Project Design
On
IoT-Based Smart City Heatwave & Public Safety Monitoring System

Course Code: CSE - 460
Course Title: IoT Laboratory

Submitted By

Sovon Mallick(380)

Md. Shakhawat Hosen(376)

Sudipta Singha(408)

Submitted To

Dr. Md. Ezharul Islamr

Professor

Samsun Nahar Khandakar

Lecturer

Department of Computer Science and Engineering

Jahangirnagar University

Dhaka, Bangladesh

March 22, 2025

Contents

1. IoT-Based Smart City Heatwave & Public Safety Monitoring System

1.1 Introduction

1.1.1 Objective

The primary objective of this project is to design and implement an IoT-based system that monitors environmental conditions—specifically heatwave indicators (temperature and humidity), air quality (gas levels), and worker safety—in urban environments. The system aims to provide real-time alerts, data visualization, and actionable insights to mitigate health risks for workers and the public in smart cities. By integrating multiple sensors and cloud-based platforms, it ensures timely notifications and accessible data for stakeholders.

1.1.2 Problem Statement

Urban areas increasingly face challenges from rising temperatures and deteriorating air quality due to climate change and industrialization. Heatwaves, combined with high pollution levels, pose severe health risks such as heatstroke, respiratory issues, and fatigue, particularly for outdoor workers. Additionally, overcrowded urban zones amplify these risks, yet current systems often lack real-time monitoring and localized alerts. This project addresses these gaps by delivering an affordable, scalable IoT solution for environmental and safety monitoring.

1.1.3 Scope

The project focuses on the following:

Monitoring temperature, humidity, gas levels, and crowd density in real-time. Tracking worker presence in high-risk zones using RFID technology. Providing immediate alerts through local displays, buzzers, and cloud notifications. Logging data locally (SD card) and remotely (cloud platforms like ThingSpeak or Blynk). Limitations include:

No advanced AI-based predictive analytics in the initial design. Focus on a single deployment site rather than a city-wide network. Dependency on stable Wi-Fi connectivity for cloud integration.

1.2 System Overview

1.2.1 Purpose

The system is designed to continuously monitor environmental and safety parameters in urban settings, detect hazardous conditions (e.g., extreme heat, poor air quality, or overcrowding), and alert users in real-time to prevent health incidents.

1.2.2 Key Features

- **Real-time Temperature and Humidity Monitoring**: Tracks heatwave conditions using precise sensors.
- **Gas Level Detection**: Monitors air pollutants (e.g., CO, smoke) to assess air quality.

- **Worker Tracking**: Uses RFID to identify and log worker presence in hazardous zones.
- **Crowd Detection**: Employs ultrasonic sensors to measure crowd density and prevent overcrowding risks.
- **Multi-channel Alerts**: Triggers local buzzers, displays warnings on an OLED screen, and sends cloud-based notifications.
- **Data Logging**: Stores data on an SD card for offline analysis and uploads it to the cloud for remote access.

1.2.3 Use Case

Consider a construction site in a smart city during summer. The system detects a temperature of 38°C, humidity of 70%, and elevated CO levels from nearby traffic. Simultaneously, it identifies a high crowd density and confirms the presence of five workers via RFID. It activates a buzzer, displays "Heatwave Warning: Poor Air Quality" on the OLED, and sends a notification to supervisors via Blynk, prompting immediate action (e.g., worker breaks, ventilation adjustments).

