TWO WAY PEOPLE COUNTER WITH AUTOMATIC LIGHTING CONTROL

A

MAJOR PROJECT-II REPORT

Submitted in partial fulfillment of the requirements

for the degree of

BACHELORS OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

By

GROUP NO. 34

Syed Ayan Ali	**0187CS201169**
Syed Mustafa Iqbal	**0187CS211174**
Syed Ahmad Ali	**0187CS211173**

Under the guidance of

Prof. Ankit Gupta

(Assistant Professor)

Department of Computer Science & Engineering
Sagar Institute of Science & Technology (SISTec), Bhopal(M.P)

Approved by AICTE, New Delhi & Govt. of M.P.
Affiliated to Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (M.P)

June - 2025

Sagar Institute of Science & Technology (SISTec), Bhopal

Department of COMPUTER SCIENCE & ENGINEERING

Bhopal (M.P.)

CERTIFICATE

We hereby certify that the work presented in the B.Tech. Major Project-I Report entitled **TWO WAY PEOPLE COUNTER WITH AUTOMATING LIGHTING SYSTEM CONTROL,** in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science & Engineering, and submitted to the Department of Computer Science & Engineering, Sagar Institute of Science & Technology (SISTec), Bhopal (M.P.), is an authentic record of our own work carried out during the period from **Jan-2025 to Jun-2025** under the supervision of **Prof. Ankit Gupta**.

The content presented in this project has not been submitted by us for the award of any other degree elsewhere.

Syed Ayan Ali	**Syed Mustafa Iqbal**	**Syed Ahmad Ali**
0187CS201169	**0187CS211173**	**0187CS211174**

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Prof. Ankit Gupta	**Dr. Amit Kumar Mishra**	**Dr. D. K. Rajoriya**
Project Guide	**HOD**	**Principal**

ACKNOWLEDGEMENT

We would like to express our sincere thanks to **Dr. D. K. Rajoriya**, **Principal, SISTec** and **Dr. Swati Saxena, Vice Principal, SISTec** Gandhi Nagar, Bhopal for giving us an opportunity to undertake this project.

.

We also take this opportunity to express a deep sense of gratitude to **Dr. Amit Kumar Mishra, HOD, Department of Computer Science & Engineering** for his kindhearted support

We extend our sincere and heartfelt thanks to our guide, **Prof. Ankit Gupta**, for providing us with the right guidance and advice at crucial junctures and for showing us the right way**.**

I am thankful to the **Project Coordinator, Prof. Deepti Jain,** who devoted her precious time in giving us the information about various aspects and gave support and guidance at every point of time.

I would like to thank all those people who helped me directly or indirectly to complete my project whenever I found myself in any issue.

TABLE OF CONTENTS

ABSTRACT

The Internet of Things (IoT) has significantly impacted numerous industries over the last decade by introducing smart, connected devices. IoT is a flexible, dynamic, and global concept that continues to evolve, and our project leverages IoT to develop a **two way people counter with automatic lighting**. This project addresses a critical need for accurate visitor statistics, widely utilized in shopping centers, academic institutions, event management, and more. By analyzing visitor flow, organizations can make informed decisions, optimize resource allocation, and even apply for grants by showcasing population metrics. Traditionally, counting methods, including beam, thermal, and video counters, have been deployed, but these methods often lack efficiency and accuracy, relying heavily on manual processes.

Our project aims to minimize human involvement by creating an automated, precise, and efficient system for tracking visitor counts. The counter system increments when someone enters a designated space and decrements upon their exit. When no visitors are present, the lighting system remains off, automatically turning on when someone enters. A 1x4 LED matrix reflects the lighting status based on visitor presence. At the core, the **ESP32 microcontroller** manages data collection and processing to accurately register entries and exits. It continuously monitors infrared sensors at the entrance, and any interference with the IR beam by an individual’s movement triggers the count.

The visitor data is transmitted to the **ThingSpeak Cloud**, an open-source platform, allowing remote monitoring of current counts, entry and exit records, and overall occupancy trends. The system also provides access to lighting status via the cloud. By eliminating human intervention at every possible stage, our project effectively minimizes errors, achieving a reliable, autonomous solution for visitor tracking.

LIST OF ABBREVIATIONS

ACRONYM	FULL FORM
IoT	Internet of Things
ESP	Esp ressif Systems
GPIO	General-Purpose Input/Output
Wi-Fi	Wireless Fidelity
DC	Direct Current
IDE	Integrated Development Environment
RPS	Relay Power Supply
VCC	Voltage Common Collector
GND	Ground
API	Application Programming Interface
UI	User Interface
LED	Light Emitting Diode

List of Figure

CHAPTER-1
INTRODUCTION

1.1 ABOUT PROJECT

This project presents a cloud-connected IoT-based visitor management system designed to monitor and manage the number of people in a specific area through a bidirectional counter, using an ESP32 microcontroller as the core processing unit. By integrating IR sensors and an automatic lighting system, the project ensures efficient power usage by automatically controlling LED lights based on the presence and count of people within the room. This system offers both functionality and energy savings, particularly suitable for smart building applications where occupancy-driven lighting and accurate visitor counts are required.

The project's primary goals include:

Visitor Count Management: To monitor entry and exit, count individuals in real-time, and adjust accordingly based on room occupancy.

Automatic Lighting Control: To activate or deactivate LED lights based on visitor presence, ensuring energy conservation and optimized lighting control.

Cloud Connectivity: To allow remote access to visitor data and control functions for the lighting system.

The key functionalities include:

Bidirectional Visitor Counting: Utilizing two IR sensors at the entrance of a room, this system detects whether a person is entering or leaving. When a visitor enters the counter increments by one, and when a visitor exits, it decrements by one[3]. This mechanism helps to maintain an accurate count of individuals present in the area.

Automatic Lighting Control: LED lights are connected via a relay module, enabling the system to automatically turn on the lights when visitors are detected and off when the room is empty. This feature minimizes unnecessary power usage by deactivating lights when no one is present.

Real-time Monitoring and Cloud Access: The project leverages the ESP32's Wi-Fi capability to send data to a cloud-based platform, allowing real-time monitoring of visitor counts.

1.2 PROJECT OBJECTIVE

The primary objective of this project is to create an efficient, automated visitor management system with bidirectional counting and automatic lighting control. This IoT-based solution aims to enhance energy efficiency, improve space utilization, and provide accurate real-time occupancy data for various environments such as offices, classrooms, and public spaces.

Automate Visitor Counting: Develop a reliable system to automatically track the number of people entering and exiting a room, ensuring an accurate count at any given time.

Energy Efficiency: Reduce energy consumption by automatically controlling the room's lighting based on occupancy, turning lights on when people are present and off when the room is empty.

Real-Time Monitoring: Provide a real-time display of the visitor count, allowing easy monitoring of room occupancy for better space utilization and management.

User-Friendly Interface: Create a simple and intuitive interface for displaying visitor count and lighting status, ensuring ease of use and accessibility.

Versatile Application: Design a scalable solution that can be implemented in various environments, such as offices, classrooms, and public spaces, to enhance automation and energy management.

Cost Reduction: Lower operational costs by saving energy, which in turn reduces electricity bills and environmental impact.

1.2.1 FUNCTIONALITY

Bidirectional Visitor Counting: Accurately counts visitors entering and exiting through the entrance by detecting the order in which two sensors are triggered. This ensures a correct count for both incoming and outgoing people.

Automatic Lighting Control: Controls room lighting based on occupancy. The light automatically turns on when the visitor count is above zero (indicating the room is occupied) and turns off when the count returns to zero.

Real-Time Display of Visitor Count: Shows the current count of people in the room on an LCD or LED display, providing a clear view of room occupancy in real-time.

Power Management: Conserves energy by turning off lights when the room is empty, reducing unnecessary power usage and supporting eco-friendly practices.

User-Friendly Interface: Provides a simple interface on the display and through any web or mobile integration, making it easy to check occupancy and light status.

