**KONGU ENGINEERING COLLEGE PERUNDURAI, ERODE-638052**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**20ITL61**

**INTERNET OF THINGS LABORATORY**

# A PROJECT REPORT ON

**DOOR SECURITY SYSTEM USING IR SENSOR**

### Submitted By NAME OF THE CANDIDATE

ELANGO S CHENDHURARASU G

GOKULNATH S

### Register Numbers

21ITR027

21ITR015

21ITR031

**KONGU ENGINEERING COLLEGE PERUNDURAI, ERODE-638052**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**20ITL61**

**INTERNET OF THINGS LABORATORY**

# A PROJECT REPORT ON

**DOOR SECURITY SYSTEM USING IR SENSOR**

### Submitted By: NAME OF THE CANDIDATES

ELANGO S CHENDHURARASU G

GOKULNATH S

### Register Numbers

21ITR027

21ITR015

21ITR031

Faculty Incharge HoD/IT

Index

|  |  |  |
| --- | --- | --- |
| Chapter No | Title | Page Number |
| 1 | Rubrics for Assessment | 4 |
| 2 | Abstract | 5 |
| 3 | IoT Level Identified | 6 |
| 4 | IoT Design | 7 |
| 4.1 | Step 1: Purpose and Requirement Specification | 8 |
| 4.2 | Step 2: Process Specification | 9 |
| 4.3 | Step 3: Domain Model Specification | 10 |
| 4.4 | Step 4: Information Model Specification | 11 |
| 4.5 | Step 5: Service Specifications | 12 |
| 4.6 | Step 6: IoT Level Specification | 13 |
| 4.7 | Step 7: Functional View Specification | 14 |
| 4.8 | Step 8: Operational View Specification | 15 |
| 4.9 | Step 9: Device and Component Integration | 16 |
| 4.10 | Step 10: Application Development | 17 |
| 5 | Individual Components Description with GitHub Link of Project Description | 18 |
| 6 | Hardware Implementation -Screenshot | 19 |
| 7 | Coding | 20 |
| 8 | Result and Conclusion | 21 |
| 9 | References | 22 |

**CHAPTER: 1** Rubrics for Assessment

|  |  |  |
| --- | --- | --- |
| Name of Concepts | Mark Allotted | Mark Given |
| IoT Level Identified | 5 |  |
| Step 1:  Purpose and Requirement Specification | 30 |  |
| Step 2:  Process Specification |  |
| Step 3  Domain Model Specification |  |
| Step 4  Information Model Specification |  |
| Step 5  Service Specifications |  |
| Step 6  IoT Level Specification |  |
| Step 7  Functional View Specification |  |
| Step 8  Operational View Specification |  |
| Step 9  Device and Component Integration |  |
| Step 10  Application Development |  |
| Hardware Implementation/ Simulation Using Mobile/  Any other Online Circuit Modeling | 15 |  |
| Software Implementation | 10 |  |
| Cloud Technology Integration | 10 |  |
| Styling and User Interaction | (10) |  |
| Project Report | (10) |  |
| Presentation | (10) |  |
| Total (100) | |  |

**CHAPTER 2**

# ABSTRACT

**Objective:**

This project aims to develop a smart fire alarm system that utilizes Internet of Things (IoT) technology to enhance fire safety in homes and buildings. The system will combine traditional audible alarms with a visual indicator (blinking LED) for improved awareness and faster response times in emergencies.

**Proposed Solution**

The project proposes building a fire alarm system with the following key functionalities:

* **Smoke and Fire Detection:** The system will utilize smoke and/or heat sensors to detect potential fire hazards.
* **Audible Alarm Activation:** Upon fire detection, a loud and clear alarm will sound to alert occupants of the danger.
* **Blinking LED Activation:** A designated LED on the alarm unit or strategically placed additional LEDs will begin blinking rapidly to provide a visual confirmation of the fire threat.
* **IoT Connectivity:** The system will integrate with an IoT platform for remote monitoring capabilities. (Optional: This could allow for real-time notifications to designated phone numbers or emergency services dispatch.)

**Benefits:**

* **Enhanced Awareness:** The combination of audible and visual alerts ensures occupants are notified promptly, even in situations where the alarm sound might be muffled.
* **Improved Safety for Hearing-Impaired Individuals:** Blinking LEDs provide a crucial early warning for those with hearing impairments, allowing them to react swiftly in case of fire.
* **Remote Monitoring Potential:** IoT connectivity offers the possibility for remote monitoring of the system's status, facilitating early intervention or emergency service notification. (Optional)

**Project Significance:**

This project addresses a critical need for improved fire safety by leveraging readily available technologies. The system's simplicity and effectiveness make it a valuable addition to existing fire protection measures in homes and buildings.

