Project Title:

IoT based security and surveillance system for railway stations.

Team Members

- Aniket Mukund Pawar
- Harshwardhan Anil Patil
- Kanishka Mittal
- Om Vijay Mangate
- Vishwajit Vikram Gaikwad
- Pratham Shah

Introduction

Objective:

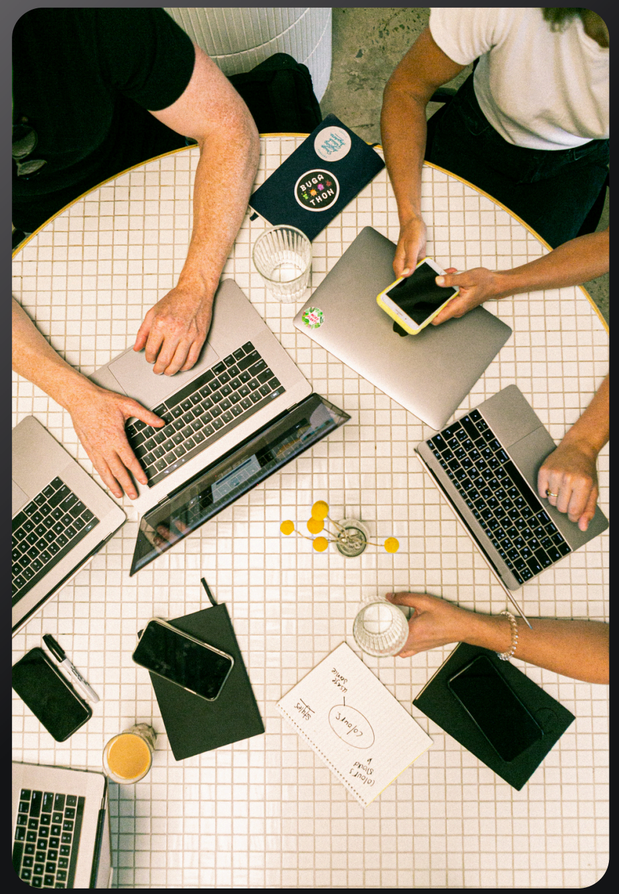
- Enhance security and surveillance at railway stations using IoT technology.
- Provide real-time monitoring and automated threat detection.

Why This Project?

- Increasing security threats at railway stations.
- Need for faster and more accurate surveillance systems.

Scope:

- Scalable to different public transport hubs.
- Can integrate with AI for advanced threat detection.



Problem Statement

1 Current Challenges:

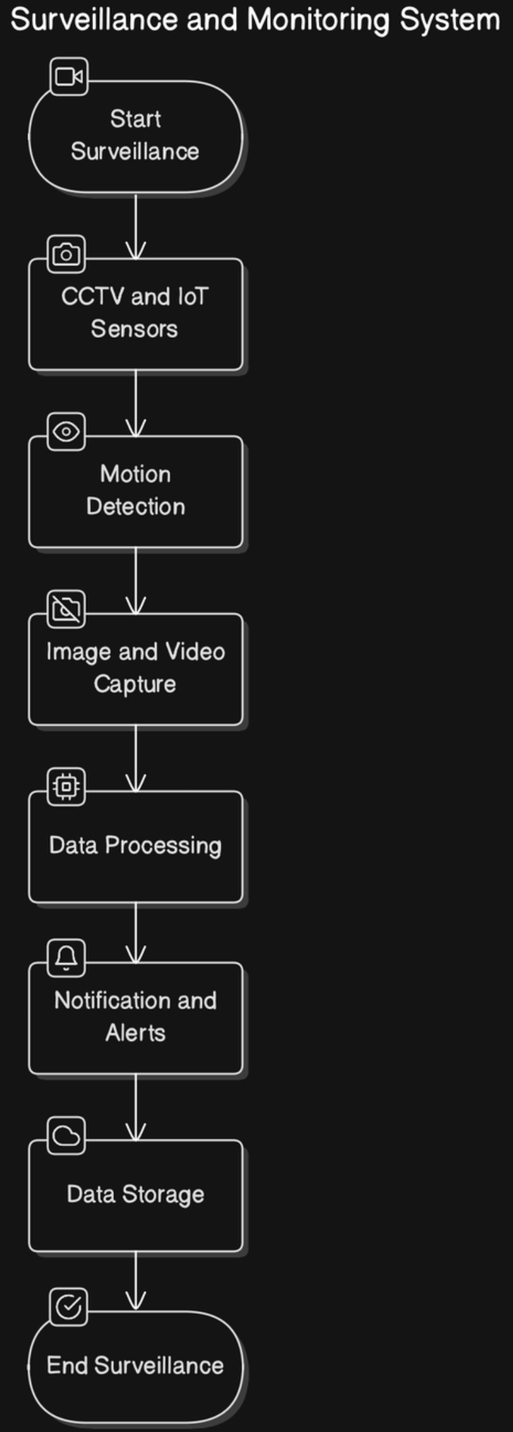
- Theft and unauthorized access.
- Unattended baggage and suspicious activities.
- Inefficient crowd management.
- Delayed response to emergencies.