1.2.2 INTERFACE

The interface for the Bidirectional Visitor Counter with Lighting Control is designed to be user-friendly, intuitive, and provide real-time information on room occupancy and lighting status[8]. Here's an outline of the key elements of the interface:

Real-Time Visitor Counter Display: A prominent digital display (e.g., LCD or LED) shows the current visitor count, updating in real-time as people enter and exit the room

Lighting Status Indicator: An indicator (such as a light icon or color-coded signal) shows whether the room's lighting is on or off. For instance, a green light icon may indicate that lights are on, while a red icon shows they are off.

1.2.3 DESIGN AND IMPLEMENTATION CONSTRAINTS

Sensor Range and Accuracy: The sensors must accurately detect people moving in both directions without missing or falsely counting entries/exits. Limited sensor range or misalignment could affect accuracy.

Power Supply Requirements: The system must have a stable power source to ensure continuous operation. Any power instability could interrupt counting and lighting control.

Space for Sensor Placement: Limited space around entrances may restrict optimal sensor placement, affecting the accuracy of detecting entry/exit sequences.

Real-Time Processing: The system must process sensor inputs in real-time to accurately update visitor counts and control the lighting. Any delays in processing could cause inaccuracies.

Expansion for Multi-Room Support:

If the system needs to support multiple rooms, it may require additional sensors and separate microcontrollers for each entrance.

Maintenance Accessibility: Regular access to the system for calibration and repairs should be feasible without disrupting daily operations, which may be challenging in high-traffic areas.

These constraints should be considered during the design and implementation phases to ensure the system's reliability, accuracy, and usability. Each constraint highlights potential challenges and factors that may influence design decisions and system performance.

1.2.4 ASSUMPTIONS AND DEPENDENCIES

The following are the assumptions and dependencies of the project:

Sensor Accuracy and Reliability: It is assumed that the sensors used (e.g., IR or ultrasonic) will accurately detect people entering and exiting without significant error or false positives/negatives.

Constant Power Supply: The system assumes access to a continuous power source, as any power interruptions would disrupt the visitor counting and lighting control functions.

Stable Power Source: The system relies on a stable power supply to operate continuously, as any power failure would disrupt the system and could cause loss of count data.

Sensor Quality and Compatibility: The system's performance depends on the quality and compatibility of the sensors with the microcontroller, as the sensors are critical for accurate visit.

CHAPTER-2
SOFTWARE & HARDWARE REQUIREMENTS

2.1 INTRODUCTION

The successful development and implementation of the Bidirectional Visitor Counter rely on both software and hardware components. These components work together to ensure the proper functioning of the system.

2.2 SOFTWARE REQUIREMENTS

For Users:

ThinkSpeak Cloud

An IoT cloud platform for real-time data monitoring and visualization.

Internet Connectivity

Required for the ESP32 to communicate with the Think Speak and for remote access to the project.

For Developers:

Arduino IDE: For writing and uploading code to the ESP32.

ESP32 Board Support Package: Enables programming of the ESP32 in the Arduino IDE.

Sensor Interface Libraries: Appropriate libraries for interfacing sensors with the ESP32.

Wi-Fi Library: To connect the ESP32 to a Wi-Fi network for remote control features

2.3 HARDWARE REQUIREMENTS

ESP32 WROOM-32: Acts as the main controller for the project.

2-Channel Relay Module: Controls the LEDs based on signals from the ESP32.

Display Module: Optional display (e.g., LCD) to show the real-time count locally.

DHT11 Sensors. Infrared (IR) sensors used for detecting the movement.

Communication Module: Wi-Fi module for transmitting data to a cloud server or app.

Power Supply: A stable 12v power source for the LED bulb.

CHAPTER-3
PROBLEM DESCRIPTION

3.1 OVERVIEW

In today’s fast-paced and highly dynamic environments, the need to manage and monitor visitor flow within various facilities has become increasingly critical. Effective visitor management is essential for ensuring operational efficiency, maintaining security, adhering to safety protocols, and delivering a smooth, seamless experience for all occupants. High-traffic areas such as shopping malls, office buildings, airports, museums, hospitals, and entertainment venues are especially vulnerable to inefficiencies in visitor tracking, and these inefficiencies can significantly hinder space management, customer satisfaction, and overall operational performance.

THE CHALLENGES OF TRADITIONAL VISITOR COUNTING METHODS

Traditional methods for tracking visitor flow, such as manual counting by staff or the use of basic one-way counters, have inherent limitations that reduce their effectiveness. Manual counting is highly prone to human error, as employees can easily lose count, especially during peak hours when hundreds or thousands of people may pass through an entry or exit point in a short period of time. This leads to unreliable data that cannot be used effectively for resource management, occupancy control, or strategic decision-making.

Basic one-way counters, which are often used in place of manual counting, also fail to provide comprehensive visitor tracking. These systems are generally designed to track only the total number of people passing through a single entrance or exit, without distinguishing between those entering or leaving the premises. As a result, these systems often provide skewed data that either overestimates or underestimates the actual number of people inside a facility at any given time. This issue is particularly problematic in large spaces with multiple entrances and exits, as well as during events or peak periods when high volumes of visitors move through the space in quick succession.

THE IMPORTANCE OF REAL-TIME AND BIDIRECTIONAL VISITOR TRACKING

Accurate, real-time visitor tracking is not just a convenience it’s a necessity, especially in environments where safety, security, and optimal space management are priorities. Facility managers need to know exactly how many people are inside a building or specific area at any given moment to ensure compliance with safety regulations, such as fire codes or maximum occupancy limits. Additionally, during emergency situations, knowing the number and location of occupants within a facility can help authorities respond more effectively, ensuring the safety of everyone on-site.

KEY FEATURES AND FUNCTIONALITY:

Bidirectional Counting Capability: Unlike traditional systems that can only track total foot traffic in one direction, this solution can count both people entering and leaving a facility. This ensures accurate data on the number of people inside a space at any given time, preventing overestimation or underestimation of occupancy levels.

Real-Time Data Monitoring: The system provides real-time updates on visitor flow and occupancy levels, allowing facility managers to respond promptly to changes in traffic patterns. Whether it's adjusting staffing levels, managing queues, or addressing safety concerns, the ability to access real-time data is crucial for maintaining operational efficiency.

Advanced Sensor Technology: The Bidirectional Visitor Counter utilizes sophisticated sensors to detect the movement of individuals with a high degree of accuracy. These sensors can differentiate between individuals entering and exiting, even in crowded conditions, ensuring precise counting regardless of the level of foot traffic.

Scalability and Flexibility: The Bidirectional Visitor Counter is scalable to accommodate spaces of different sizes and complexity. Whether it’s a small retail store or a large airport terminal with multiple entry points, the system can be customized to meet the specific needs of the facility.

CHAPTER-4
LITERATURE SURVEY

4.1 INTRODUCTION

Accurate tracking of visitors entering and exiting a space is critical for facility management, security, and safety. Various methods have been developed over the years to automate this process, ranging from simple mechanical counters to advanced IoT-based systems[3]. This section reviews the current state of visitor counting technologies, with a focus on bidirectional counting, sensor integration, and IoT solutions. It highlights the strengths, limitations, and opportunities for improvement in existing systems, providing a foundation for the development of a more efficient, accurate, and scalable visitor counting solution.

4.1.1 Existing Smart Control Systems

The existing smart control systems provide various solutions for visitor tracking, each with its own set of advantages and limitations. The goal of developing a **Bidirectional Visitor Counter[1]** system is to combine the best features of these technologies, offering a more reliable and scalable solution for real-time, accurate counting of people entering and exiting spaces

4.1.2 Infrared (IR) Sensor-Based Systems

Functionality: Detects the presence of people by measuring interruptions in an infrared beam.
Limitations: Does not distinguish entry from exit, leading to potential inaccuracies in tracking movement direction.

4.1.3 Ultrasonic Sensor-Based Systems

Functionality: Uses sound waves to detect the distance and movement of people, providing more accurate direction detection than IR sensors.
Limitations: Sensitive to environmental factors like temperature and obstructions, affecting accuracy.

4.1.4 IoT-Based Visitor Counting Systems

Functionality: Integrates sensors with IoT platforms to provide real-time monitoring and data analysis over the cloud.
Advantages: Offers remote access, data logging, and real-time alerts for efficient visitor management.