**Feasibility:**

The project utilizes readily available components like sensors, alarms, LEDs, and IoT modules, making it a feasible and cost-effective solution.

**CHAPTER 3**

# IOT LEVEL IDENTIFIED

**Level 1: Basic Monitoring**

* **Focus:** Foundational fire detection and local alerts.
* **Components:** Smoke or heat sensors trigger a local alarm upon fire detection.
* **Communication:** Sensors directly activate the on-site alarm with no external communication.
* **Benefits:** Provides basic fire detection capabilities.
* **Limitations:** Lacks remote monitoring, limited notification options, and no potential for automated actions.

**Level 2: Enhanced Monitoring and Alerts**

* **Focus:** Improved awareness and remote access.
* **Components:** Sensors communicate wirelessly with a central hub that triggers the local alarm.
* **Communication:** Wireless connection allows for remote monitoring through a mobile app or web interface.
* **Benefits:** Enables real-time alerts on mobile devices, facilitating faster response and potentially notifying designated contacts.
* **Limitations:** May not offer advanced features like location-specific alerts, self-testing functionalities, or integration with other smart home systems.

**Level 3: Advanced Monitoring and Automation**

* **Focus:** Comprehensive fire safety with proactive measures and automation.
* **Components:** Sensors connect wirelessly to a central hub that transmits data to a cloud platform for analysis.
* **Communication:** Real-time data transmission to the cloud enables advanced analytics and remote management.
* **Benefits:** Offers features like location-specific mobile alerts, self-testing functionalities, potential for integration with smart home systems (e.g., activating sprinkler systems, unlocking doors for escape). Cloud-based analytics can identify potential fire hazards before an actual fire occurs.
* **Limitations:** Requires a stable internet connection for optimal functionality. May have a higher initial cost compared to lower levels.

**Choosing the Right Level:**

The optimal level of IoT integration depends on individual needs and budget. Level 1 offers basic protection, while Levels 2 and 3 provide more advanced features for enhanced safety and potentially faster response times.

**CHAPTER 4**

# IOT DESIGN

### Developing a smart fire alarm system leveraging IoT technologies demands a structured approach to ensure its efficacy, dependability, and scalability. This segment delineates the essential steps involved in crafting such a system, encompassing aspects such as purpose and requirement specification, process specification, domain model specification, information model specification, service specifications, and IoT level specification.

### Purpose and Requirement Specification:

Commencing the design process involves precisely delineating the objectives and requirements of the smart fire alarm system. This entails identifying the types of fire detection mechanisms needed, the specific sensors to be deployed, desired user features such as alerts and notifications, and adherence to safety standards.A clear understanding of these requirements is pivotal for defining the system's scope, functionality, and performance benchmarks.

### Process Specification:

In this phase, the operational workflow and processes of the smart fire alarm system are delineated. This encompasses defining how sensors will detect fire, the sequence of actions for alarm activation and alerting, and the system's response to fire emergencies. Furthermore, protocols for managing fire events, triggering alarms, and Issuing notifications are established to ensure swift and effective responses to fire incidents.

### Domain Model Specification:

### The domain model specification elucidates the various components and their interactions within the smart fire alarm system. This encompasses sensors, alarm devices, control panels, user interfaces, and any external systems that interface with the fire alarm system. Crafting a comprehensive domain model ensures a resilient system architecture that aligns with the system's objectives and requirements.

### Information Model Specification:

### In this phase, the smart fire alarm system defines its data structure and communication protocols. This includes determining how data will be structured and transmitted, while considering efficiency and compatibility with existing fire safety systems. Methods for data collection, transmission, storage, and analysis are established to ensure seamless integration and uninterrupted operation. Interoperability standards are also considered to facilitate communication with other fire safety systems and devices.