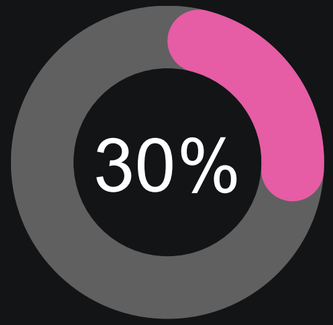
2 Need for Improvement:

- Real-time data collection and analysis.
- Automated monitoring and faster alerts.

WorkFlow

This workflow outlines the operation of an IoT-based security and surveillance system for railway stations. The system captures real-time data using CCTV cameras and IoT sensors, detects motion in restricted areas, and processes the captured images and videos. It then analyzes the data, triggers notifications and alerts in case of anomalies, and securely stores the data for future use and analysis. This end-to-end process ensures efficient monitoring, quick response, and enhanced safety at railway stations.





Motion Detection Data

This data helps in triggering real-time alerts and capturing video evidence, ensuring swift action against security breaches.

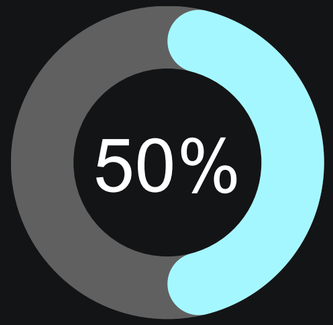
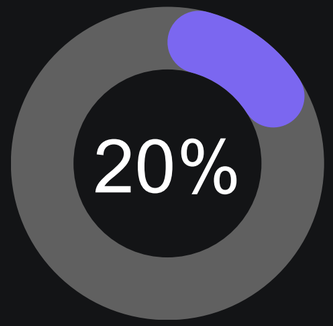


Image and Video Capture Data

This data aids in monitoring live activities, identifying suspicious behavior, and supporting investigations when needed.



Processed and Stored Data

This data enables pattern recognition, incident reviews, and long-term security planning, helping railway authorities improve safety protocols.

Existing Systems

CCTV Surveillance:

- Manual monitoring, prone to human error.
- Limited coverage and delayed response.

Security Personnel:

- High dependency on manpower.
- Inefficient during peak hours.

Metal Detectors and Baggage Scanners:

