DECLARATION

We hereby declare that this lab project has been done by us under the supervision of Indrojit Sarkar, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere as lab projects.

Submitted To:
Indrojit Sarkar
Lecturer
Department of Computer Science and Engineering
Daffodil International University

Submitted by:

Ashir Ahbab Sium Student ID: 221-15-6028 Dept. of CSE, DIU	
Tanber Ahmmed Student ID: 221-15-5838 Dept. of CSE, DIU	Istiak Jaman Anan Student ID: 221-15-5840 Dept. of CSE, DIU
Maskaud Imran Plaban Student ID: 221-15-5845 Dept. of CSE, DIU	Utsab Das Student ID: 221-15-5207 Dept. of CSE, DIU

COURSE & PROGRAM OUTCOME

The following course have course outcomes as following:

1. Course Outcome Statements

CO's	Statements
CO1	**Demonstrate** practical use of basic electronic components and microcontrollers in embedded lab experiments.
CO2	**Develop** and debug C++ based arduino codes with circuit diagrams for embedded and IoT systems.
CO3	**Design** Arduino-based projects using sensors and actuators for real-world embedded system applications.
CO4	**Design** and Implement projects using ESP32 and STM32 microcontrollers for advanced embedded and IoT applications.

2. Mapping Course Outcome (COs) with the Teaching-Learning

COs	Teaching Learning Activities	Assessment Strategy	Domain Level/Learning Taxonomy	Level of Knowledge Profile	Complex Engineering Problems	Complex Engineering Activities
CO1	TLA1	Lab Experiment	C2	K1-K4	EP1	EA1
CO2	TLA2	Report/Project	C3, A1	K1-K4	EP2	EA2
CO3	TLA3	Project	C4, P1	K5	EP3	EA3

8. **Program Outcome (PO) Attainment**

COs	CO Result	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1													
CO2													
CO3													
CO4													
CO5													
PO Attainment (%)													
PO Attained (Yes/No)													

Table of Contents

Introduction

1.1 Introduction

The Smart Environment Detection System is designed to monitor indoor air quality and environmental conditions in real time. It uses an **MQ2 sensor** to detect harmful gases and smoke, along with a **DHT11 sensor** to measure temperature and humidity levels. By integrating with **Blynk IoT**, the system provides users with live updates through mobile devices, enabling timely alerts and preventive actions. This project contributes to creating safer and healthier living or working environments by combining IoT technology with affordable sensors.

1.2 Motivation

In today's world, environmental quality plays a critical role in human health and comfort. Exposure to harmful gases such as carbon monoxide, methane, and smoke can pose severe risks, while high temperature and humidity levels create discomfort and may even affect productivity. Unfortunately, people often remain unaware of the quality of air around them until it becomes a serious problem.

This project is motivated by the need to build an affordable and accessible **IoT-based monitoring solution** that can continuously track air quality and environmental conditions. By providing **real-time data and instant alerts**, individuals and families can take proactive measures to maintain a safer and healthier environment.

1.3 Objectives

The main objectives of this project are:

1. **Environmental Monitoring** – To continuously measure gas levels, temperature, and humidity using MQ2 and DHT11 sensors.

2. **Real-Time Data Access** – To transmit sensor data to the **Blynk IoT platform**, enabling users to view conditions remotely via smartphone.

3. **Alert System** – To generate instant notifications when gas concentration or temperature/humidity exceeds safe thresholds.

4. **User Awareness** – To raise awareness about indoor environmental conditions and empower users to take corrective actions.

5. **Low-Cost IoT Solution** – To design an affordable, scalable, and energy-efficient system that can be deployed in homes, offices, and small industries.

1.4 Feasibility Study

The Smart Environment Detection System using MQ2 and DHT11 sensors is designed to monitor gas levels, temperature, and humidity in real time.

- **Technical Feasibility**
  The project is technically feasible since the required hardware (ESP32, MQ2, DHT11, buzzer, LEDs) is low-cost, widely available, and easy to integrate. The system uses well-documented sensors and libraries, making development straightforward.

- **Operational Feasibility**
  The system is simple to operate, as it provides real-time monitoring and alerts. Users can easily understand the visual indicators (LEDs/buzzer) or digital notifications, making it practical for home, office, or industrial use.

- **Economic Feasibility**
  The cost of implementation is low since MQ2 and DHT11 sensors are inexpensive, and the ESP32 provides both processing power and wireless connectivity at an affordable price. The long-term benefit of improving air quality awareness and environmental safety outweighs the initial cost.

- **Social Feasibility**
  With growing concerns about air pollution and environmental health, the project has significant social relevance. It can be used to raise awareness about indoor air quality, prevent health hazards from harmful gases, and improve living standards in both urban and rural communities.

