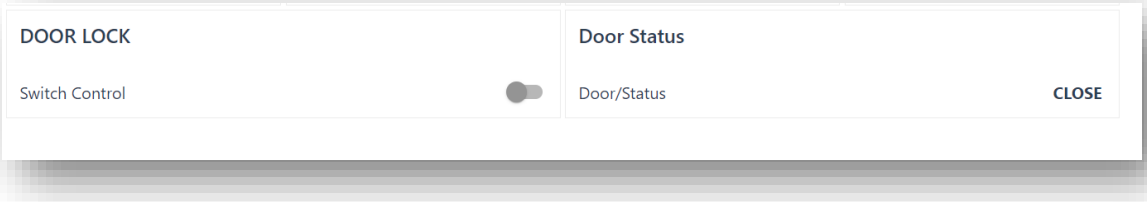




Test Case 3: Sensor Data Visualization

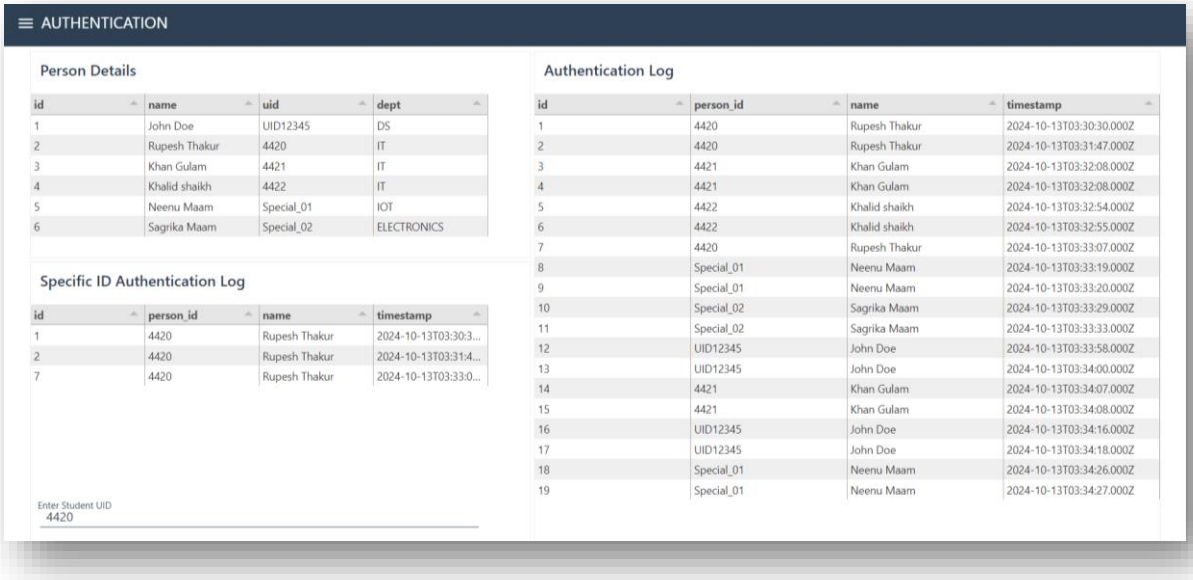


Test Case 4: Solenoid Lock Operation

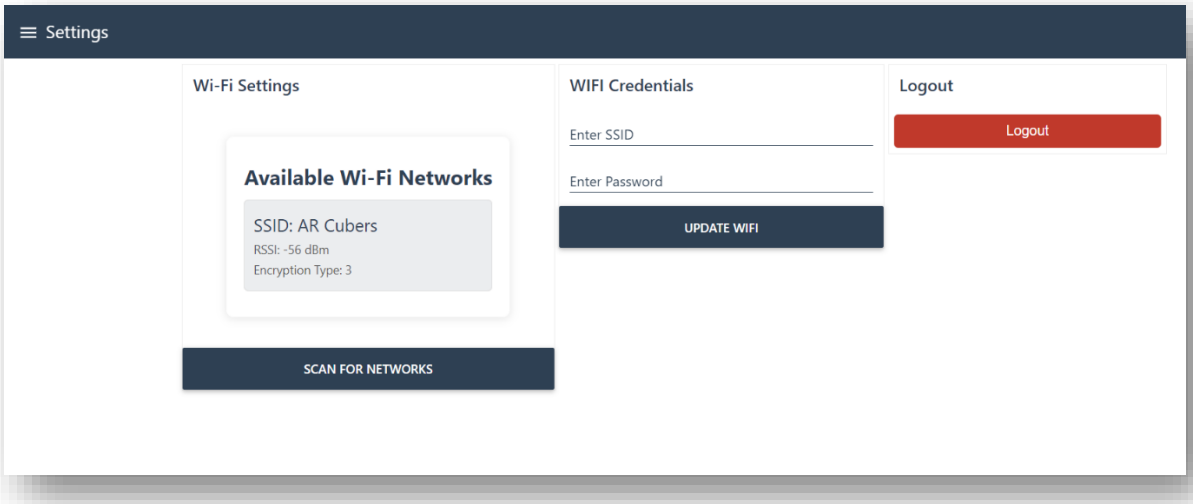


Test Case 5: Surveillance Camera Operation

Test Case 5: User Authentication Log Details



Test Case 5: Settings Operation



7. SYSTEM MAINTENANCE AND EVALUTION

7.1. Maintenance:

System maintenance is a crucial phase in the project lifecycle that ensures the long-term functionality, security, and efficiency of the IoT-based security and surveillance system. This phase involves regular monitoring, updates, troubleshooting, and enhancements to maintain system reliability and performance. Maintenance is essential to prevent system failures, improve security measures, and adapt to evolving user needs.

Types of Maintenance:

1.Preventive Maintenance:

- Regularly checking and updating firmware to avoid potential failures.
- Ensuring that the ESP32 microcontroller and other hardware components function optimally.
- Cleaning and inspecting sensors (MQ2, MQ3, DHT22, fingerprint, RFID) for accurate data collection.

2.Corrective Maintenance:

- Identifying and fixing system bugs or errors in hardware/software.
- Replacing damaged components (e.g., relay, solenoid lock, ESP32 module).
- Debugging MQTT communication issues to ensure reliable data transfer.

3.Adaptive Maintenance:

- Updating the system to integrate new security measures (e.g., additional authentication methods).
- Enhancing the Node-RED dashboard for better user experience and functionality.
- Expanding the system by adding new sensors or upgrading components.

4.Perfective Maintenance:

- Optimizing the system for better performance, such as improving response time for sensor readings and authentication.
- Upgrading the power supply for increased efficiency and reliability.
- Reducing false alarms by refining threshold values for sensor detection.

7.2. Evaluation:

System evaluation is performed to assess the effectiveness, reliability, and usability of the IoT-based security and surveillance system. This process ensures that the system meets its intended objectives and provides valuable feedback for further improvements.

Evaluation Criteria:

1. **Functionality Testing:**
    - Ensuring that all sensors and components work as expected.
    - Verifying the accuracy of sensor readings and authentication success rates.
    - Testing the automation of door access control using RFID and fingerprint verification.
2. **Performance Analysis:**
    - Measuring system response time for sensor data processing, authentication, and alerts.
    - Evaluating network stability and MQTT communication efficiency.
    - Checking the real-time video streaming quality of the ESP32-CAM module.
3. **Security Assessment:**
    - Performing penetration testing to identify vulnerabilities in the system.
    - Ensuring encrypted communication between the ESP32 and MQTT broker.
    - Testing unauthorized access scenarios to validate security measures.
4. **User Experience & Feedback:**
    - Collecting feedback from users to improve dashboard usability.
    - Enhancing mobile notifications and alert mechanisms for better real-time response.
    - Analysing ease of system operation and troubleshooting guidelines.

5. **Reliability & Fault Tolerance:**

    - Stress testing the system under different environmental conditions (temperature, humidity, gas exposure).
    - Evaluating how the system handles sensor failures or unexpected power loss.
    - Ensuring the system continues functioning during network interruptions.

8. USER / OPERATIONAL MANUAL

1. Introduction

1.1 Overview

This manual provides step-by-step instructions for setting up, operating, and maintaining the IoT-based security and surveillance system. The system integrates multiple sensors, including gas detection, temperature monitoring, fingerprint authentication, RFID access control, and real-time video surveillance. Data is managed via an ESP32 microcontroller, communicated through MQTT, and visualized using a Node-RED dashboard.