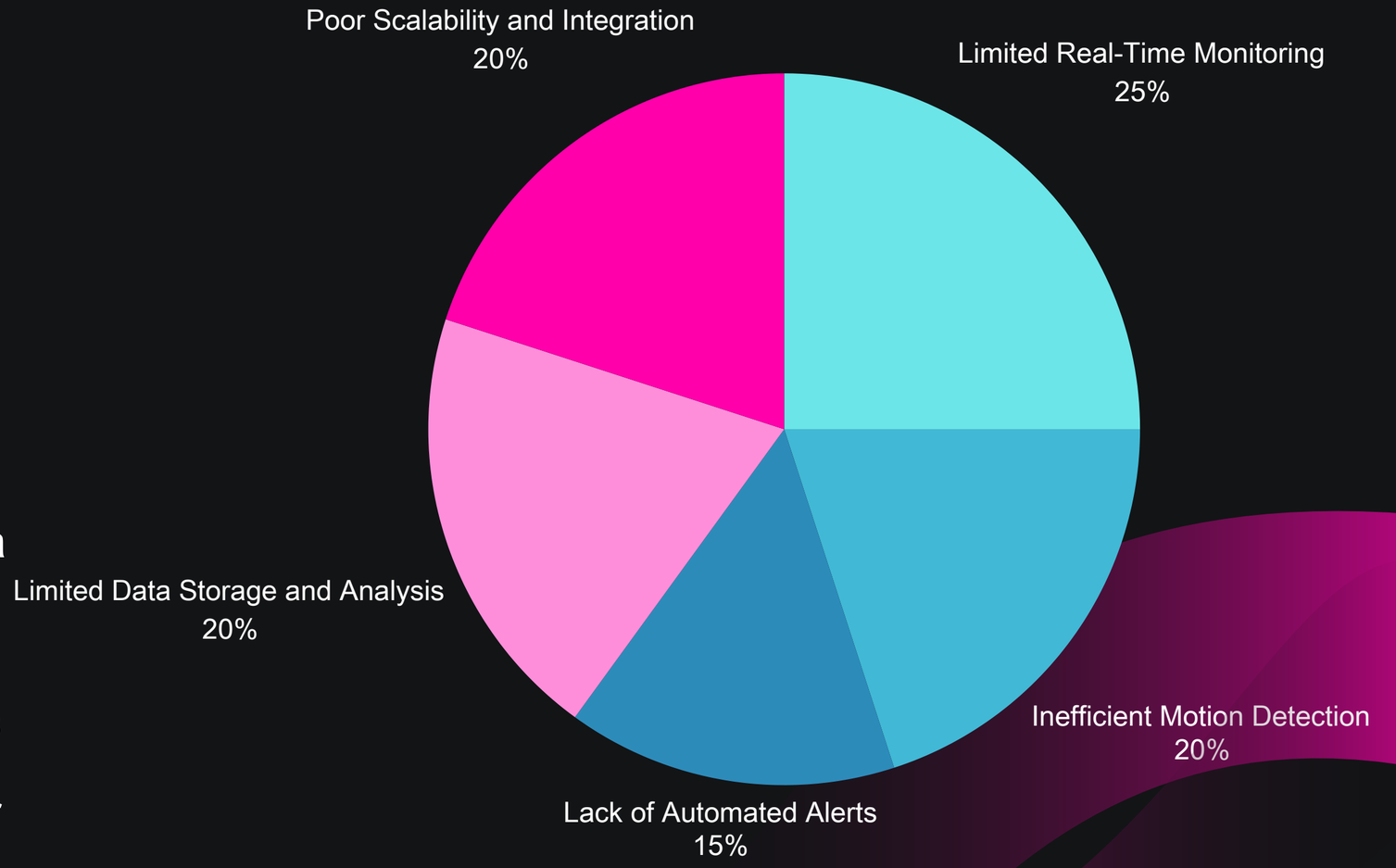
- Limited to entry points.
- Cannot monitor activities across the station.

Problems faced by existing security and surveillance systems

"Existing security and surveillance systems face several challenges that affect their efficiency and reliability.

These include delayed monitoring, high false alarm rates, lack of real-time alerts, limited data management capabilities, and difficulties in scaling the system.

Our IoT-based solution addresses these issues by integrating smart sensors, automated notifications, and advanced data processing for seamless and effective surveillance."