1.5 Project Outcome

The project successfully develops a Smart Environment Detection System that monitors air quality, temperature, and humidity in real-time. By integrating the MQ2 gas sensor and the DHT11 sensor with an ESP32 microcontroller, the system provides accurate environmental readings and notifies users through visual and digital alerts. This outcome enhances awareness of indoor and outdoor environmental conditions, helping to ensure health, safety, and comfort. Its low cost and scalability make it suitable for homes, workplaces, and public spaces. Overall, the project demonstrates the effectiveness of IoT-based solutions in promoting a safer and healthier living environment.

Requirement Analysis

2.1 Overview

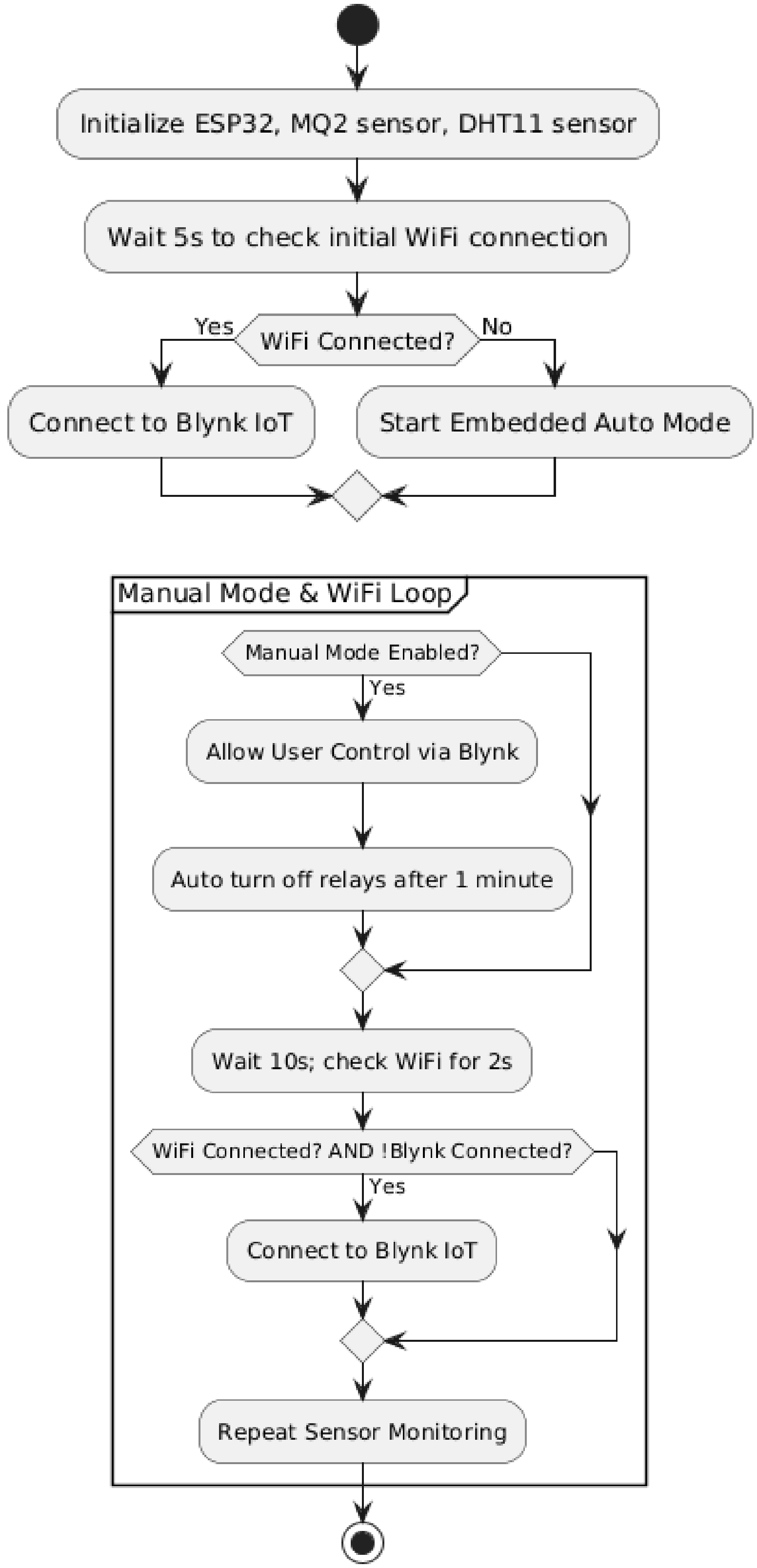
The Smart Environment Detection System is an IoT-based solution designed to monitor air quality, temperature, and humidity in real time. It uses an **MQ2 sensor** to detect harmful gases like smoke, methane, and carbon monoxide, while a **DHT11 sensor** measures temperature and humidity. An **ESP32 microcontroller** collects sensor data and controls relays, LEDs, and a buzzer for alerts. The system can operate both **offline** using embedded logic and **online** via the **Blynk IoT platform**, providing real-time monitoring, notifications, and remote access. It offers a cost-effective, scalable, and user-friendly solution for maintaining a safer environment.