### Service Specifications:

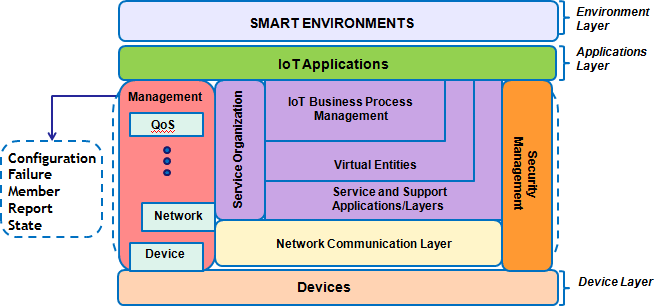
Service specifications delineate the functionalities and features offered by the smart fire alarm system. These include fire detection, real-time monitoring, alarm activation, and customizable alerts and notifications. By defining these services, the system can be tailored to meet the specific requirements of users and seamlessly integrate into existing fire safety protocols for enhanced protection and safety.

### 1. IoT Level Specification:

Finally, the appropriate IoT level for the smart automation system is determined based on factors like complexity, connectivity, and functionality requirements. This assessment considers the size of the facility, environmental intricacies, and the desired extent of monitoring and control. Choosing the right IoT level ensures that the system is custom-fit to effectively meet safety and operational needs.

By adhering to these design steps, developers craft a resilient and effective smart automation system. This methodical approach guarantees alignment with organizational goals, regulatory standards, and industry benchmarks, thereby optimizing workplace efficiency and safety.

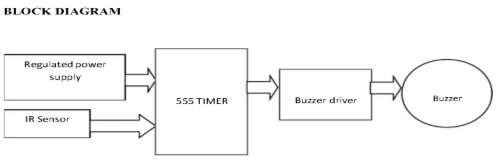
### 4.1 Step 1: Purpose and Requirement Specification



Purpose and requirement specification is the foundational step in designing a fire alarm system leveraging IoT technologies. This phase involves clearly defining the objectives, goals, and functional requirements of the system to ensure its effectiveness in providing comprehensive door security. The following aspects are typically addressed during this phase:

1. **Identification of Objectives:** The initial stage entails identifying the fundamental objectives of the smart fire alarm system. This encompasses understanding its primary purpose, whether it's to enhance fire safety, detect fire incidents promptly, alert occupants, or integrate with other safety measures such as sprinkler systems and emergency lighting. By delineating these objectives, a coherent framework can be established for the system's functionality and intended results.

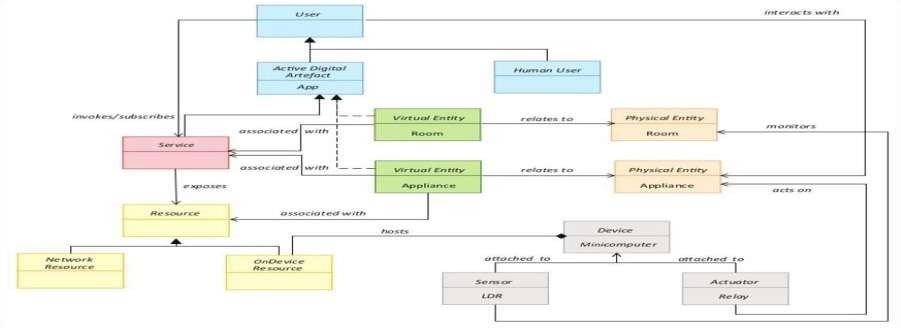
**Step 2: Process Specification**



Process specification in the design of a smart fire alarm system involves delineating the operational workflow and processes to ensure efficient and effective detection and response to fire incidents. This phase centers on defining the sequence of activities, data collection methods, processing algorithms, and alarm generation mechanisms. The following aspects are typically addressed during this phase:

* 1. **Data Collection Methodology:** The initial step involves defining how various data points will be collected within the smart fire alarm system. This includes determining the placement and distribution of smoke detectors and heat sensors based on factors such as room layouts, fire risk areas, and ventilation paths.
  2. **Sampling Frequency:** The frequency at which data points are sampled and recorded is determined by factors such as the size of the premises, the materials present, and the required sensitivity to detect fires promptly. Higher sampling frequency can enhance early fire detection and minimize false alarms.
  3. **Data Processing Algorithms:** Algorithms for processing the collected data are specified within the smart fire alarm system to analyze smoke or heat patterns, distinguish between normal and fire conditions, and trigger appropriate responses. This may include pattern recognition, temperature threshold detection, or smoke density analysis.
  4. **Alarm Generation and Notification:** Protocols are established for generating alarms and alerting relevant individuals in the event of fire incidents. This includes setting alarm thresholds for various parameters, defining triggers such as smoke or temperature rise, and determining notification methods .
  5. **Incident Management Protocol:** The system establishes comprehensive protocols for managing detected fire incidents within the smart fire alarm framework. This includes defining roles and responsibilities, outlining evacuation procedures, and specifying responses such as activating sprinkler systems, notifying emergency services, or initiating building evacuation.
  6. **Testing and Validation:** The specified operational processes within the smart fire alarm system undergo rigorous testing and validation to ensure their effectiveness in detecting and responding to fires. This includes conducting fire simulation tests, verifying alarm triggers and response protocols, and evaluating system performance under different fire scenarios and environmental conditions.