4.1.5 Technologies Used

The implementation of smart control systems relies on various technologies, including[11]:

Microcontrollers: Devices like the ESP32 and Arduino serve as the backbone of smart control systems, enabling connectivity and control of multiple components.

Sensors: IR sensors detect the presence of people by emitting an infrared beam and measuring interruptions in the beam as individuals pass through.

Web Applications: Platforms like the Think Speak app facilitate user interaction with showing the real-time data.

Communication Protocols: Wi-Fi, Bluetooth, and Zigbee enable seamless communication between devices, ensuring reliable operation and user access.

4.1.6 Addressing Drawbacks of Conventional Approaches

By addressing the shortcomings of conventional approaches with these modern solutions, the bidirectional visitor counting system can provide greater accuracy, reliability, scalability, and operational efficiency[5].

Manual Operation: Traditional visitor counting systems often require manual intervention for calibration, data recording, and monitoring, leading to inefficiencies and human errors.

Lack of Automation: Conventional systems often rely on manual counting or basic mechanical processes, resulting in delays, inaccuracies, and limited real-time monitoring.

Limited Accessibility: Traditional visitor counting systems often lack remote access, restricting real-time monitoring and data retrieval, making it difficult to manage from multiple locations.

4.1.7 Emerging Trends in Visitor Counting Systems

In this section, we explore the latest trends and innovations in visitor counting technology, particularly those influencing the development of next-generation systems[8].

4.1.8 Integration with Smart Building Systems

Overview: Visitor counting systems are increasingly being integrated with broader smart building technologies, including lighting, HVAC systems, and security[2].
Benefits: This integration allows for more efficient resource management, enhanced safety, and a better overall user experience. For instance, when occupancy levels exceed a certain threshold, lighting or HVAC systems can automatically adjust to save energy.

4.1.9 Use of Artificial Intelligence (AI) and Machine Learning

Overview: AI and machine learning algorithms are being utilized to improve the accuracy and predictive capabilities of visitor counting systems.
Applications: These technologies can help predict peak visitor times, analyze foot traffic patterns, and optimize space utilization in real-time.

CHAPTER-5
SOFTWARE REQUIREMENT SPECIFICATION

5.1 FUNCTIONAL REQUIREMENTS

5.1.1 VISITOR COUNTING

The system must accurately count the number of people entering and exiting through a specific entry/exit point.
It should distinguish between inbound (entering) and outbound (exiting) visitors, ensuring bidirectional tracking.

5.1.2 REAL-TIME DATA PROCESSING

The system should provide real-time visitor counts, immediately reflecting any changes as visitors pass through the entry/exit point.
It must be able to continuously monitor visitor traffic without delays or data loss.

5.1.3 SENSOR INTEGRATION

The system must integrate with infrared, ultrasonic, or other suitable sensors for detecting movement and direction.
The sensors should have the capability to accurately track both entry and exit, even when multiple individuals pass through simultaneously.

5.1.4 CLOUD CONNECTIVITY

The system should be connected to a cloud-based platform for remote monitoring, data storage, and analytics.

5.1.5 USER INTERFACE

The system must have a user-friendly interface that displays real-time visitor data.
The interface should allow users to easily view and interpret visitor statistics, and provide visual alerts when occupancy limits are reached.

5.2 NON-FUNCTIONAL REQUIREMENTS

Security: Data privacy regulations (e.g., GDPR) must be adhered to, ensuring that any personal or sensitive data collected is securely handled.

Scalability: The system should be scalable to accommodate the growing needs of larger facilities. It should support the addition of new entry/exit points without requiring major redesigns or disruptions.

Performance: The system must be capable of processing visitor data in real-time with minimal delay, providing instant updates as visitors pass through entry/exit points.

Usability: The system should feature an intuitive and user-friendly interface, allowing facility managers to monitor visitor counts and access reports easily without requiring advanced technical knowledge.

Reliability: The system must operate continuously with minimal downtime.

Maintainability: The system should be designed for easy maintenance and updates.