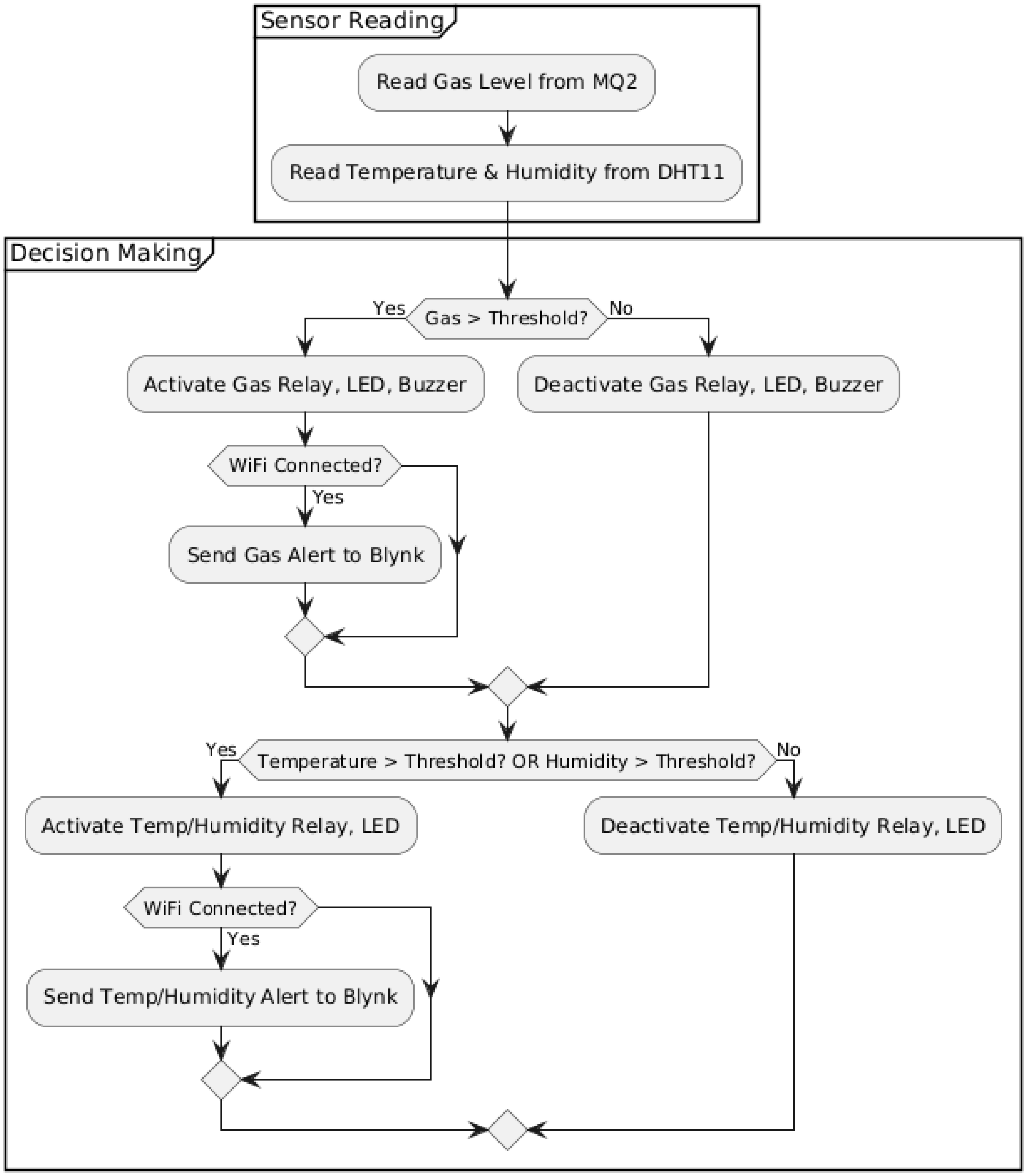
2.2 Methodology

The Smart Environment Detection System is developed using an **ESP32 microcontroller** interfaced with an **MQ2 gas sensor** and a **DHT11 temperature and humidity sensor**. The methodology involves continuous data acquisition from these sensors at regular intervals. Sensor readings are processed by the ESP32 to determine if environmental parameters exceed predefined thresholds. In **offline mode**, the system activates relays, LEDs, and a buzzer to alert users locally. In **online mode**, data is transmitted to the **Blynk IoT platform**, enabling remote monitoring and notifications on smartphones. The system also supports a **manual mode**, allowing users to control relays directly through the Blynk app, with safety mechanisms to automatically turn them off after one minute. This hybrid methodology ensures reliable operation in both connected and disconnected environments.