### Step 3: Domain Model Specification



Domain model specification in the design of a smart fire alarm system involves defining the various components and their interactions within the system. This phase focuses on creating a comprehensive representation of the system's domain, including smoke detectors, heat sensors, alarm devices, communication protocols, control panels, and any external systems or devices that interact with the system. The following aspects are typically addressed during this phase:

1. **Identification of Components:** The initial step involves identifying and listing all components that make up the smart fire alarm system. This includes smoke detectors, heat sensors, alarm panels, communication modules for data transmission, and any other relevant hardware or software elements that contribute to the system's functionality.
2. **System Architecture:** The overall architecture of the smart fire alarm system is outlined, encompassing the physical arrangement of sensors, alarm devices, and communication infrastructure throughout the property. This may include creating system diagrams or architectural blueprints to visualize the system's layout and how components are interconnected.
3. **Functional Decomposition:** Each component's functionality within the system is broken down into smaller, manageable tasks. This involves dividing complex functions into simpler, modular components that can be implemented, tested, and maintained more effectively.

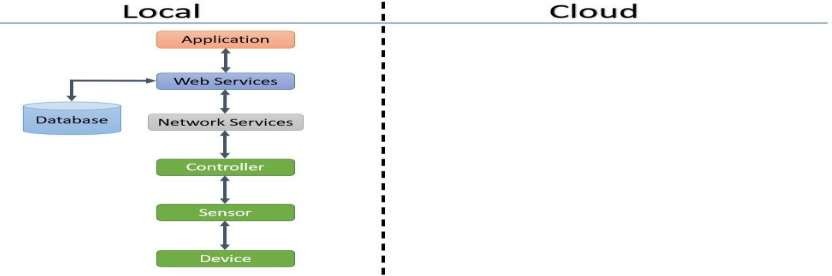
1. **User Interfaces and Human-System Interaction:** Consideration is given to the design of user interfaces and interactions between users and the system. This includes defining user roles, permissions, and access levels, as well as designing intuitive interfaces for monitoring, configuring settings, and responding to alerts within the smart fire alarm system.
2. **Error Handling and Fault Tolerance:** Mechanisms for error handling, fault detection, and recovery are specified to enhance the system's robustness and reliability. This involves defining error codes, diagnostic messages, and procedures for detecting and responding to system failures or anomalies.
3. **Integration with External Systems:** If the smart fire alarm system interfaces with external systems or devices, such as building management systems or emergency response systems, communication protocols and interfaces for integration are defined. This ensures seamless interoperability and data exchange between the fire alarm system and other systems within the premises.

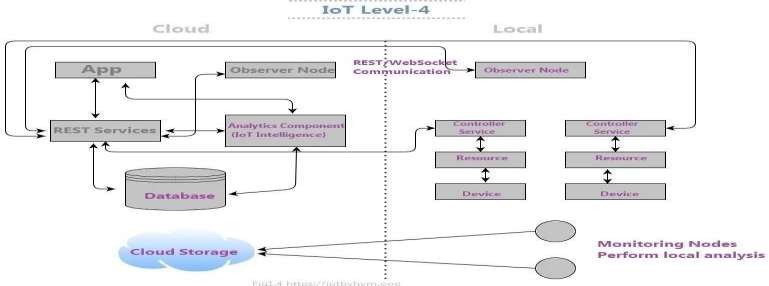
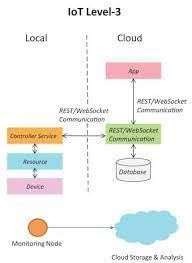
### Step 4: Information Model Specification