1.2 System Components

- **ESP32**: Central microcontroller for data processing and communication.
- **ESP32-CAM**: Captures real-time video footage.
- **MQ2 & MQ3 Sensors**: Detect flammable gases and LPG leaks.
- **DHT22 Sensor**: Monitors temperature and humidity.
- **R307 Fingerprint Sensor**: Provides biometric authentication.
- **RFID Module (WS1850S)**: Ensures secure access control.
- **Reed Switch**: Detects door status (open/closed).
- **12V Relay & Solenoid**: Automates door locking/unlocking.
- **MQTT Broker**: Handles data transmission between system components.
- **Node-RED Dashboard**: Provides a visual interface for system control and monitoring.

2. System Setup and Installation

2.1 Hardware Installation

1. **Mount Sensors & Modules:** Securely attach the sensors in the designated locations (e.g., MQ2 near gas sources, DHT22 in an open area, RFID/fingerprint near the entry door).
2. **Connect Components to ESP32:**

    - Follow the provided wiring diagram.
    - Ensure correct GPIO connections for sensors and modules.

3. **Power Supply Setup:**

    - Use a 12V DC power source for the relay and solenoid.
    - ESP32 should be powered via a 5V USB adapter or regulated power supply.

4. **Network Configuration:**

    - Ensure Wi-Fi connectivity for the ESP32 to communicate with the MQTT broker.
    - Configure the MQTT broker on a Raspberry Pi or cloud service.

2.2 Software Setup

1. **Firmware Upload:**

    - Use Arduino IDE or ESP-IDF to upload the firmware to ESP32.
    - Install necessary libraries (e.g., MQTT, fingerprint, RFID).

2. **Node-RED Dashboard Setup:**

    - Install Node-RED on a PC/Raspberry Pi.
    - Import the system dashboard flow.
    - Configure MQTT topics for data visualization and control.

3. **Database Setup :**

    - Store access logs and sensor data using MySQL or Firebase.

3 System Operation

3.1 Access Control & Authentication

1. **RFID Authentication:**

    - Tap an authorized RFID card on the reader.
    - If valid, the solenoid unlocks the door.
    - Unauthorized attempts trigger an alert.

2. **Fingerprint Authentication:**

    - Place a registered fingerprint on the R307 sensor.
    - If matched, access is granted.
    - Failed attempts trigger an alert.

3.2 Surveillance & Monitoring

1. **Live Video Streaming:**

    - Open the Node-RED dashboard.
    - Access the camera feed from the ESP32-CAM module.

2. **Sensor Data Monitoring:**

    - View real-time values for temperature, humidity, and gas concentration.
    - System triggers alarms if predefined thresholds are exceeded.

3.3 Alert System & Notifications

1. **Real-Time Alerts:**

    - Notifications via email, SMS, or Telegram for security breaches.
    - Alerts when gas levels are high, or an unauthorized access attempt occurs.

2. **Automated Actions:**

    - Door auto-locking after unsuccessful authentication attempts.
    - Buzzer activation for emergency alerts.

4. System Maintenance

4.1 Routine Maintenance

- Regularly clean the fingerprint sensor and RFID reader.
- Check gas sensors for dust accumulation.
- Inspect wiring connections for wear or damage.
- Ensure ESP32 firmware is up to date.

9. FUTURE WORK

1. **AI/ML - Powered Facial Recognition**:
    - **Surveillance Camera Enhancement**: Implement AI/ML algorithms (e.g., OpenCV, TensorFlow Lite) for facial recognition. This will enable real-time identification of authorized and unauthorized individuals, enhancing overall system security.
    - **Facial Recognition Database**: Set up a MySQL or SQLite database to store and manage user profiles. The system will log entries and allow automatic updates when new users are added.
2. **Additional Sensors**:
    - **Depth Sensor for Object Detection**: Integrate a depth sensor to improve the accuracy of facial and object detection by analysing 3D data. This will reduce false positives, making the security system more reliable.
    - **Motion Sensor**: Add motion sensors to trigger alarms or surveillance when movement is detected, providing better real-time monitoring of secured areas.
3. **Remote Firmware Updates for ESP32**:
    - Implement OTA (Over-The-Air) firmware updates for the ESP32 microcontroller, enabling seamless updates to the system software without requiring physical access. This feature will ensure easy maintenance and upgrades.
4. **Ecosystem for Cross-Platform Access**:
    - **Android and Windows Applications**: Create an ecosystem that includes Android apps and Windows software to allow users to access and manage the security system dashboard from anywhere. By offering a cross-platform interface, users will have flexibility to monitor sensor data, view camera feeds, and control security features remotely.
    - **Other OS Capabilities**: Extend support to other operating systems, such as iOS and Linux, ensuring universal accessibility across devices for remote management and monitoring.
5. **Enhanced Remote Management**:
    - Expand the remote management functionality, allowing users to control the system (e.g., lock/unlock doors, view real-time surveillance) via mobile or web apps. This will make the system accessible and manageable from any location.