2.3 Work Flow of the System

The Smart Environment Detection System follows a structured workflow to monitor and respond to environmental conditions. First, the **ESP32 microcontroller** continuously reads data from the **MQ2 gas sensor** and the **DHT11 temperature and humidity sensor**. The readings are compared against predefined safety thresholds. If any parameter exceeds its limit, the system triggers local alerts using **LEDs, relays, and a buzzer**. Simultaneously, if connected to the internet, the data is sent to the **Blynk IoT platform**, where users receive real-time notifications on their smartphones. The system also supports a **manual control mode** through Blynk, which automatically turns off relays after one minute to prevent misuse. This workflow ensures both automated safety responses and remote monitoring capabilities.

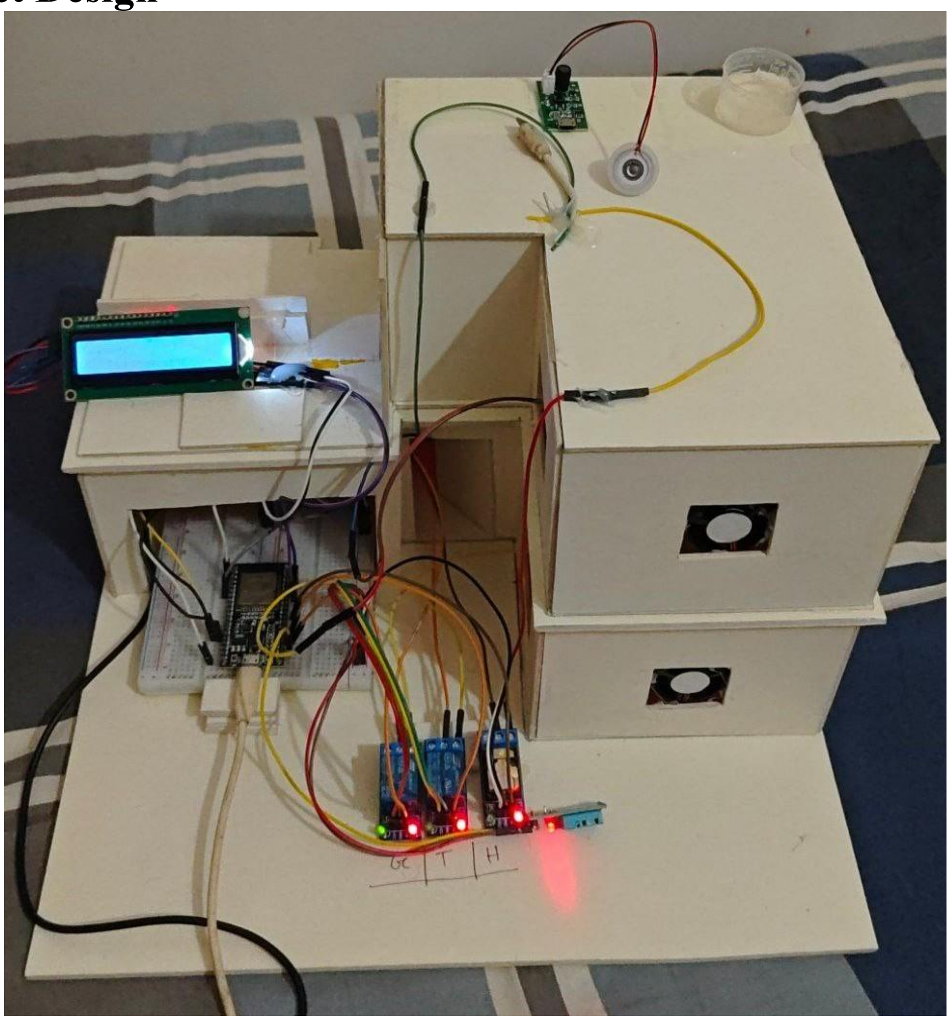

Initialize ESP32, MQ2 sensor, DHT11 sensor
Wait 5s to check initial WiFi connection
WiFi Connected?
Yes
No
Connect to Blynk IoT
Start Embedded Auto Mode
Manual Mode & WiFi Loop
Manual Mode Enabled?
Yes
Allow User Control via Blynk
Auto turn off relays after 1 minute
Wait 10s; check WiFi for 2s
WiFi Connected? AND !Blynk Connected?
Yes
Connect to Blynk IoT
Repeat Sensor Monitoring