CHAPTER 6
SOFTWARE DESIGN

6.1 USE CASE DIAGRAM

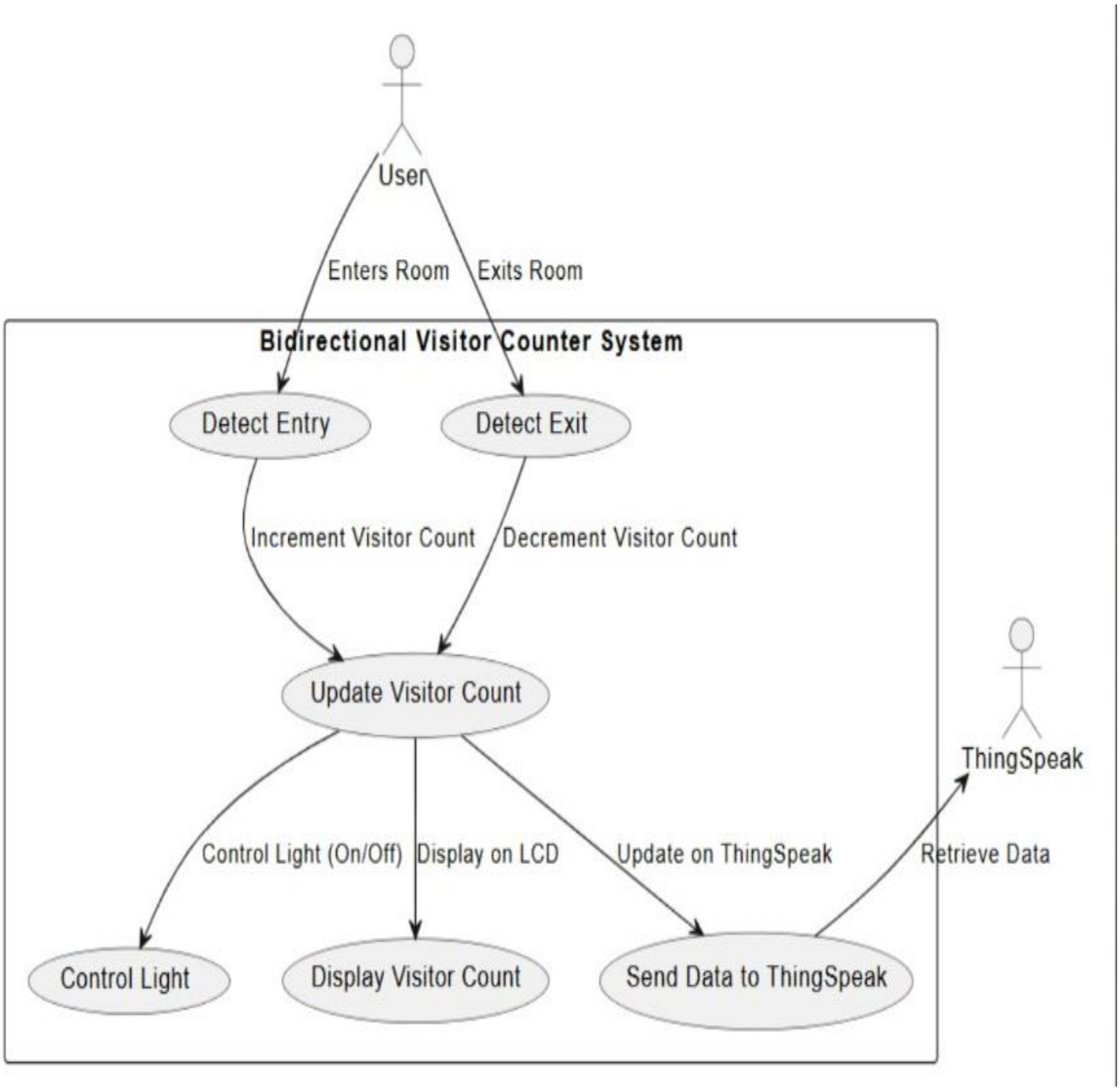

Figure 6.1 Use Case Diagram

6.2 ARCHITECTURE

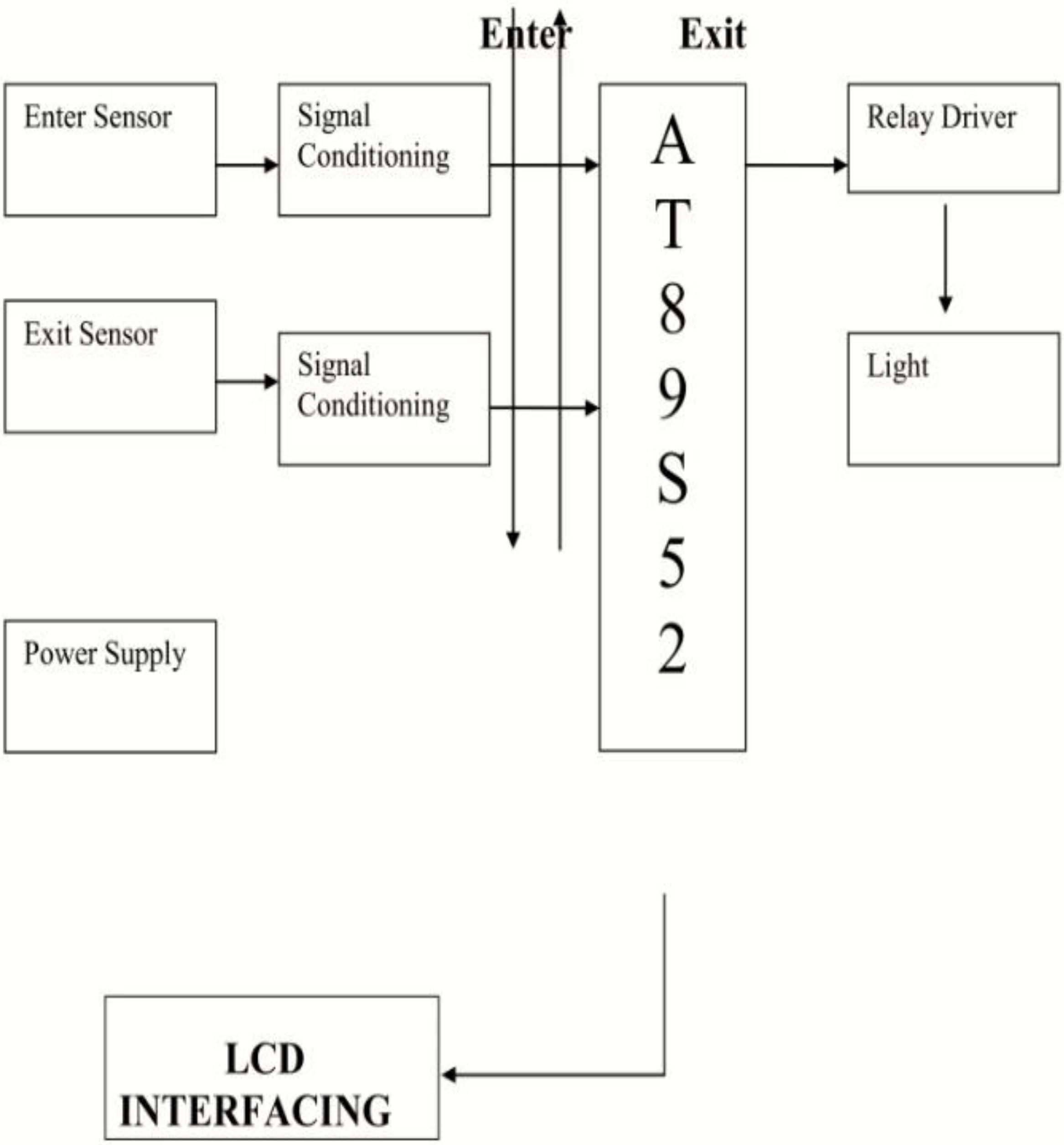

Figure 6.2 Architecture

6.3 CIRCUIT DIAGRAM

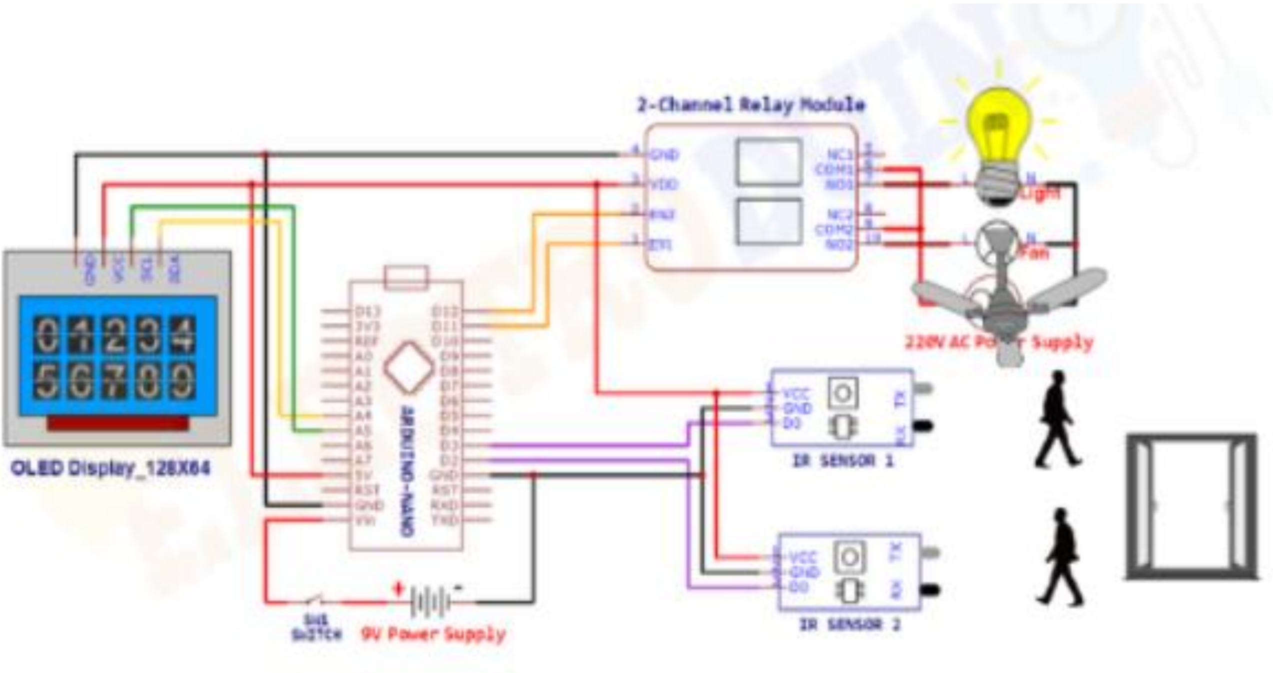

Figure 6.3 Circuit Diagram

6.4 PIN DIAGRAM OF MICROCONTROLLER

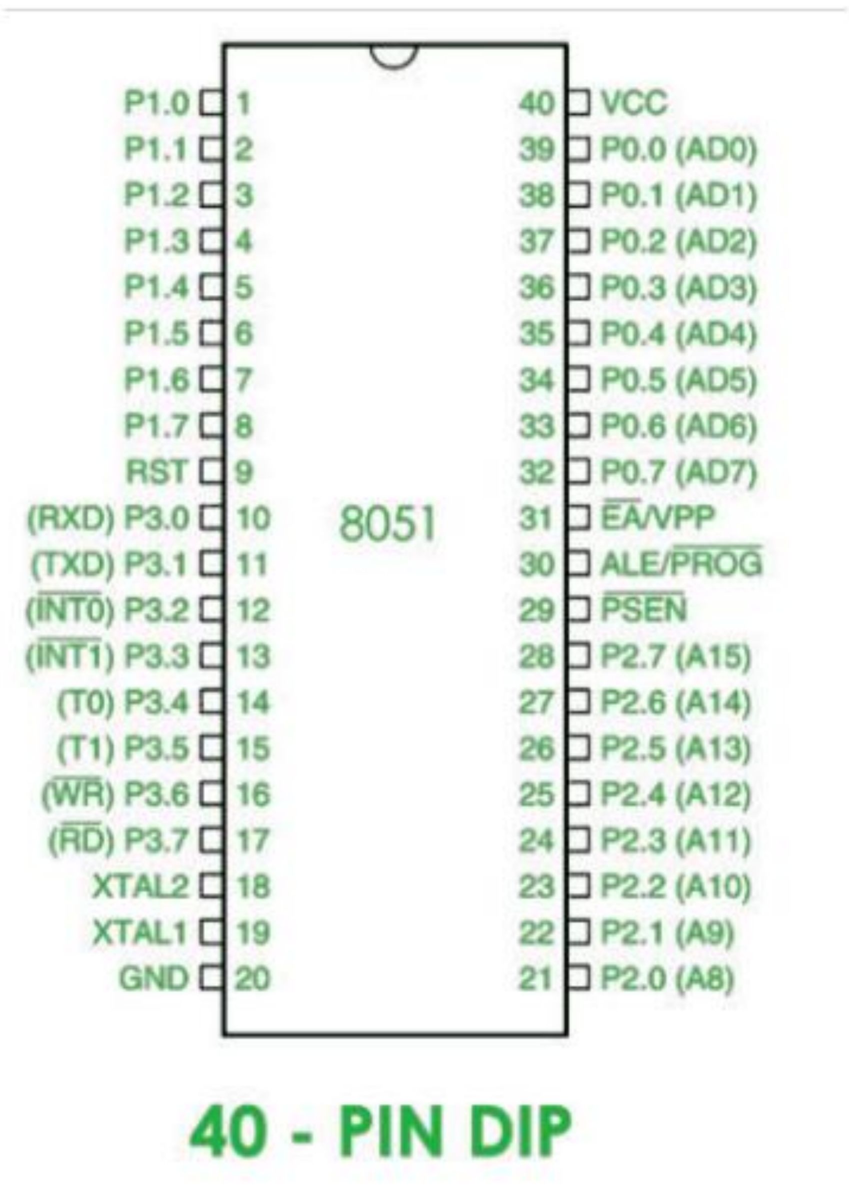

Figure 6.4 Pin Diagram of Microcontroller

CHAPTER-7
IoT MODULE

7.1 PRE-PROCESSING STEPS

In an Internet of Things (IoT)-based system, pre-processing is an essential phase that prepares raw sensor data for further analysis, visualization, and decision-making. Pre-processing ensures that the data fed into the system is accurate, relevant, and optimized for the desired output. For a Bidirectional Visitor Counting System, pre-processing involves several steps to filter out noise, handle anomalies, and normalize data for real-time processing.

7.1.1 DATA COLLECTION

The first step in pre-processing is data collection, which occurs via sensors placed at strategic points in the facility to monitor people's movement. Commonly used sensors include:

Infrared (IR) sensors: Detect presence based on interruption in infrared beams.

Ultrasonic sensors: Measure distance to objects and detect movement.

Camera-based systems: Capture visual data that can be processed using image recognition algorithms.

The IoT module connects these sensors to a central microcontroller (e.g., ESP32, Arduino) via wireless communication protocols (such as Wi-Fi, Bluetooth, or Zigbee), enabling real-time data streaming to a cloud-based platform or server.

7.1.2 DATA FILTERING AND NOISE REMOVAL

Sensor data, especially in environments with fluctuating environmental conditions (e.g., lighting, temperature, or motion interference), can often contain noise. Noise refers to irrelevant or erroneous data that can distort the results. For example:

False positives: An IR sensor might register a movement due to a passing shadow or nearby heat source.

Data spikes: Ultrasonic sensors might report an unusually high or low distance due to obstruction or improper calibration.

Data filtering techniques are applied to eliminate or minimize this noise. Methods include:

Smoothing algorithms: Techniques like moving averages or Gaussian filters help reduce random fluctuations in the data.

Thresholding: Setting a minimum or maximum value for the data to discard extreme or implausible readings.

Debouncing: This is used with motion sensors to avoid counting multiple signals from a single event (e.g., a visitor walking slowly and triggering multiple sensor detections).

7.1.3 DATA NORMALIZATION

After filtering, the data needs to be normalized to ensure consistency across sensors and formats. For example, sensor data from different devices (e.g., IR and ultrasonic) may have different units (e.g., distance in meters, detection in binary). Normalization techniques, such as scaling the data to a common range (e.g., 0-1), allow the system to process the data uniformly.

Standardization: Ensuring that all inputs fall within a specified range helps in the integration of different sensor types into a unified system.

Temporal Synchronization: Ensures that timestamps on data are synchronized between sensors, which is crucial for accurate bidirectional counting. For instance, if an entry sensor detects a person's presence but the exit sensor fails to register their departure, the system may inaccurately track the visitor count.

7.1.4 EVENT DETECTION AND PRE-PROCESSING ALGORITHMS

After basic noise removal and normalization, the data is processed to detect specific events or activities (e.g., a visitor entering or exiting). For example, an IR sensor might output a series of data points, which need to be analyzed to determine if an individual entered or exited. Event detection involves:

Change detection algorithms: Identifying when a significant change (e.g., an interruption in the sensor beam) occurs that corresponds to an event.

Rule-based logic: For bidirectional counting, an algorithm may track the order of sensor activations (e.g., if an entry sensor detects motion before an exit sensor, the system counts it as an entry).

7.2 DATA VISUALIZATION

Data visualization is an essential component of IoT-based systems as it enables users to easily understand complex data through graphical representations. In the case of a Bidirectional Visitor Counting System, visualization aids in monitoring real-time visitor flow, detecting anomalies, and managing facility operations.

7.2.1 REAL-TIME DASHBOARD

A real-time dashboard is typically used to visualize the count of visitors entering and exiting a space. It displays up-to-the-minute data, offering facility managers a live view of occupancy levels. Key features of a real-time dashboard include:

Entry/Exit Counters: Real-time visualization of people entering and leaving, often shown as separate counters for entry and exit.

Occupancy Levels: A dynamic graph or gauge showing current occupancy relative to the space's capacity.

Alerts and Notifications: Automated alerts when occupancy thresholds are exceeded, which can trigger actions like adjusting HVAC settings or notifying security personnel.

7.2.2 HISTORICAL DATA ANALYSIS

Visualization is not limited to real-time data; it also includes historical analysis, which can be crucial for planning and optimizing facility usage. A historical data dashboard can show:

Visitor Trends Over Time: Line or bar graphs showing visitor trends over days, weeks, or months.

Peak Usage Hours: Heatmaps or bar charts highlighting times of high visitor volume, helping facility managers optimize staffing or resources

Visitor Demographics (if applicable): If integrated with other systems (e.g., facial recognition), visualizing visitor demographics such as age, gender, or purpose of visit can offer deeper insights into space utilization.

7.2.3 HEATMAPS AND SPACE UTILIZATION

A powerful visualization tool for IoT systems is the heatmap, which represents the flow of visitors across a facility. Heatmaps help in identifying areas with high or low visitor traffic, enabling managers to optimize the layout, staffing, or facility design. In a bidirectional system:

Entry/Exit Heatmaps: Visualizing where visitors are entering or exiting most frequently within a space, helping with crowd management and safety planning.

Space Utilization: Overlaying the entry/exit data on a floor plan can show how various zones of a building are used throughout the day, helping in optimizing resource allocation and improving customer experience.

7.2.4 CLOUD-BASED PLATFORMS

Cloud platforms such as ThingSpeak, Google Cloud IoT, and AWS IoT provide robust environments for data storage, visualization, and sharing. They allow the integration of multiple sensors, centralized data storage, and advanced analytics. A cloud dashboard can be accessed remotely by managers to monitor visitor counts and manage operations from anywhere.

7.3 IOT MODEL DESCRIPTION

The IoT model for a Bidirectional Visitor Counting:

System consists of several layers working together to ensure smooth data flow and processing. These layers include sensing, data processing, communication, and cloud-based storage.

7.3.1 SENSOR LAYER

The **sensor layer** involves various sensors like IR sensors, ultrasonic sensors, and sometimes cameras. These devices capture data related to the movement of individuals. In a bidirectional visitor counting system, sensors are strategically placed at entry and exit points to monitor the flow of visitors in both directions.