Proposed System

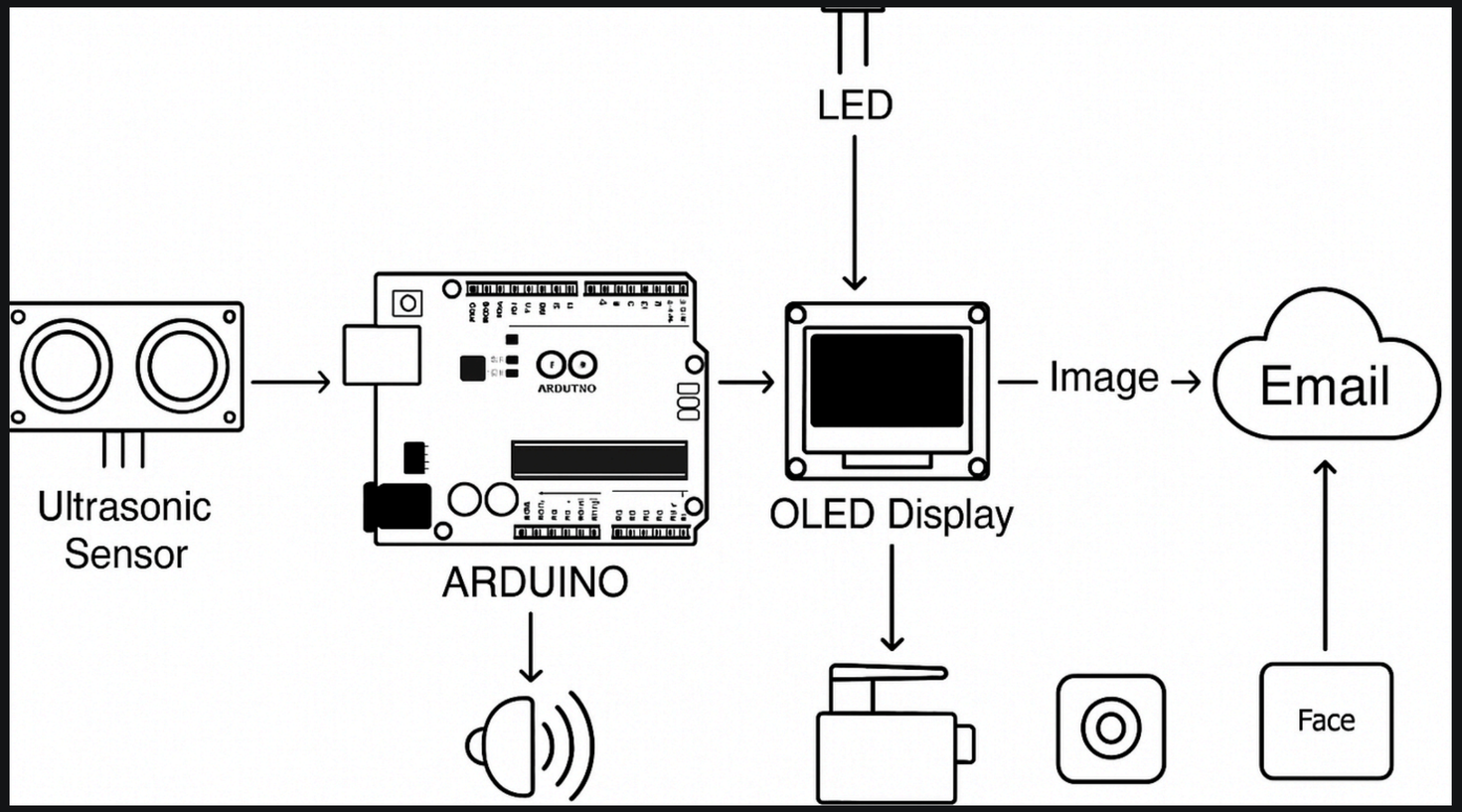


Figure : Block diagram of the system

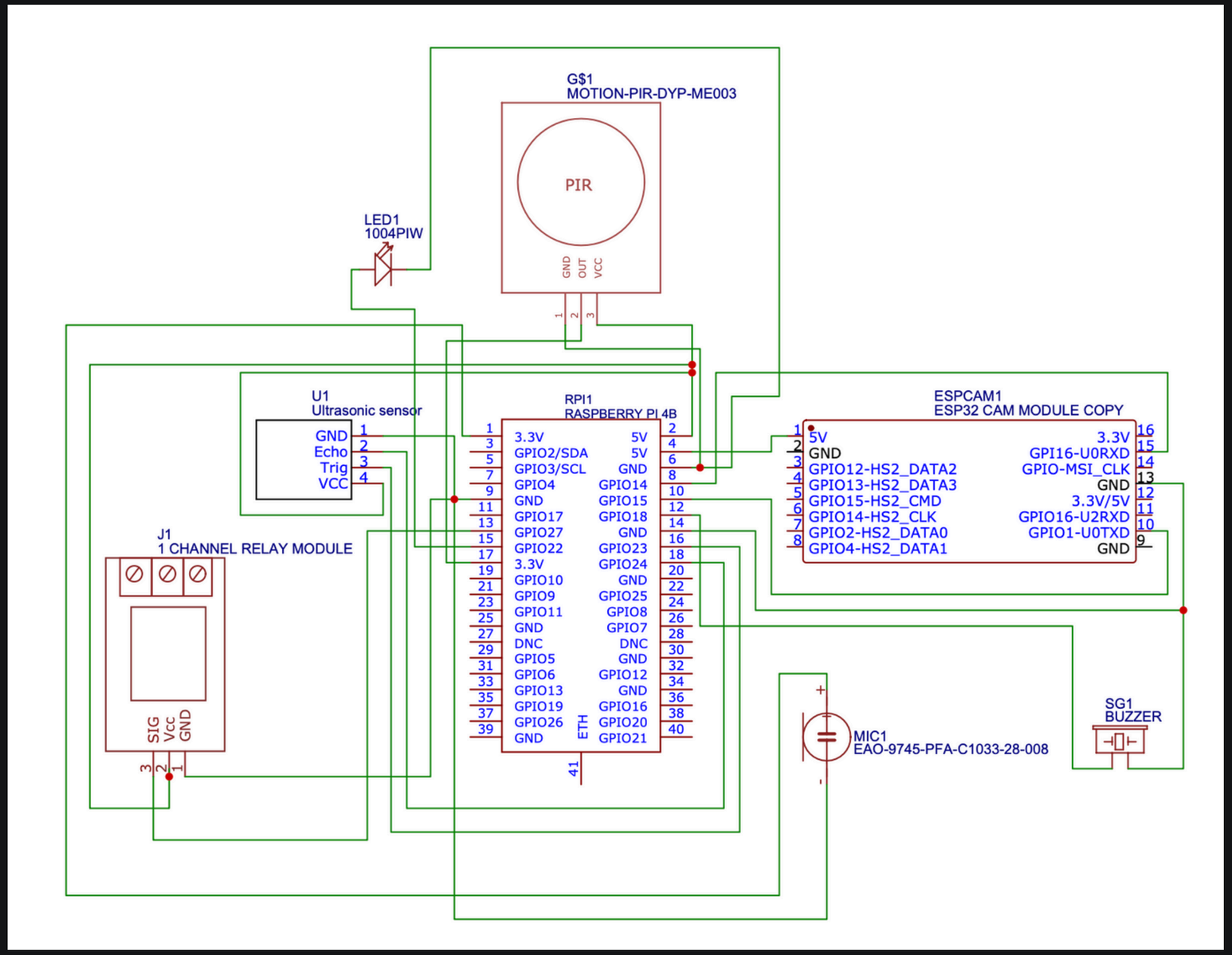
System Architecture:

- Sensors: Motion, IR, sound sensors.
- Cameras: Smart CCTV for video surveillance.
- Controller: Arduino for data processing.
- Cloud Platform: AWS for real-time storage and analysis.
- Monitoring Interface: Web or mobile app for remote access.

Data Flow:

Sensor/CCTV → Raspberry-Pi → Cloud → Monitoring App → Alerts

Architecture



Hardware Components

- Arduino: Central control and data processing unit.
- Cameras: Smart CCTV with object detection.
- Motion Sensors: Detect unauthorized movement.
- IR Sensors: Night-time surveillance.
- Sound Sensors: Detect unusual noise or alarms.

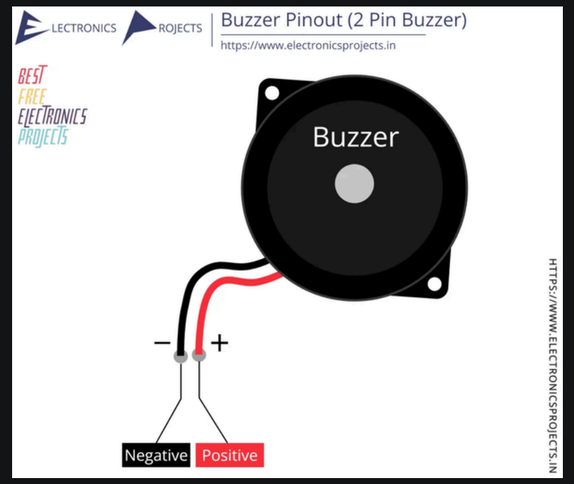
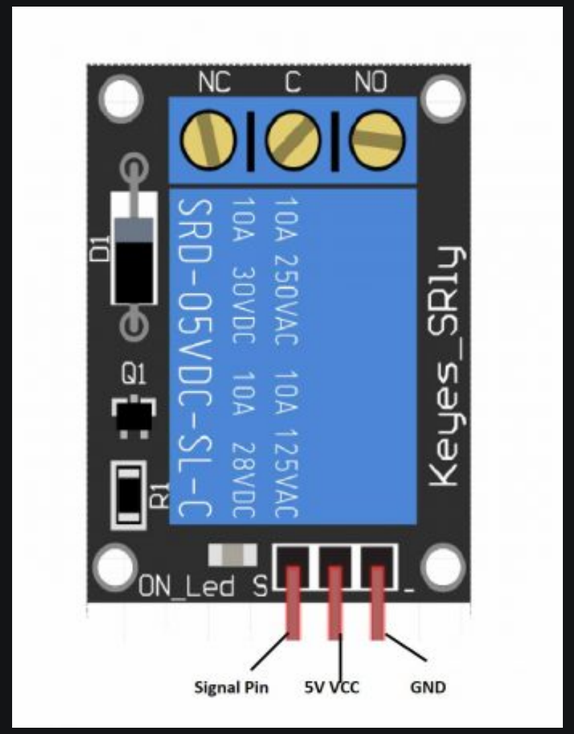
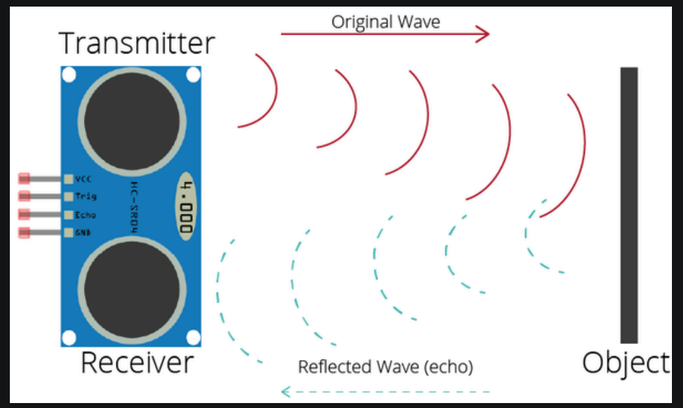
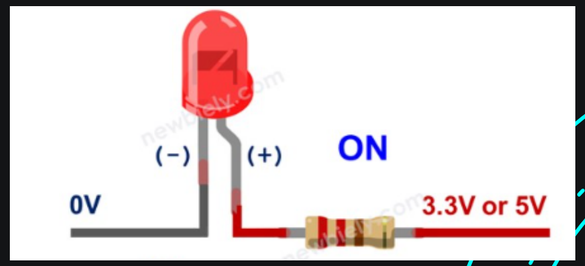