1.3 Architecture Design

1.3.1 Circuit Design

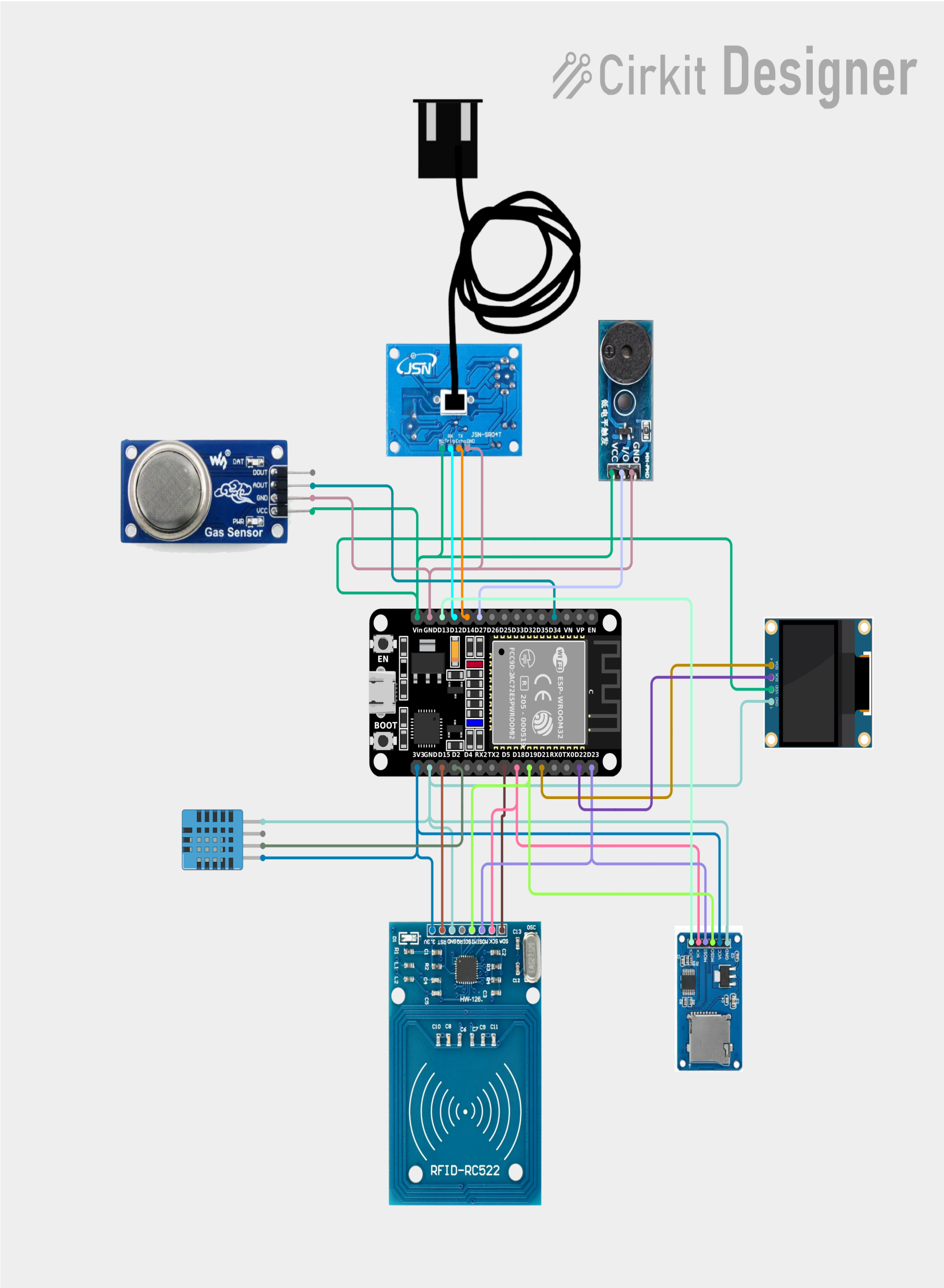

Figure 1.1: Circuit Design

Circuit Design Link: https://app.cirkitdesigner.com/project/9c0c040a-aed3-4cb9-8d3c-52e612976cfb

1.3.2 Data Flow

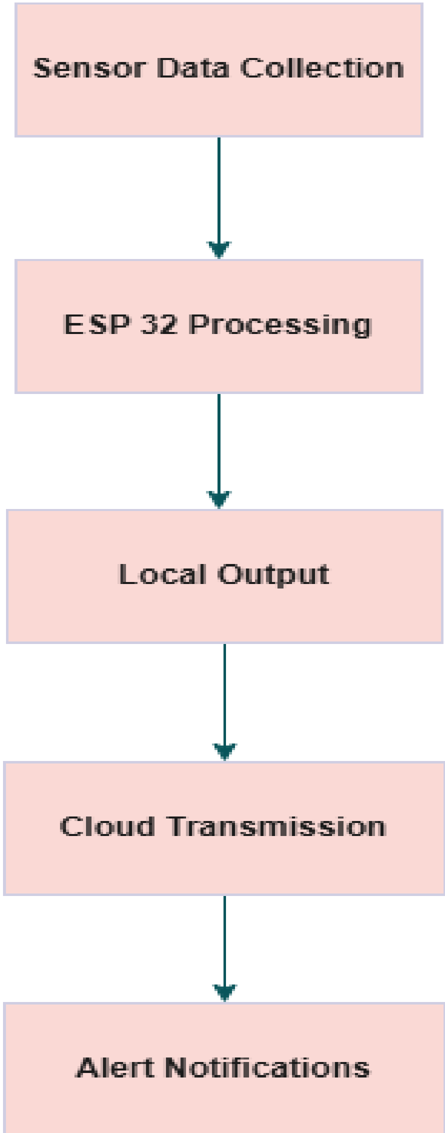

Figure 1.2: Data Flow Process

Sensor Data Collection: DHT11, MQ gas sensor, ultrasonic sensor, and RFID module gather environmental and safety data.
ESP32 Processing: The microcontroller processes raw data (e.g., averages temperature readings, flags thresholds).

Local Output: Processed data is displayed on the OLED and logged to the SD card; the buzzer activates if thresholds are exceeded.
Cloud Transmission: Data is sent to ThingSpeak/Blynk via Wi-Fi for remote monitoring.
Alert Notifications: Alerts are pushed to users' mobile devices or dashboards.

1.3.3 Network Topology

The system adopts a **star topology**, with the ESP32 acting as the central node. Peripheral devices (sensors, OLED, SD card) connect directly to the ESP32 via wired interfaces (I2C, SPI), while the ESP32 communicates with the cloud over Wi-Fi using MQTT or HTTP protocols. This topology ensures simplicity and scalability for a single-site deployment.