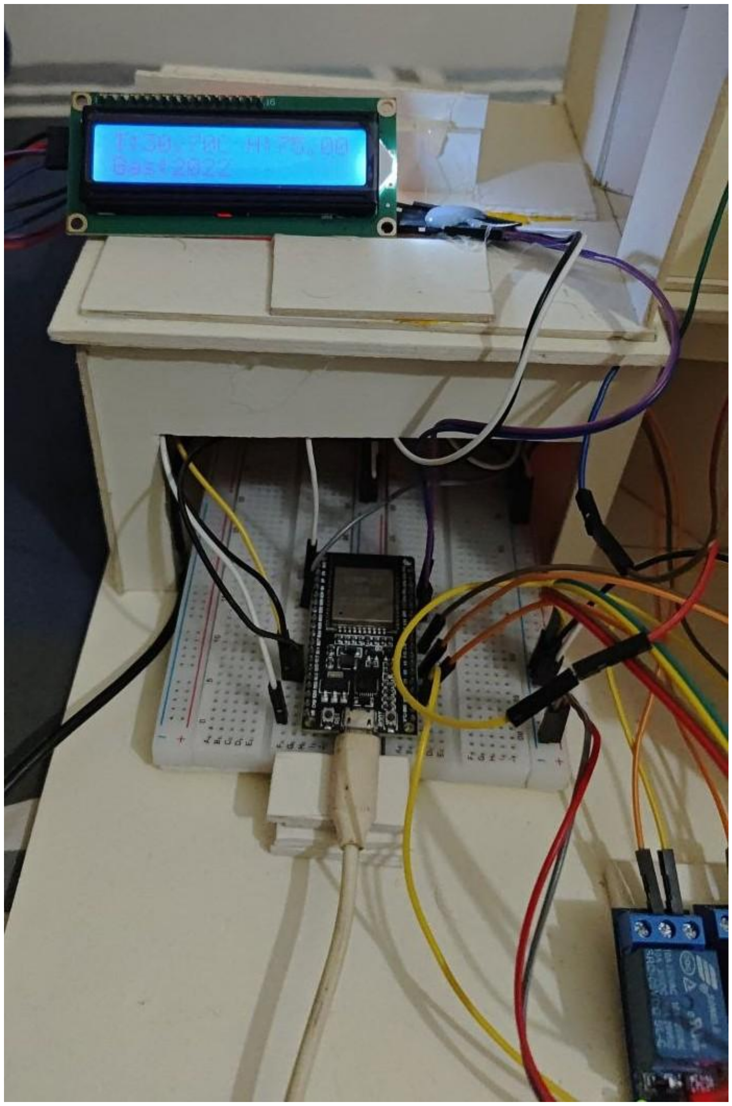

Sensor Reading
Read Gas Level from MQ2
Read Temperature & Humidity from DHT11
Decision Making
Gas > Threshold?
Yes
No
Activate Gas Relay, LED, Buzzer
Deactivate Gas Relay, LED, Buzzer
WiFi Connected?
Yes
Send Gas Alert to Blynk
Temperature > Threshold? OR Humidity > Threshold?
Yes
No
Activate Temp/Humidity Relay, LED
Deactivate Temp/Humidity Relay, LED
WiFi Connected?
Yes
Send Temp/Humidity Alert to Blynk

2.4 Overall Project Plan

The project plan is structured in four main phases: **Planning, Hardware Setup, Software Development, and Testing & Deployment**. In the planning phase, sensor selection (MQ2 for gas, DHT11 for temperature and humidity) and microcontroller choice (ESP32) were finalized. Hardware setup involved connecting sensors, relays, buzzer, and LCD to the ESP32. Software development included programming sensor reading, relay control, manual and auto modes, LCD display, and Blynk integration. Testing focused on verifying sensor accuracy, relay operation, and Blynk notifications. Finally, deployment ensures continuous monitoring with automatic switching between embedded and Blynk modes based on internet connectivity, providing a reliable smart environment detection system.

2.5 Project Design





2.6 Project Costs and Budget Overview

The total cost of the smart environment detection system was carefully planned to balance affordability and functionality. Key components included the ESP32 microcontroller, MQ2 gas sensor, DHT11 temperature-humidity sensor, relay modules, LCD display, buzzer, and necessary connecting wires and power supply. Additional costs were incurred for replacements due to damaged relay modules and calibration adjustments for the LCD. Overall, the budget emphasized using widely available and cost-effective components while maintaining system reliability, safety, and expandability for future enhancements.