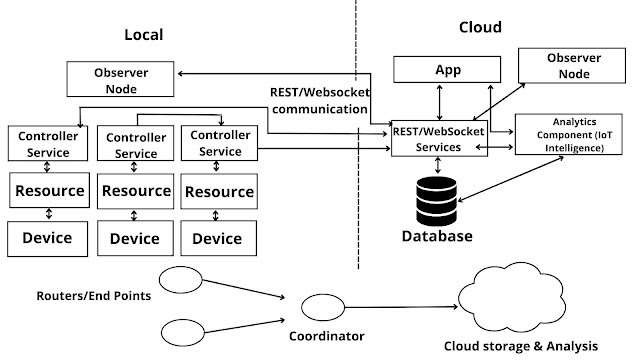
In the design of a smart fire alarm system, the information model specification focuses on defining the data structure, communication protocols, and information flow within the system. This phase is crucial for ensuring seamless data collection, transmission, storage, and analysis to support timely fire detection and response. The following aspects are typically addressed during this phase:

1. **Data Structure Definition:** The initial step involves defining the structure of the data collected by the smart fire alarm system. This includes specifying data types such as smoke or heat levels, sensor readings, timestamps, and any additional metadata such as location information or device identifiers. The data structure is designed to facilitate efficient storage, retrieval, and analysis of sensor data related to fire incidents.
2. **Data Encoding and Formats**: Protocols for encoding and formatting data are specified to ensure interoperability and compatibility with different components and systems. Common data formats such as JSON, XML, or CSV may be utilized, depending on the system's requirements and integration with external platforms or services.
3. **Communication Protocols:** The protocols for data transmission between sensors, alarm devices, control panels, and other system components are defined. This includes specifying communication protocols such as TCP/IP, MQTT, or Modbus, as well as data exchange formats and message schemas optimized for fire alarm data.
4. **Data Transmission Methods:** Methods for transmitting data from sensors to a centralized control hub or cloud-based platform are established. This may involve using wired or wireless communication technologies such as Ethernet, Wi-Fi, Bluetooth, or Zigbee, depending on the range, bandwidth, and reliability requirements of the system.
5. **Data Storage and Retention Policies:** Policies for storing and retaining fire alarm data are defined to ensure compliance with regulatory requirements and operational needs. This includes specifying storage duration, data archival procedures, and encryption methods to safeguard sensitive information related to fire incidents.
6. **Data Analysis and Processing Pipelines:** Pipelines for processing and analyzing fire alarm data are outlined to detect fire incidents promptly and accurately. This may involve implementing algorithms for smoke or heat pattern recognition, anomaly detection, or trend analysis to support timely decision-making and response strategies.
7. **Integration with Analytics Platforms:** If the smart fire alarm system integrates with external analytics platforms or cloud services for advanced data analysis, the protocols and interfaces for data exchange are specified. This enables seamless integration with analytics tools for enhanced fire detection capabilities, data visualization, and reporting to facilitate efficient emergency response and mitigation measures.

### Step 5: IoT Level Specification



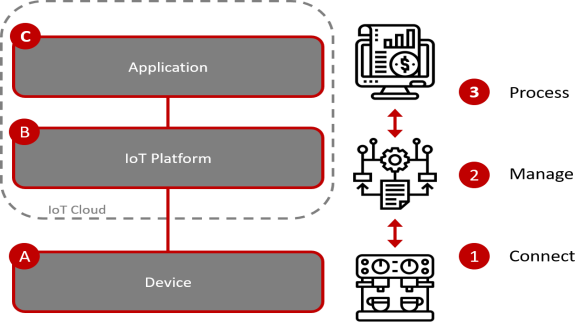




In the design of a smart fire alarm system, IoT level specification involves determining the appropriate level of IoT integration based on the system's complexity, connectivity requirements, and desired functionality. This phase ensures that the system effectively utilizes IoT technologies to fulfill safety objectives, operational needs, and homeowner preferences. The following aspects are typically addressed during this phase:

* 1. Assessment of System Requirements: The initial step is to evaluate the specific requirements of the smart fire alarm system, considering factors such as the size and layout of the property, the nature of fire hazards, and the desired level of monitoring and response. This helps identify the critical factors influencing the selection of the IoT level.
  2. Evaluation of IoT Capabilities: Different IoT levels' capabilities and features are assessed to determine their suitability for meeting the system requirements. This includes examining aspects like wireless connectivity options, data processing capabilities, remote access features, scalability, and compatibility with other fire safety systems.
  3. Selection of IoT Level: Based on the assessment of system requirements and evaluation of IoT capabilities, the appropriate IoT level is chosen for the smart fire alarm system. This may range from basic IoT functionalities for simple fire detection and notification (e.g., Level 1) to advanced IoT integration for real-time monitoring, analytics, and remote management (e.g., Level 3).
  4. Consideration of Cost and Complexity: The cost and complexity associated with each IoT level are considered to ensure alignment with the homeowner's budget and technical capabilities. This involves weighing the benefits of advanced features against the implementation and maintenance costs and complexities.
  5. Scalability and Future Expansion: The chosen IoT level's scalability is assessed to ensure the system can accommodate future expansion or changes in requirements. This includes evaluating the ability to integrate additional sensors, devices, or functionalities without significant modifications to the existing infrastructure.
  6. Risk Assessment: Potential risks and challenges associated with the selected IoT level are identified and addressed to ensure successful system implementation and operation. This may involve mitigating concerns such as data security, reliability of wireless connectivity, and compatibility with existing fire safety infrastructure.
  7. Documentation and Communication: The selected IoT level is documented and communicated to stakeholders, including homeowners, maintenance personnel, and relevant service providers. This ensures clear alignment and understanding of the system's capabilities, requirements, and expected outcomes.