IR Sensors: Detect presence by emitting infrared beams, which are interrupted when a person walks through the sensor zone.

Ultrasonic Sensors: Measure distance and movement patterns, often used to determine the direction of movement more accurately.

Cameras (Optional): Provide visual data for advanced analytics, such as detecting if an individual is entering or exiting based on their position in the frame.

7.3.2 DATA PROCESSING LAYER

Once the data is collected from the sensors, it passes to a data processing layer, often hosted on an embedded microcontroller like an ESP32 or Arduino. This layer handles:

Sensor Data Aggregation: Combining the data from multiple sensors into a single stream for processing.

Event Detection and Filtering: Applying the pre-processing steps (noise filtering, normalization) and identifying specific events (entry/exit).

Decision Making: Based on the processed data, the system decides whether to increase or decrease the visitor count.

7.3.3 COMMUNICATION LAYER

The communication layer ensures that data collected by the sensors is transmitted to the cloud or server. This is typically achieved via:

Wi-Fi or Bluetooth: These are the most common communication protocols used to transmit data from local sensors to a central hub or cloud platform.

Zigbee or LoRa: For environments where low-power, long-range communication is needed, these protocols may be used.

7.3.4 CLOUD-BASED STORAGE AND ANALYTICS

The data is uploaded to a cloud platform, such as **AWS IoT**, **Microsoft Azure IoT**, or **Google Cloud IoT**, for storage and advanced analysis. The cloud platform:

Stores Historical Data: Ensures all sensor data is archived for future analysis and reference.

Performs Advanced Analytics: Provides tools for deeper insights, such as anomaly detection, predictive analytics, and historical trend analysis.

7.4 RESULT ANALYSIS

Once the system has collected and processed data, it's essential to analyze the results to draw meaningful conclusions. This analysis helps identify trends, optimize operations, and improve the overall visitor experience.

7.4.1 ACCURACY OF VISITOR COUNT

One of the primary results of the IoT-based system is the accuracy of visitor counts. Comparing real-time data from sensors with manual counting or other traditional methods allows for assessing the **accuracy and reliability** of the system. For instance:

Error Rates: Calculating the error rate of the system by comparing the predicted count with actual counts.

False Positives/Negatives: Analyzing the occurrence of false entries or exits due to sensor errors.

7.4.2 VISITOR FLOW OPTIMIZATION

Analyzing the flow of visitors over time can help identify bottlenecks or congestion areas. For example, if certain entry or exit points consistently show high traffic, the facility can optimize these areas to manage crowd flow better.

CHAPTER-8
CODING

8.1 SOURCE CODE

```
#include<WiFi.h>
#include<WifiClient.h>
#include<HTTPClient.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

int counter = 0;
LiquidCrystal_I2C lcd(0x27, 16, 2);

WiFiClient client;
HTTPClient http;

String url;
String API = "4NPWBG0ZKXKPCZBT";
int fieldno = 1;

#define inSensor 18 //D5
#define outSensor 19 //D6

int inStatus;
int outStatus;

int countin = 0;
int countout = 0;

int in;
int out;
```

```
int now;
int backlightPin = 9;
#define relay 2 //D3

unsigned long inSensorDebounce = 0;    // To track debounce time for inSensor
unsigned long outSensorDebounce = 0;   // To track debounce time for outSensor
unsigned long debounceDelay = 2000;    // 1 second debounce delay
void connectToWifi(void);

void setup() {

  Wire.begin(21, 22);
  Serial.begin(115200);

  connectToWifi();

  lcd.begin(16, 2);
  lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print("Initializing..");
  delay(2000);
  lcd.clear();

  pinMode(inSensor, INPUT);
  pinMode(outSensor, INPUT);
  pinMode(2, OUTPUT);
  digitalWrite(2, HIGH);

  pinMode(backlightPin, OUTPUT);
  analogWrite(backlightPin, 128);

  Serial.println("Visitor Counter Demo");
```

```
  delay(1000);

  // Turn on the backlight
}

int httpcode;
void loop() {
  SendGetRequest(now);
  delay(1000);

unsigned long currentMillis = millis();  // Get the current time

  inStatus = digitalRead(inSensor);
  outStatus = digitalRead(outSensor);

  // If inSensor is triggered and debounce time has passed
  if (inStatus == 0 && (currentMillis - inSensorDebounce) > debounceDelay) {
    countin++;
    in = countin;
    inSensorDebounce = currentMillis;  // Reset debounce time
    outSensorDebounce = currentMillis; // Disable outSensor for debounceDelay time
  }

  // If outSensor is triggered and debounce time has passed
  if (outStatus == 0 && (currentMillis - outSensorDebounce) > debounceDelay) {
    countout++;
    out = countout;
    outSensorDebounce = currentMillis;  // Reset debounce time
    inSensorDebounce = currentMillis;   // Disable inSensor for debounceDelay time
  }

  now = in - out;
```

```
   // Ensure now doesn't go below zero
  if (now < 0) {
    now = 0;
  }

  Serial.println("Current Count: " + String(now));
  if (now <= 0) {
    digitalWrite(2, LOW); // Turn off light
    Serial.println("No Visitors! Light Off");
  } else {
    digitalWrite(2, HIGH); // Turn on light
    Serial.println("Visitors Present! Light On");
  }

  lcd.setCursor(0, 0);          // Set the cursor to the first column and first row
  lcd.print("Total person");    // Print some text
  lcd.setCursor(0, 1);
  lcd.print(now);
  delay(1000);
  lcd.clear();
}
void SendGetRequest(int data){
  url = "http://api.thingspeak.com/update?api_key=";
  url = url + API;
  url = url + "&field";
  url = url + fieldno;
  url = url + "=";
  url = url + data;

  http.begin(client, url); // HTTP request
  Serial.println("waiting for response...");
```

```
  httpcode = http.GET();
  if (httpcode > 0) {
    Serial.println("Data sent successfully");
  } else {
    Serial.println("Failed to send data");
  }
  http.end();
}
void connectToWifi() {
  WiFi.mode(WIFI_STA);
  WiFi.begin("POCO X4 Pro 5G", "123445678");
  Serial.print("Connecting to WiFi");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".....");
    delay(200);
  }
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());
  Serial.print("Mac Address: ");
  Serial.println(WiFi.macAddress());
}
```