Fig 2.3.3.1 : PIR Sensor

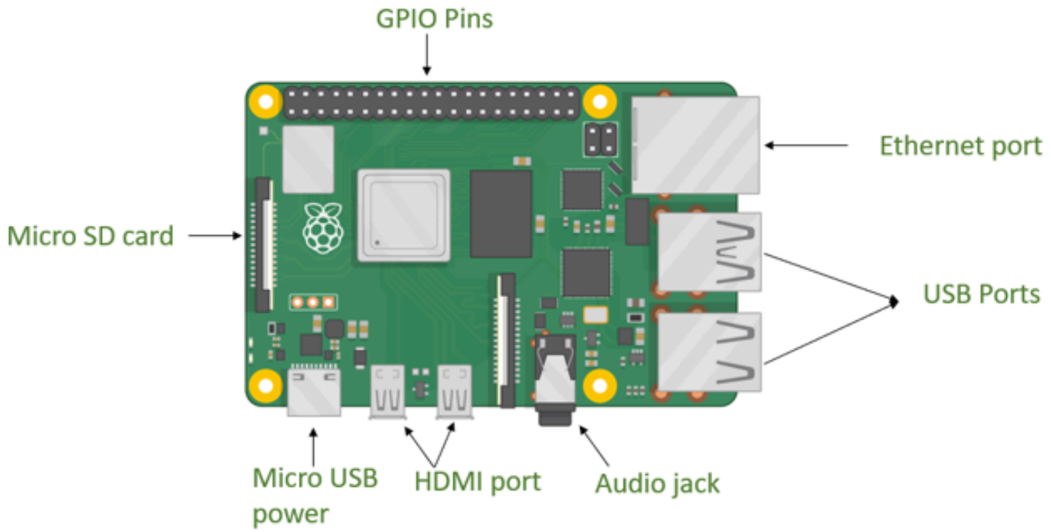
Fig 2.3.4.1 : ESP32 Camera module

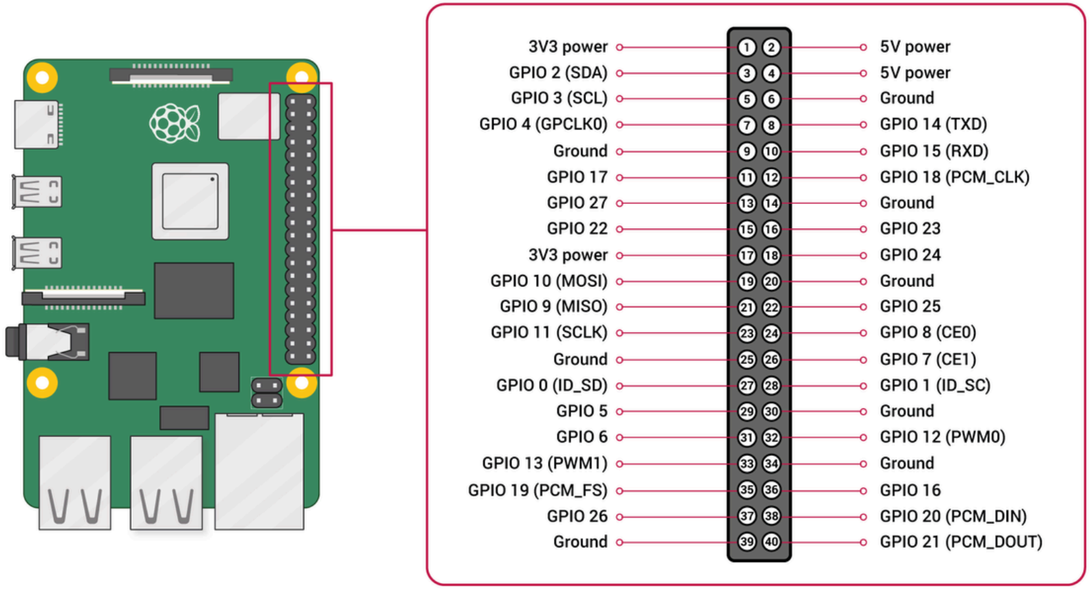






Raspberry Pi





Software Components

- Programming Language: C/C++ for Arduino scripting.
- Cloud Services: AWS IoT Core for data storage and analysis.
- Database: AWS DynamoDB for structured data storage.
- Web Application: Real-time monitoring interface.