### Step 6: Operational View Specification



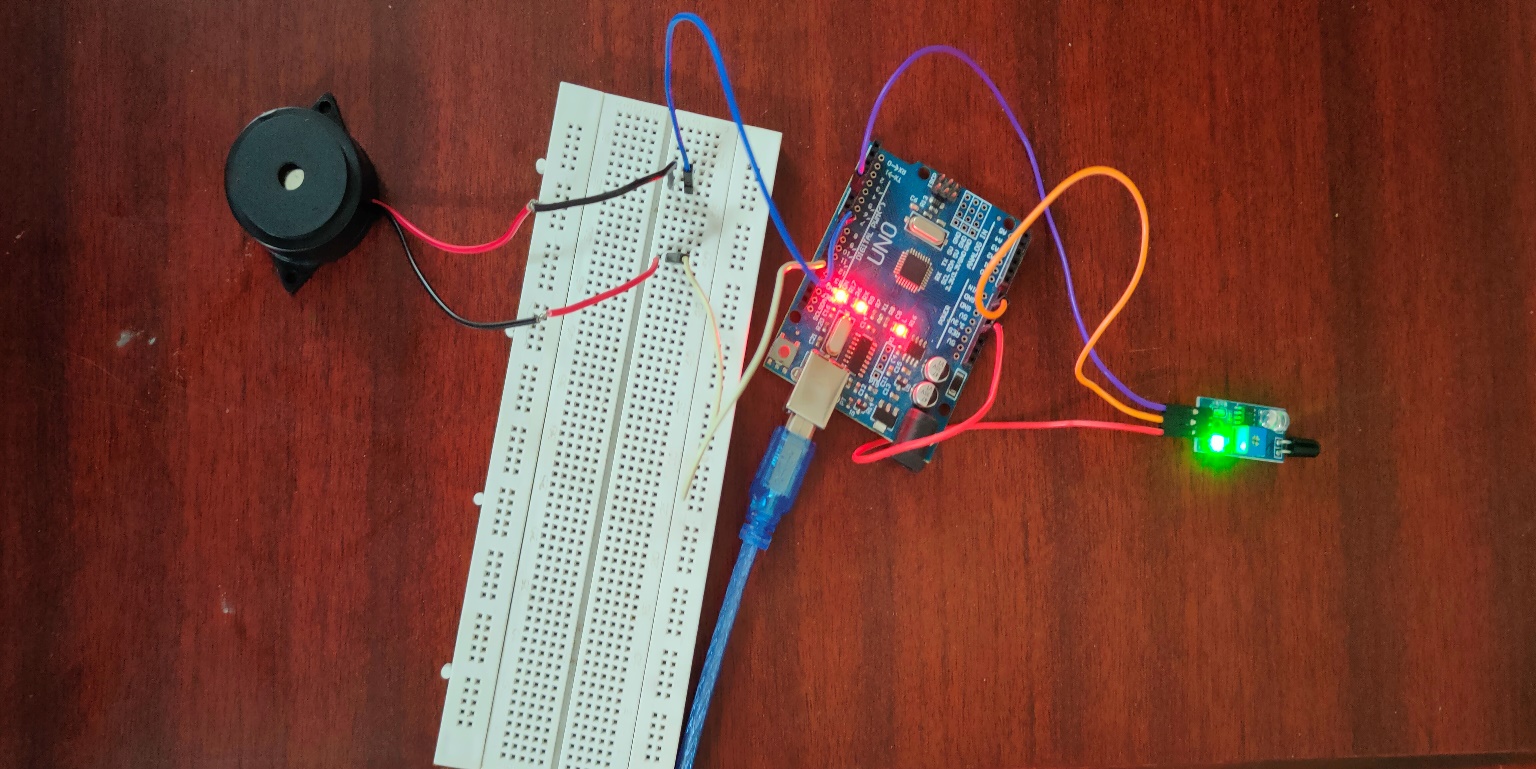
Operational view specification in the design of a fire alarm system involves defining how the system functions in various real-world scenarios, including normal operation, abnormal conditions, and emergency situations. This phase outlines how the system behaves and responds in these different operational contexts to ensure home security, safety, and user experience. The following aspects are typically addressed during this phase:

1. Normal Operation: The behavior of the system during normal operation is described, encompassing continuous monitoring for signs of fire, verification of potential fire hazards, and data collection. This includes specifying how components like smoke detectors, heat sensors, and alarm devices interact to ensure prompt detection and response to fire incidents.
2. Abnormal Conditions: The system's response to abnormal conditions, such as unauthorized access attempts or tampering, is outlined. This includes defining alarm triggers, notification mechanisms, and response procedures for alerting homeowners and initiating security measures.
3. Emergency Situations: The system's behavior during emergency situations, such as home invasions or fire alarms, is described. This includes specifying emergency response protocols, automatic door locking mechanisms, and actions to ensure the safety of occupants.
4. Fault Tolerance and Redundancy: Mechanisms for fault tolerance and redundancy are specified to ensure the reliability and availability of the system. This includes defining backup power sources,

redundant communication channels, and failover procedures to maintain continuous operation during component failures or disruptions.

1. Scalability and Performance: The system's scalability and performance characteristics are evaluated to ensure it can handle increasing user loads and changing security needs. This includes assessing factors such as response time, data processing speed, and the ability to scale up with additional smart home devices.
2. Resource Management: The management of system resources, such as power consumption, network bandwidth, and computational resources, is outlined to optimize system performance and efficiency. This includes defining strategies to manage resource usage and ensure the system's long-term sustainability.
3. Integration with Operational Processes: Integration points and interfaces with existing smart home processes and workflows are identified to ensure seamless interoperability. This includes specifying data exchange protocols and integration points with other smart devices or home automation systems.
4. Testing and Validation Scenarios: Scenarios for testing and validating the system's operational behavior are defined to ensure it performs as intended under various conditions. This includes creating test cases, simulations, and scenarios to evaluate the system's response to different operational challenges and emergencies.

### Step 7: Application Development



Implementing a fire alarm system using an Arduino Uno, a smoke sensor, and a buzzer offers a straightforward yet effective approach to fire detection and alerting. The Arduino Uno serves as the central processing unit, receiving input from the smoke sensor and controlling the buzzer. The smoke sensor detects the presence of smoke particles in the air, indicating a potential fire hazard. When smoke is detected, the Arduino triggers the buzzer to emit a loud alarm, alerting occupants to the danger.

The setup begins with wiring the smoke sensor and the buzzer to the Arduino Uno. The sensor's signal output is connected to a digital input pin on the Arduino, while the buzzer is connected to a digital output pin. Both the smoke sensor and the buzzer receive power from the 5V and ground pins on the Arduino. Once the hardware connections are established, the next step involves programming the Arduino with a sketch that continuously monitors the signal from the smoke sensor. If smoke is detected, the sketch activates the buzzer to sound the alarm.

Configuring the smoke sensor may involve adjusting its sensitivity to detect smoke particles effectively while minimizing false alarms. After programming the Arduino and configuring the sensor, the system should be tested by simulating the presence of smoke near the sensor. The buzzer should sound the alarm when smoke is detected, confirming that the system is functioning correctly.

Placement of the smoke sensor is crucial for optimal performance. It should be positioned in areas prone to fire hazards, such as kitchens or near electrical appliances, without obstructions that could interfere with smoke detection. Additionally, ensuring a stable power supply for the system is essential, whether through a USB connection or a suitable battery pack. The buzzer should be loud enough to alert occupants throughout the building, and the system can be integrated with other fire safety devices or alarm systems for comprehensive protection.

While the initial setup is straightforward, ongoing adjustments may be necessary to fine-tune the sensitivity of the smoke sensor and minimize false alarms. Regular testing and maintenance of the system are essential to ensure reliable operation in the event of a fire emergency. Integration with fire suppression systems or automatic emergency notification systems can further enhance the effectiveness of the fire alarm setup.