CHAPTER-9
RESULT AND OUTPUT SCREENS

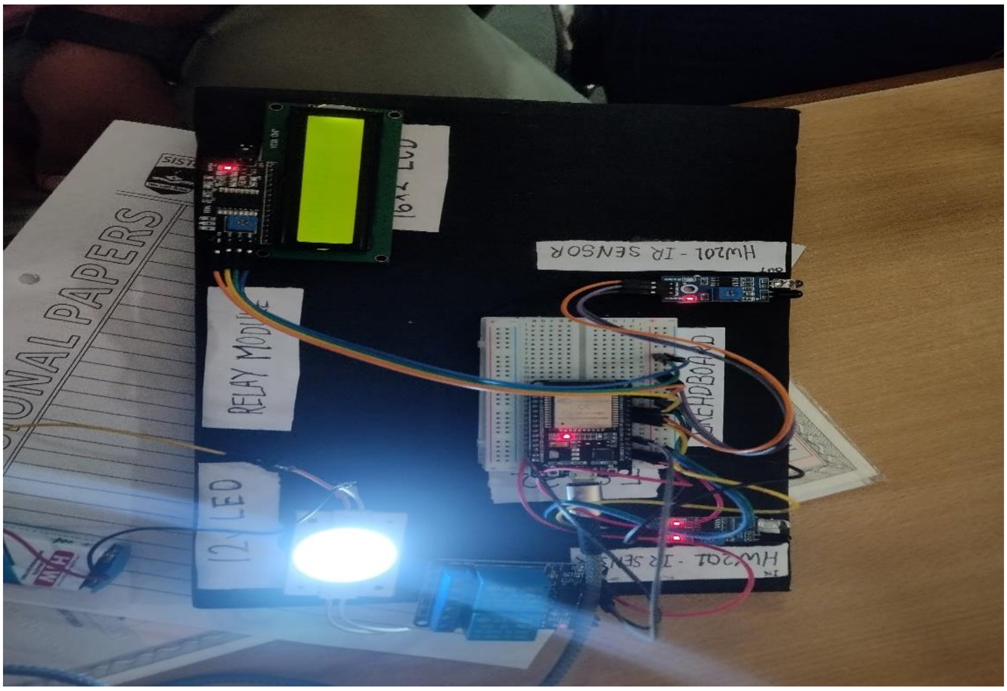

Figure 9.1 Project Output

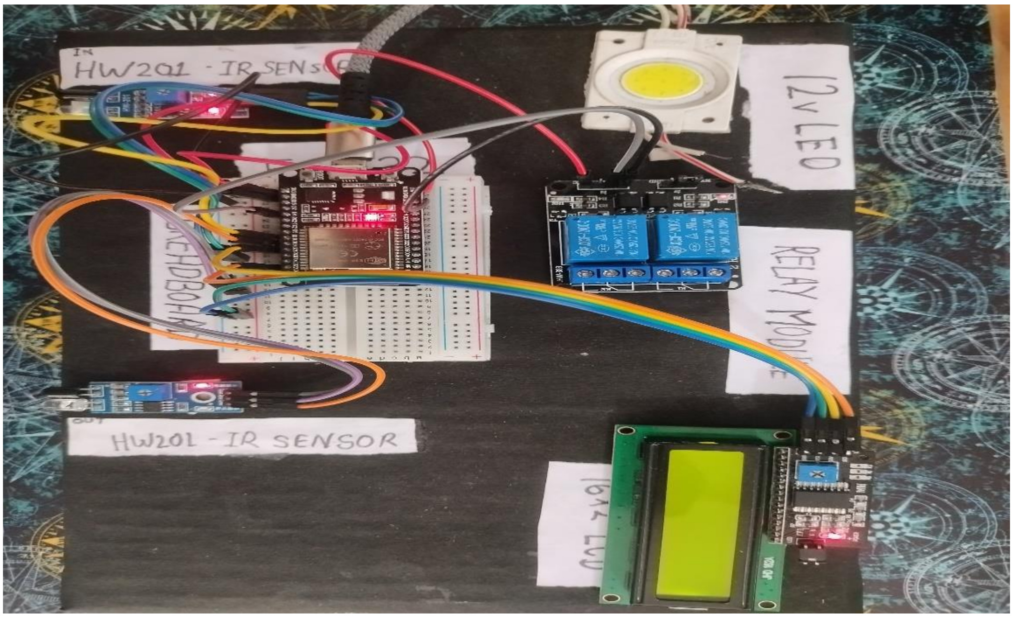

Figure 9.2 Project Output

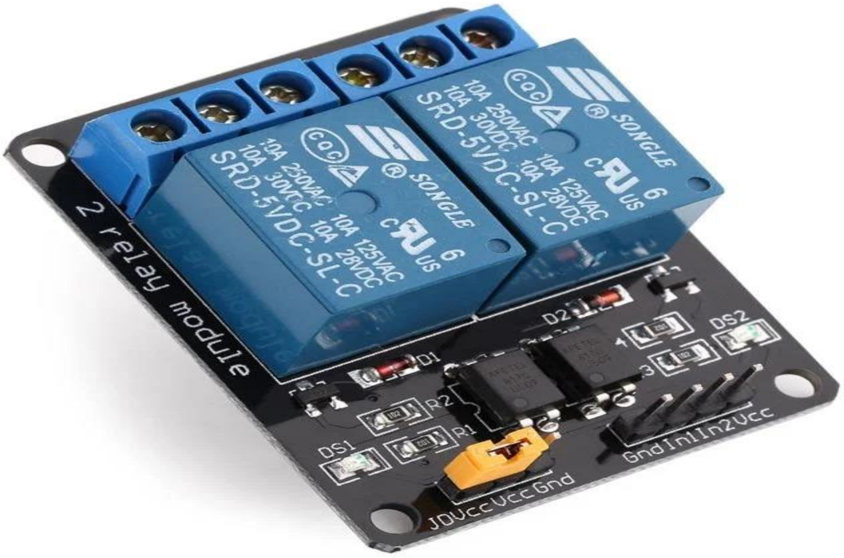

Figure 9.3 Relay Module

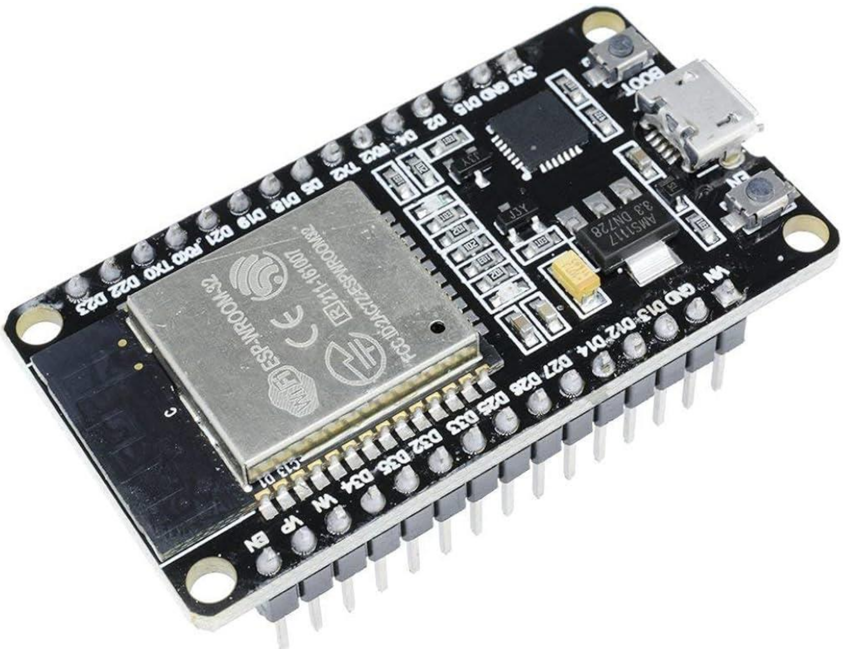

Figure 9.4 ESP 32

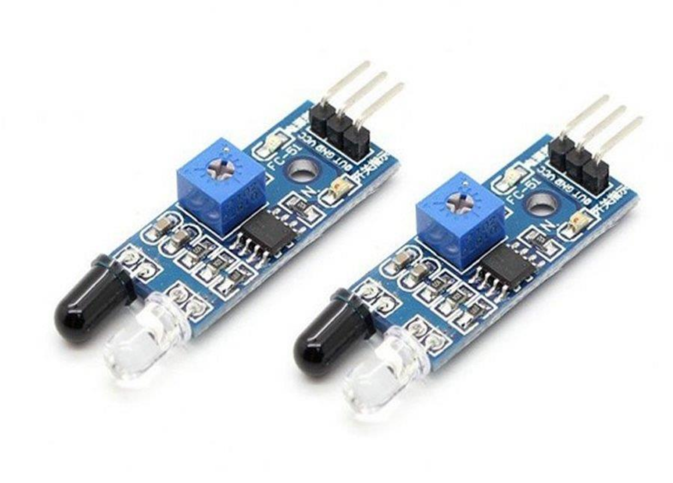

Figure 9.5 IR Sensors

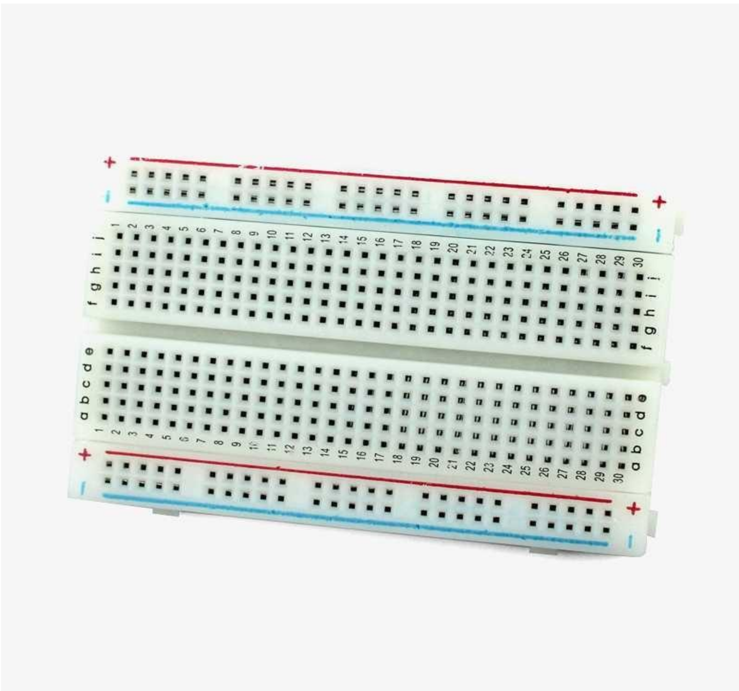

Figure 9.6 Breadboard

Figure 9.7 16x2 LCD

Figure 9.8 12v LED

CHAPTER-10
CONCLUSION AND FUTURE WORK

10.1 CONCLUSION:

The bidirectional visitor counter successfully provides an efficient way to track the movement of people in both directions through a designated entry point. By using sensors (such as infrared, ultrasonic, or laser sensors), the system accurately distinguishes between individuals entering and exiting a given area, offering real-time data on foot traffic. This technology is especially valuable in applications such as retail analytics, event management, public transport stations, and building access control systems. The system can improve resource allocation, optimize space usage, and enhance overall operational efficiency.

The bidirectional nature of the counter ensures that the system provides a more granular and accurate picture of visitor behavior compared to traditional one-way counters. Moreover, integrating with a data processing backend allows for deeper analysis, including peak traffic times, patterns of movement, and other actionable insights.

10.2 FUTURE WORK:

While the current implementation of the bidirectional visitor counter offers significant advantages, there are several potential improvements and areas for future work:

Real-Time Alerts and Automation:

Future systems could be integrated with automated responses, such as notifying security personnel if unusual traffic patterns are detected (e.g., overcrowding or unauthorized access).

Real-time alerts can help prevent accidents or optimize operations in real-time based on crowd flow.

Enhanced Data Privacy:

To ensure privacy compliance, especially with regulations like GDPR, future systems should incorporate privacy-preserving features, such as anonymizing or aggregating data at the source to avoid tracking individuals.

Scalability:

Enhancing the scalability of the system to support large venues, multiple entry/exit points, or entire building infrastructures will increase the applicability of the system for larger organizations, malls, airports, or smart cities.

User Interface Improvements:

The development of more sophisticated user interfaces, such as mobile apps or dashboard tools, could enhance user experience by allowing managers to monitor visitor flow, analyze data, and make decisions on-the-go.

REFERENCES

WEBSITES

[1] https://circuitdigest.com/microcontroller-projects/iot-based-visitor-counter-using-nodemcu-esp8266

[2] https://docs.espressif.com/projects/esp-idf/en/latest/

[3] https://thingspeak.com/docs

[4] https://www.arduino.cc/en/Guide/HomePage

[5] https://hackaday.io/project/19032-cloud-based-visitor-counter

[6] https://www.youtube.com

[7] https://www.trume.in/visitor-management/

[8] https://www.wattsense.com/blog/

[9] https://www.geeksforgeeks.org/pin-diagram-of-8051-microcontroller/

[10] https://www.electroduino.com/bidirectional-visitor-counter-using-arduino/

[11] https://www.slideshare.net/slideshow/automatic-roomlightcontrollerwithbidirectionalvisitorcounter/30538722

PROJECT SUMMARY

About Project

Title of the project	CLOUD CONNECTED VISITOR MANAGEMENT BIDIRECTIONAL COUNTER WITH AUTOMATIC LIGHTING
Semester	8th
Members	3
Team Leader	Syed Ayan Ali
Describe role of every member in the project	Syed Ayan Ali: Work on hardware setup and integration of ESP32, IR sensor, relay as well as all the related documentation. Syed Mustafa Iqbal: Work on system coding and hardware setup Syed Ahmed Ali: Work on system testing, optimization, and performance.
What is the motivation for selecting this project?	The project aims to save energy and improve convenience by automatically controlling lighting based on real-time occupancy.
Project Type (Desktop Application, Web Application, Mobile App, Web)	Iot Based Project

Tools &Technologies

Programming language used	C
IDE used (with version)	Aurduino IDE
Front End Technologies (with version, wherever Applicable)	NA