Components	**Costs**
MQ-2	250
DHT11	250
LCD	350
ESP32	450
Fan	450
Relay	500
Connecting Wire	250
Breadboard	240
Led	50
Buzzer	20
Humidifier	150
USB Cable	300
Home Structure	450
Travelling Costs	200
Other	650
Total	**= 5560 Taka**

Implementation and Results

3.1 Implementation

The implementation of the smart environment detection system involved integrating the ESP32 microcontroller with MQ2 and DHT11 sensors, relays, a buzzer, and a 16×2 LCD display. The system was programmed to operate in two modes: **Auto Mode** (embedded logic) and **Blynk Mode** (remote control via IoT). Sensor readings are displayed on the LCD in real-time, while the relays and buzzer respond to threshold exceedances for gas, temperature, and humidity. The system automatically sends notifications through Blynk when values exceed safe limits. During testing, the device successfully switched between Auto Mode and Blynk Mode based on internet connectivity. Manual mode was verified, with automatic turn-off after one minute. Results demonstrate that the system reliably monitors environmental parameters, provides timely alerts, and maintains safety measures even in the absence of an internet connection, fulfilling the project objectives effectively.

3.2 Performance Analysis

The smart environment detection system was evaluated based on **sensor accuracy, response time, reliability, and alert effectiveness**. The MQ2 sensor accurately detected gas concentration above the threshold, while the DHT11 sensor reliably measured temperature and humidity with minimal deviation. Relays and buzzer responded promptly to threshold breaches, ensuring immediate safety actions. The system maintained stable performance in both **Auto Mode** and **Blynk Mode**, with smooth switching based on internet connectivity. Manual mode functioned correctly, with automatic turn-off after one minute to prevent prolonged activation. Blynk notifications were sent without delay when WiFi was available. Overall, the system demonstrated **high reliability, real-time monitoring, and effective alerting**, making it suitable for smart home or laboratory environments. Any minor delays in sensor readings were negligible and did not affect the system's operational efficiency.

3.3 Results and Discussion

The implementation of the smart environment detection system yielded reliable and accurate monitoring of gas concentration, temperature, and humidity. Sensor readings were consistently displayed on the LCD, and relays and buzzer activated appropriately when thresholds were exceeded. In **Auto Mode**, the system maintained safety without internet, while in **Blynk Mode**, it allowed remote monitoring and control. Manual mode operated effectively, with automatic turn-off preventing prolonged activation. Notifications through Blynk were successfully sent for gas and temperature alerts. The system demonstrated **seamless switching between modes** based on WiFi availability, ensuring continuous operation. Performance analysis indicated minimal latency in sensor detection and relay response. Overall, the results confirm that the project meets its objectives, providing a **robust, automated, and IoT-enabled environment monitoring solution** suitable for homes, labs, or industrial applications, combining safety, convenience, and real-time data accessibility.

Impacts & Problems

4.1 Impact on Life

The smart environment detection system positively impacts daily life by providing **real-time monitoring of air quality, temperature, and humidity**, ensuring safer living and working conditions. Early detection of harmful gas concentrations through the MQ2 sensor helps prevent accidents, while temperature and humidity monitoring improves comfort and reduces health risks. Integration with Blynk allows remote control and notifications, giving users peace of mind even when away from home. Automatic switching between embedded and IoT modes ensures continuous operation, regardless of internet availability. By combining safety, convenience, and real-time alerts, the system promotes a **healthier, more secure, and connected environment**, demonstrating the practical benefits of IoT-enabled home and laboratory automation. It empowers individuals to make informed decisions about their surroundings and enhances overall quality of life.

4.2 Impact on Environment

The smart environment detection system enhances environmental safety by continuously monitoring gas, temperature, and humidity levels. Early detection of harmful gases prevents potential hazards, while automated control of relays reduces unnecessary energy consumption. Remote alerts via Blynk enable timely actions like ventilation, minimizing pollution and resource waste. By promoting efficient energy use and raising awareness of air quality, the system encourages **eco-friendly practices** and supports a safer, healthier, and more sustainable environment in homes, labs, and workplaces.