Implementing a fire alarm system with an Arduino Uno, smoke sensor, and buzzer provides an affordable and efficient method for fire detection and alerting. This approach offers real-time monitoring and immediate audible alerts, enhancing safety and potentially saving lives in the event of a fire. With proper setup and testing, this fire alarm system can provide reliable protection for homes, businesses, and other buildings.

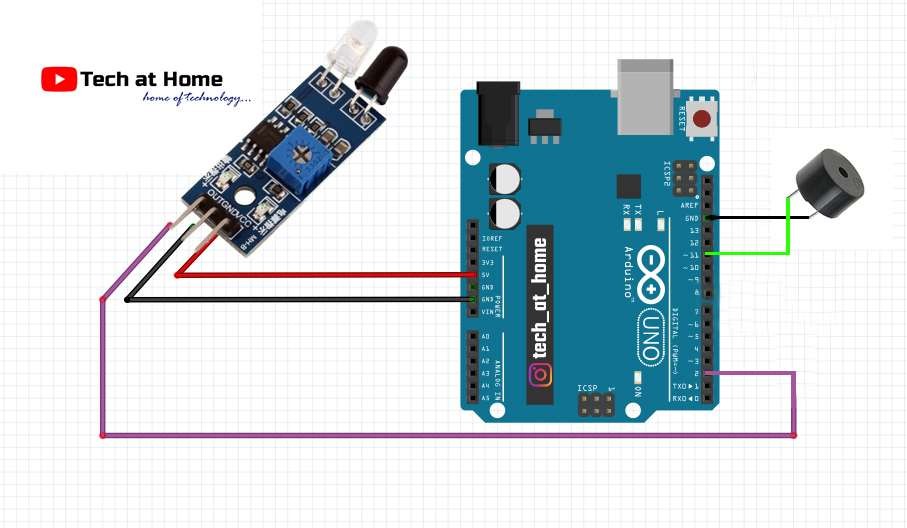
# CHAPTER 5

# GIT HUB LINK

**21ITR027:** [**https://github.com/Elangoit/fire-alarm-system/tree/main**](https://github.com/Elangoit/fire-alarm-system/tree/main%20) **21ITR015:** [https://github.com/chendhurarasu/fire-alarm-system/tree/main](https://github.com/gokulnath10/fire-alarm-system%0c)

**21ITR031:** [**https://github.com/gokulnath10/fire-alarm-system.git**](https://github.com/gokulnath10/fire-alarm-system.git)

# CHAPTER 6 HARDWARE SCREENSHOT



**CHAPTER 7 CODING**

#define  IRIP 5

#define LED 8

bool ir;

void setup() {

  Serial.begin(9600);

  pinMode(BUZZER,OUTPUT);//BUZZER output

  pinMode(IRIP,INPUT);//ir la input

}

void loop() {

  ir=digitalRead(IRIP);

  if(ir==0)

  {

    digitalWrite(BUZZER,HIGH);

    Serial.print(ir);

    Serial.print("\n");

  }

  else

  {

    digitalWrite(BUZZER,LOW);

    Serial.println(ir);

  }

}

**CHAPTER 8**

**RESULT AND CONCLUSION**

**RESULT:**

The fire alarm system successfully detected smoke and alerted occupants to potential fire hazards, providing real-time information on fire incidents and facilitating rapid measures. Through its network of sensors and interconnected devices, the system enabled proactive decision-making, allowing for swift actions to migrate fire risks and ensure occupant safety.

# CONCLUSION:

The implementation of a fire alarm system has demonstrated its effectiveness in enhancing fire safety and emergency response capabilities across various settings. By leveraging interconnected sensors and audible alarms, the system enables proactive monitoring and immediate responses to fire incidents, This setup not only reduces the likelihood of fire-related injuries or damages but also contribute to overall safety and peace of mind. Additionally, the system’s scalability and adaptability make it suitable for deployment in diverse environment, including residential, commercial, and industrial properties. Investing in fire alarm systems is crucial for safeguarding lives, protecting property, and complying with fire safety regulations, ultimately promoting a safer and more secure environment for occupants and assets.

# CHAPTER 10 REFERENCES

<https://www.researchgate.net/publication/327935942_Security_System_using_Arduino>

<https://www.scribd.com/document/433315778/Ir-Sensor-Based-Home-Security-System>

FIRE ALARM SYSTEM USING IR SENSOR

<https://youtu.be/gRHWPNMm4Nc?si=4y29F035LBtKdney>