10. CONCLUSION

The Internet of Things (IoT) has transformed security and surveillance, moving away from traditional systems that often lack integration and real-time responsiveness. This project presents an advanced IoT-based security and surveillance system, utilizing the ESP32 microcontroller as the core processing unit to enhance monitoring and control capabilities.

1. **System Overview**

- This IoT security system integrates various sensors, including MQ2 and MQ3 gas sensors for detecting harmful gases, and a DHT22 sensor for monitoring humidity and temperature. The ESP32-CAM module provides live video surveillance, offering users visual confirmation of security events. A user-friendly dashboard created with Node-RED allows for easy interaction with the system, enabling real-time data visualization and control.
- Automated access control is achieved through a 12V relay and solenoid, allowing secure entry for authorized individuals. The system employs RFID technology for enhanced access management. Using the MQTT protocol ensures efficient communication among components, enabling swift responses to detected anomalies.

2. Real-Time Monitoring and Alerts

- A key feature of this system is its real-time monitoring capability. Users receive instant notifications through SMS or app alerts when security breaches or hazardous conditions are detected. This proactive alert system significantly reduces response times, empowering users to take immediate action. Customizable thresholds for alerts allow users to tailor the system to their specific needs, enhancing its effectiveness.

3. Future Enhancements

- Future developments could integrate artificial intelligence and machine learning for improved threat detection and predictive analytics. This would enable the system to

learn from past incidents and adjust its responses accordingly. Exploring edge computing could also enhance data processing speed and efficiency.

- Additionally, incorporating blockchain technology may improve data security and integrity, making the system more robust. There is also potential to expand the project into a comprehensive ecosystem with cross-platform applications, allowing users to monitor their security systems from anywhere.

In conclusion, this IoT-based security and surveillance system addresses the shortcomings of traditional solutions while advancing smart security technology. By integrating multiple sensors and IoT capabilities, it enhances safety in various settings. This project serves as a foundation for future developments in smart home technology, contributing to a safer environment for all. As IoT continues to evolve, this system can adapt and grow, meeting the increasing demand for reliable and sophisticated security solutions.

11. REFERENCES

1. **Mishra, A., & Ghosh, A. (2019).** "Internet of Things (IoT): A Review of Concepts, Applications, and Challenges." *Journal of Computer Networks and Communications*, 2019. DOI: 10.1155/2019/5867351
2. **Sarkar, S., & Ghosh, A. (2020).** "IoT-Based Smart Security System Using ESP32 Microcontroller." *International Journal of Engineering Research & Technology (IJERT)*, 8(12). Link
3. **Ravikumar, K., & Patil, S. (2021).** "Design and Implementation of IoT Based Smart Home Security System." *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 6(2), 46-50. Link
4. **Gurjar, H., & Jain, R. (2018).** "An Overview of Smart Surveillance System Using IoT." *International Journal of Advanced Research in Computer Science*, 9(3), 1-5. DOI: 10.26483/ijarcs.v9i3.6736
5. **Gheorghe, A. G., & Muntean, C. H. (2020).** "Smart Surveillance Systems: A Comprehensive Overview." *MDPI Sensors*, 20(21), 6155. DOI: 10.3390/s20216155
6. **Xia, F., Yang, L. T., & Wang, L. (2017).** "The Internet of Things: A Survey." *International Journal of Information and Computer Security*, 9(4), 388-410. Link
7. **Rodrigues, A. C., & Almeida, M. F. (2020).** "MQTT Protocol: A Survey on its Applications in IoT." *IEEE Internet of Things Journal*, 7(10), 8953-8970. DOI: 10.1109/JIOT.2020.2990982
8. **Sakurai, K., & Matsuura, H. (2017).** "A Survey of the MQTT Protocol and its Applications in IoT." *Journal of Information Processing*, 25, 1-10. DOI: 10.2197/ipsjjip.25.1

Feel free to use and adapt these references based on your project's specific needs!