4.3 Sustainability Plan

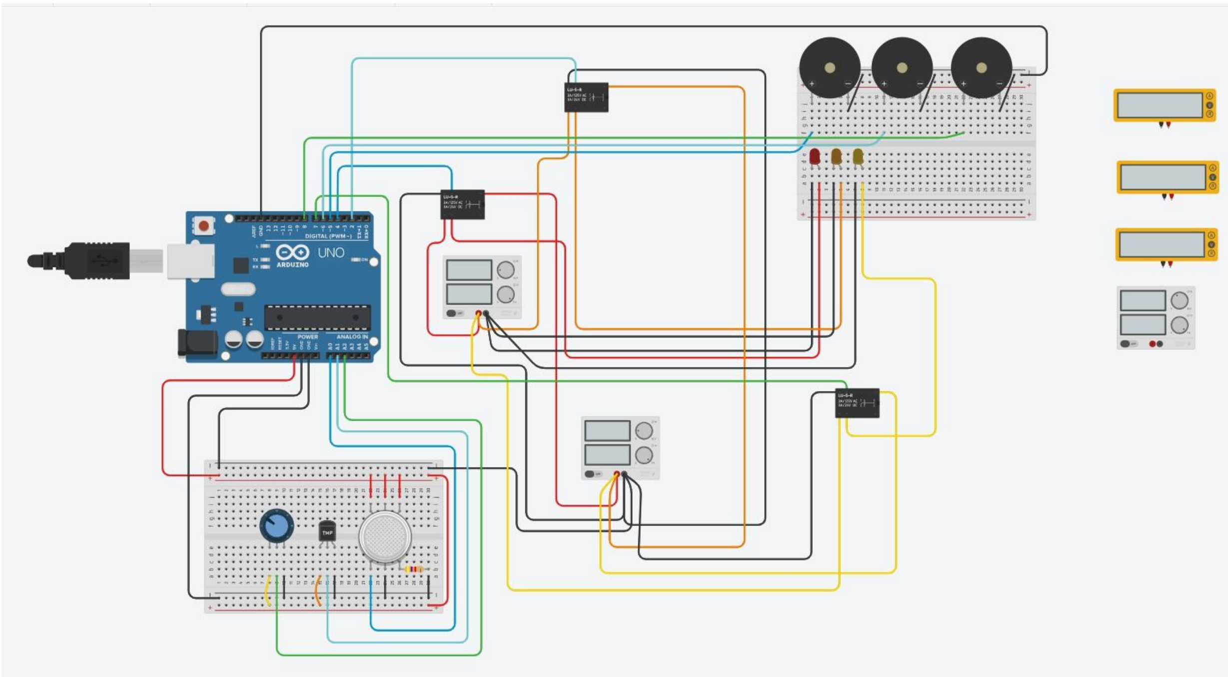
The sustainability plan for the smart environment detection system emphasizes **long-term reliability, energy efficiency, and minimal environmental impact**. The ESP32 microcontroller and sensors are chosen for low power consumption, and the relays operate only when necessary, ensuring efficient energy use. Automatic mode reduces human intervention, while remote monitoring via Blynk allows timely adjustments, extending the lifespan of hardware components. The modular design facilitates easy maintenance, replacement, or upgrades of individual parts without discarding the entire system. Regular software updates improve performance and adaptability to changing environmental conditions. By combining **smart automation, energy efficiency, and maintainable design**, the system ensures a sustainable approach to continuous environmental monitoring, promoting safety, resource conservation, and long-term usability in homes, laboratories, and workplaces.

4.4 Project Management and Teamwork

The development of the smart environment detection system was carried out through a **well-coordinated and structured team effort**. Ashir Ahbab Sium and Tanber Ahmmed were responsible for the initial design and planning phase. They created the system architecture, workflow diagrams, and functional requirements, ensuring that all team members had a clear understanding of the project objectives and scope. Following this, Ashir Ahbab Sium and Istiak Jaman Anan handled the software development. They coded the ESP32 microcontroller, integrated the MQ2 and DHT11 sensors, managed relay and buzzer control, and implemented LCD display functionality. They also integrated Blynk IoT connectivity to enable real-time monitoring, notifications, and remote control features. Meanwhile, Maskaud Imran Plaban and Utsab Das focused on the **hardware assembly, building plan, wiring, and installation** of all sensors, relays, and output devices. Throughout the project, the team maintained **regular meetings, task tracking, and problem-solving sessions**, which helped in timely troubleshooting and ensured smooth coordination between hardware and software components. Responsibilities were divided according to expertise, which improved efficiency and allowed members to focus on their strengths. This collaborative approach ensured the system was completed with **high reliability, full functionality, and efficiency**, highlighting strong teamwork, communication, and project management skills.

4.5 Initial Planning

The initial planning phase focused on defining the **scope, objectives, and system architecture** of the smart environment detection system. Key components like ESP32, MQ2, DHT11, relays, buzzer, and LCD were identified. Functional requirements, including auto/manual modes, threshold alerts, and Blynk IoT integration, were outlined. Workflow diagrams and task allocation were prepared based on team expertise. A timeline with milestones was set for prototyping, integration, and testing. This planning ensured **organized development, minimized errors, and aligned the team**.