Back End Technologies (with version, wherever applicable)	NA
Database used (with version)	ThinkSpeak Cloud

Software Design& Coding

Is prototype of the software developed?	Yes
SDLC model followed (Waterfall, Agile, Spiral etc.)	Agile
Why above SDLC model is followed?	Agile is a SDLC model that defines how software development needs to be done. It’s not a single or specific method, and it is the collection of various methodologies and best practices that follow the value statement signed with the customer
Justify that the SDLC model mentioned above is followed in the project.	Since, we didn’t exactly know all the functionalities or the functionalities were frequently changing, we use Agile model, so that we could make desired changes whenever needed.
Software Design approach followed (Functional or Object Oriented)	
Name the diagrams developed (according to the Design approach followed)	Use Case diagram, Architecture
In case Object Oriented approach is followed, which of the OOPS principles are covered in design?	
No. of Tiers (example 3-tier)	3-tier
Total no. of front end pages	
Front end validations applied (Yes / No)	Yes
Session management done (in case of web applications)	
Is application browser compatible (in case of web applications)	
Exception handling done (Yes / No)	No
Commenting done in code (Yes / No)	Yes
Naming convention	Yes

followed (Yes / No)	
What difficulties faced during deployment of project?	
Total no. of Use-case s	
Give titles of Use-cases	Use-case Diagram

Project Requirements

MVC architecture followed (Yes / No)	
If yes, write the name of MVC architecture followed (MVC-1, MVC-2)	
Design Pattern used (Yes / No)	
If yes, write the name of Design Pattern used	
Interface type (CLI / GUI)	
No. of Actors	
Name of Actors	
Total no. of Functional Requirements	
List few important non-Functional Requirements	Correctness, Flexibility, Reliability and Maintainability.

Testing

Which testing is performed? (Manual or Automation)	Manual
Is Beta testing done for this project?	No

Write project narrative covering above mentioned points

This project creates a bidirectional visitor counter that automatically controls room lighting based on occupancy. Using IR sensors and a microcontroller, it counts entries and exits, turning lights on when someone enters and off when the room is empty. This system save energy, enhances convenience, and provides a practical automation solution for various spaces

This project uses IR sensors and a microcontroller to track people entering and exiting a room, automatically controlling the lights. Lights turn on when someone enters and off when the room is empty, promoting energy efficiency. It's a convenient, eco-friendly solution for spaces with variable occupancy.

Syed Ayan Ali	0187CS201169	Guide Signature
Syed Mustafa Iqbal	0187CS211174	(Prof. Ankit Gupta)
Syed Ahmed Ali	0187CS211173	

APPENDIX-1
GLOSSARY OF TERMS

A	
Arduino IDE	An integrated development environment used for programming the ESP32 with custom code for controlling the LED, and sensors.
C	
Cloud Integration	The process of connecting the system to an online platform (e.g., cloud services) for data storage and analysis.
E	
ESP32	The ESP32 is a microcontroller with built-in Wi-Fi and Bluetooth capabilities, used for IoT applications and wireless communication in embedded systems.
I	
IoT (Internet of Thing)	The Internet of Things (IoT) is a network of interconnected devices that collect, share, and act on data through the internet, enabling automation and smart solutions.
L	
LED	An LED (Light Emitting Diode) is an energy-efficient semiconductor light source that emits light when current flows through it.

R	
Relay	A relay is an electrically operated switch that allows a low-power signal to control a higher-power circuit, commonly used for isolating and controlling high-voltage devices like lights, motors, and appliances.
S	
Sensor	Sensors are devices that detect and measure physical properties and convert them into signals that can be read and processed by electronic systems.

W	
Wi-Fi	Wi-Fi is used to enable remote monitoring and control of the visitor counter and lighting system via a mobile app or web interface. The ESP32 microcontroller connects to the internet, allowing real-time data exchange and automation from anywhere.