Implementation

1

Setup:

- Connect sensors and cameras to Arduino.
- Configure Arduino scripts for data collection.
- Send data to AWS IoT Core.
- Develop web app for real-time monitoring.

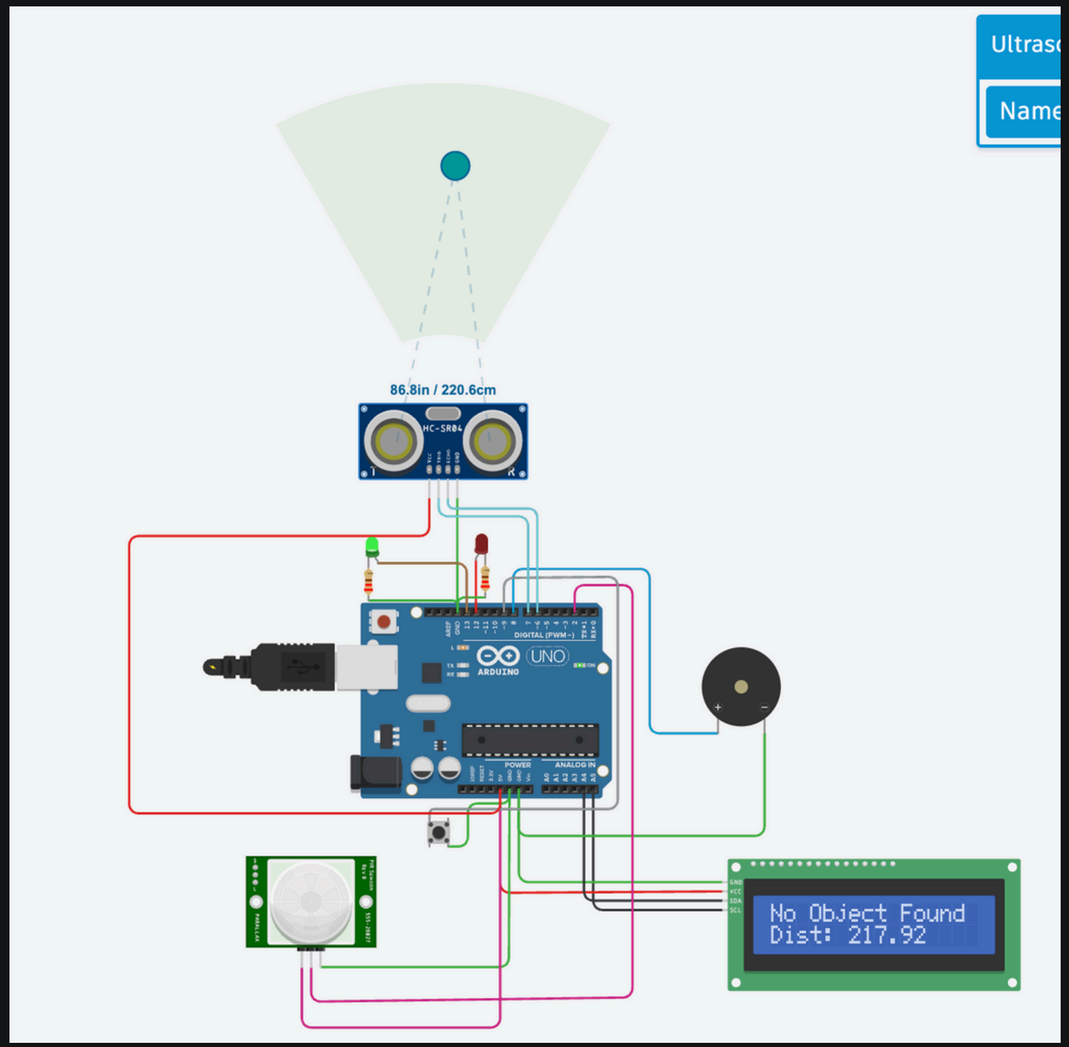
2

Real-Time Alerts:

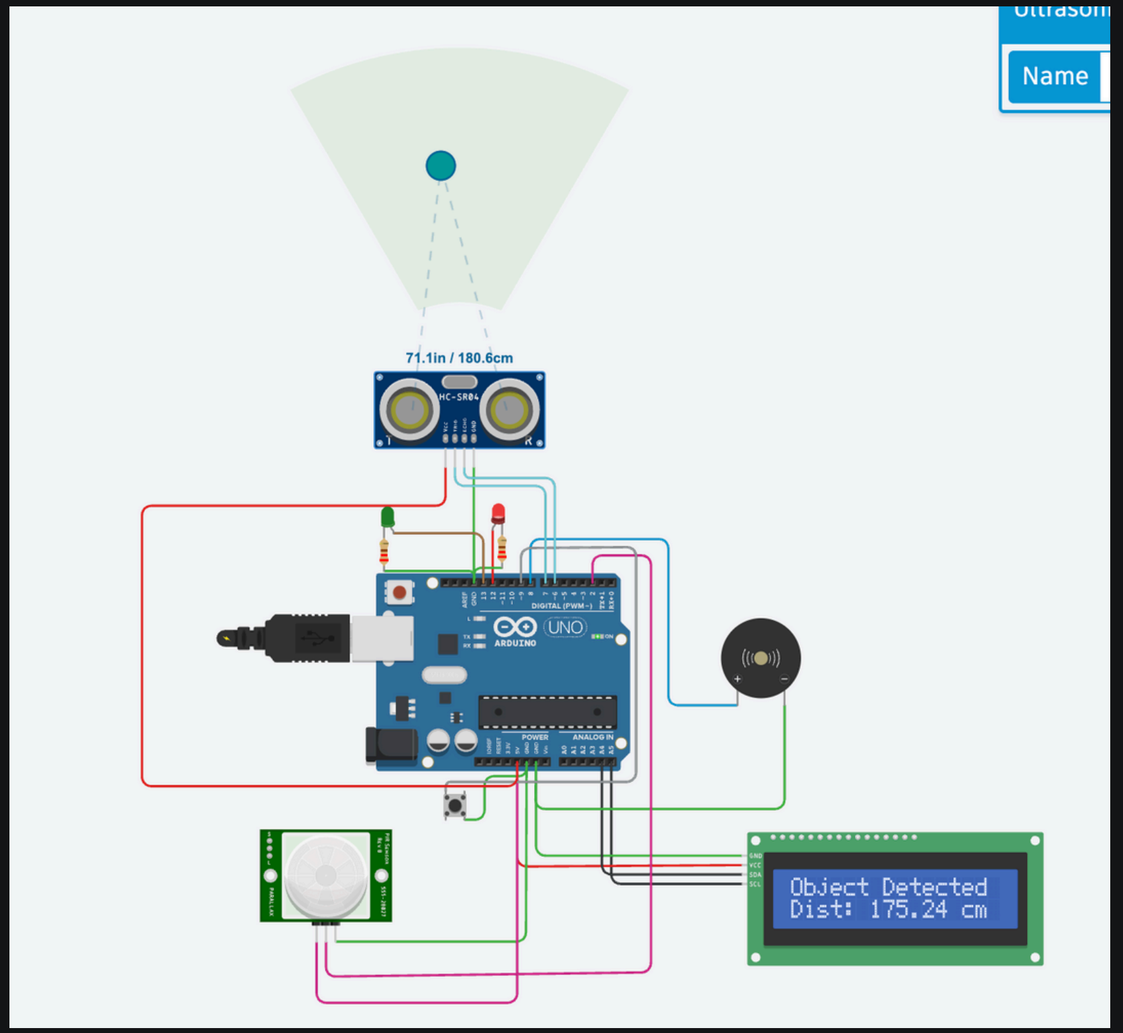
- Intruder detection.
- Unattended baggage alerts.
- Noise-based emergency detection.

Simulation

Before Object Detection



After Object Detection



Web App UI

Intrusion Detection Simulation

Upload a captured image to identify the intruder. The system will simulate identity detection using static data.

Reset Simulation

Detection Time: 11:16:42 PM

Identity Information

Name
Om Vijay Mangate

Date of Birth
22/05/2004

Age
21

Status
Unauthorized Track Access

Applications

- Intruder Detection: Identify unauthorized access.
- Suspicious Object Detection: Detect unattended luggage.
- Crowd Management: Monitor and manage crowd density.
- Messaging: Sends message to railway incharge if any of the above situations are observed.
- Cloud Storage: Would help to access the required data for future needs

Future Scope

- AI/ML Integration: Advanced threat analysis.
- Facial Recognition: Identify known offenders.
- Predictive Analytics: Manage crowd flow and prevent congestion.

Conclusion

- Enhanced Security: Real-time, automated surveillance.
- Faster Response: Immediate alerts and quicker actions.
- Scalable Solution: Suitable for various public places.

Thank You

Q&A

If anyone has any questions, feel free to ask.