4.6 Problem Statements

During the development of the smart environment detection system, several challenges were encountered:

1. **Relay Issues:** During the initial Tinkercad design, current started flowing back to the Arduino, causing relay malfunctions and requiring adjustments to the circuit.

2. **ESP32 Board Manager Problems:** Installing the ESP32 board manager failed multiple times (over 7 attempts), taking 3 days to successfully configure the board for development.

3. **LCD Calibration Problems:** The initial LCD library caused display issues, requiring manual calibration and adjustments to ensure proper output.

4. **Damaged Relay Modules:** Two days before submission, all relay modules stopped functioning. Attempts to replace them were complicated as the suppliers sent another set of faulty relays. This caused significant delays, as the team had to identify reliable suppliers, purchase new modules, and rewire the circuit. Additional testing was needed to ensure the new relays worked correctly with the ESP32 board without causing backflow current or damage to other components.

5. **WiFi and Blynk IoT Connectivity:** The project initially could not run without WiFi. Ensuring stable internet connection and Blynk integration took 4 days of debugging and

code adjustments.

6. **Manual/Auto Mode Logic:** On the Blynk app, manual mode did not automatically revert to auto mode if left on. This required implementing a timeout feature to prevent system lock in manual mode.

7. **Integration Challenges:** Combining sensor readings (MQ2 and DHT11), relay control, LCD display, buzzer alerts, and IoT notifications required careful synchronization to ensure seamless functionality.

These challenges highlight the **complexity of integrating hardware and software**, IoT connectivity, and user interface features. Overcoming these issues involved debugging, manual calibration, code optimization, and hardware replacement, ultimately improving the robustness and reliability of the smart environment detection system.

Conclusion

5.1 Summary

The smart environment detection system was successfully designed and implemented using ESP32, MQ2, and DHT11 sensors, along with relays, a buzzer, and an LCD display. The system can operate in both auto and manual modes, controlling relays based on sensor readings while sending alerts via Blynk IoT. Key challenges, including hardware malfunctions, WiFi connectivity issues, and software integration, were addressed through careful troubleshooting, code optimization, and component replacement. The final system reliably monitors gas levels, temperature, and humidity, providing real-time notifications and improving safety and awareness in the environment. This project demonstrates the effective combination of IoT technology, embedded systems, and user-friendly interfaces to create a functional and adaptable environmental monitoring solution.

5.2 Future Works

In the future, the smart environment detection system can be enhanced in several ways. Integration with additional sensors, such as CO2, smoke, or particulate matter detectors, can provide more comprehensive environmental monitoring. The system could also incorporate cloud-based data logging and analytics for long-term trend analysis and predictive alerts. Mobile app features can be expanded to allow remote control of relays, historical data visualization, and multi-user access. Additionally, implementing energy-efficient designs, solar power integration, and machine learning algorithms for automatic threshold adjustments could further improve the system's sustainability, accuracy, and adaptability to different environments.

5.3 Performance and Scalability

The smart environment detection system demonstrated robust and reliable performance, effectively monitoring gas levels, temperature, and humidity in both auto and manual modes. The MQ2 and DHT11 sensors provided accurate and consistent readings, while the relay modules responded promptly to control signals. The LCD display delivered clear real-time information, and Blynk IoT integration allowed instant remote notifications and monitoring. Scalability of the system is practical, as additional sensors, relays, or actuators can be easily incorporated through modular code updates. Multiple ESP32 nodes can also be networked to a single Blynk account or cloud server, enabling deployment across larger environments such as industrial facilities or smart buildings. Furthermore, by optimizing WiFi connectivity, power management, and sensor calibration, the system maintains high performance and reliability even as complexity increases. This adaptability ensures the system can evolve with future environmental monitoring needs while maintaining operational efficiency.

5.4 Hardware and Technical Constraints

The smart environment detection system faced several hardware and technical constraints during development. The ESP32 board, while versatile, required stable WiFi connectivity for full Blynk IoT functionality, limiting offline performance. Relay modules were prone to damage under high current or faulty connections, requiring careful handling and replacement. The MQ2 gas sensor and DHT11 temperature-humidity sensor have inherent accuracy limitations, impacting precise measurement under extreme conditions. LCD displays required manual calibration and compatible libraries to function correctly. Power supply stability was critical, as fluctuations could affect both sensor readings and relay operations. Additionally, Blynk IoT relies on external servers, making the system dependent on internet availability for real-time remote monitoring. These constraints highlighted the need for careful component selection, calibration, and robust coding practices to ensure reliable and consistent system performance.