1.4 Hardware Components

- **ESP32 Microcontroller**: A dual-core processor with built-in Wi-Fi and Bluetooth, chosen for its processing power, connectivity, and GPIO versatility.
- **DHT11 Sensor**: Measures temperature (0–50°C) and humidity (20–80%) with ±2°C and ±5% accuracy, respectively, suitable for heatwave detection.
- **MQ Gas Sensor (e.g., MQ-135)**: Detects air pollutants like CO, NH3, and smoke, providing analog output for air quality assessment.
- **RFID Module (MFRC522)**: Uses 13.56 MHz RFID tags to

track worker presence, offering a low-cost, reliable identification method.

- **Ultrasonic Sensor (HC-SR04)**: Measures distances (2–400 cm) to estimate crowd density, with ±3 mm accuracy.
- **Buzzer**: A 5V active buzzer for audible alerts, triggered by threshold breaches.
- **OLED Display (SSD1306, 0.96")**: A 128x64 pixel I2C display for real-time data visualization.
- **SD Card Module**: Enables local data logging with SPI interface, using a microSD card (up to 32 GB).
- **Power Supply**: 5V USB or 3.7V Li-ion battery with a regulator, ensuring portability and reliability.

1.5 Software Components

- **Programming Language**: C++ via Arduino IDE, chosen for its extensive library support and ESP32 compatibility.
- **Platforms**:

  ThingSpeak: For cloud storage, visualization, and data analytics.
  Blynk: For mobile app integration and real-time notifications.
- **Data Processing**: Local: ESP32 performs basic filtering (e.g., outlier removal) and threshold checks.
  Cloud: ThingSpeak aggregates data for trends; Blynk pushes alerts.

- **Libraries**: DHT.h, SPI.h, Wire.h, Adafruit_SSD1306.h, MFRC522.h, etc., for sensor and module interfacing.

1.6 Communication Protocols

- **Wi-Fi**: ESP32 connects to the cloud using MQTT (lightweight, real-time) or HTTP (simpler setup).
- **SPI**: High-speed communication for SD card and RFID module, ensuring reliable data transfer.
- **I2C**: Low-bandwidth protocol for OLED display, minimizing pin usage.
- **Digital/Analog Pins**: Used for DHT11, MQ sensor, ultrasonic sensor, and buzzer interfacing.

1.7 Implementation Plan

- **Week 1: Hardware Assembly**

  Solder and connect sensors, OLED, buzzer, and SD card to ESP32. Test individual component functionality.
- **Week 2: Software Development and Sensor Integration** Write C++ code for sensor data collection and local output. Integrate RFID and ultrasonic sensor logic.
- **Week 3: Cloud and Alert System Integration** Configure ThingSpeak and Blynk accounts. Implement Wi-Fi connectivity and alert triggers.
- **Week 4: Testing and Final Evaluation** Conduct system-wide tests in simulated conditions. Refine code and hardware based on results.

Milestones: Functional prototype by Week 2, cloud-integrated system by Week 3, fully tested system by Week 4.

1.8 Budget Estimation

Component	**Quantity**	**Cost (Approx., TK)**
ESP32	1	500
DHT11 Sensor	1	150
MQ Gas Sensor (MQ-135)	1	200
RFID Module (MFRC522)	1	400
Ultrasonic Sensor (Waterproof)	1	590
OLED Display (0.96”)	1	290
Digital Buzzer	1	150
SD Card Module	1	100
MicroSD Card (8 GB)	1	300
Power Connection(Battery)	-	500
Miscellaneous (wires, breadboard)	-	200
Total	-	**3380 TK**

1.9 Testing and Evaluation

– **Sensor Accuracy**: Test DHT11 and MQ sensors against reference devices in controlled heat and pollution conditions.

– **Alert Timeliness**: Verify buzzer activation and Blynk notifications occur within 5 seconds of threshold breaches.

– **RFID Response**: Ensure worker detection occurs within 1 second of tag proximity.

– **Crowd Detection**: Validate ultrasonic sensor accuracy at varying distances (1–10 meters).

- **Data Logging**: Confirm SD card and cloud data match real-time readings.

Success Criteria: 95% sensor accuracy, 100% alert reliability, and ¡10-second system response time.

1.10 Future Enhancements

- **AI Integration**: Integration of Machine learning to predict heatwave patterns based on historical data.
- **Wireless Nodes**: Deployment of battery-powered sensor nodes with LoRa for wider coverage.
- **Government Integration**: Link alerts to municipal systems for city-wide warnings.
- **Mobile App**: Developing a custom app for detailed analytics and user controls.

1.11 Conclusion

This IoT project integrates affordable hardware and robust software to address heatwave and pollution challenges in smart cities. By monitoring environmental conditions, tracking workers, and issuing real-time alerts, it enhances public safety and worker well-being. The scalable design and cloud connectivity lay a foundation for future expansions, making it a practical solution for urban resilience.