National Textile University NTU

Name:	**Muhammad Abdullah, Muhammad Kaif, Faaiz Imtiaz**
Reg No:	**22-NTU-CS-1358,1360,1364**
Course Code and title	**CSE-4079 and Internet of Things**
Submitted To:	**Mr. Nasir Mehmood**
Date of Submission:	**12-3-2025**
Department:	**Computer Science**

Table of Contents

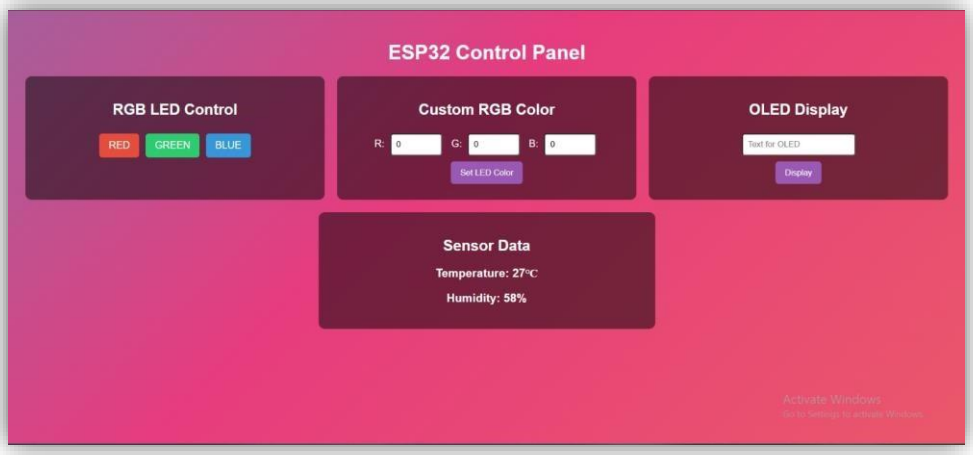
Abstract

This report presents an IoT project titled **Environmental Control System**, developed using the **ESP32 microcontroller**. The project provides a web interface for controlling RGB LEDs, displaying messages on an OLED screen, and monitoring environmental data through a DHT11 sensor. It integrates Wi-Fi connectivity using both **STA and AP modes**, making it accessible through local and external networks.

1. Introduction

The **ESP32 microcontroller** is known for its powerful processing capabilities, built-in Wi-Fi, and support for various peripherals. This project aims to create a smart **Environmental Control System** that combines RGB LED control, message display via OLED, and sensor data monitoring.

The system uses socket programming to create a web server that allows users to interact with the system through a browser. The OLED displays user messages, while the RGB LED offers visual feedback. The DHT11 sensor provides real-time temperature and humidity readings, which are made accessible via the web interface.



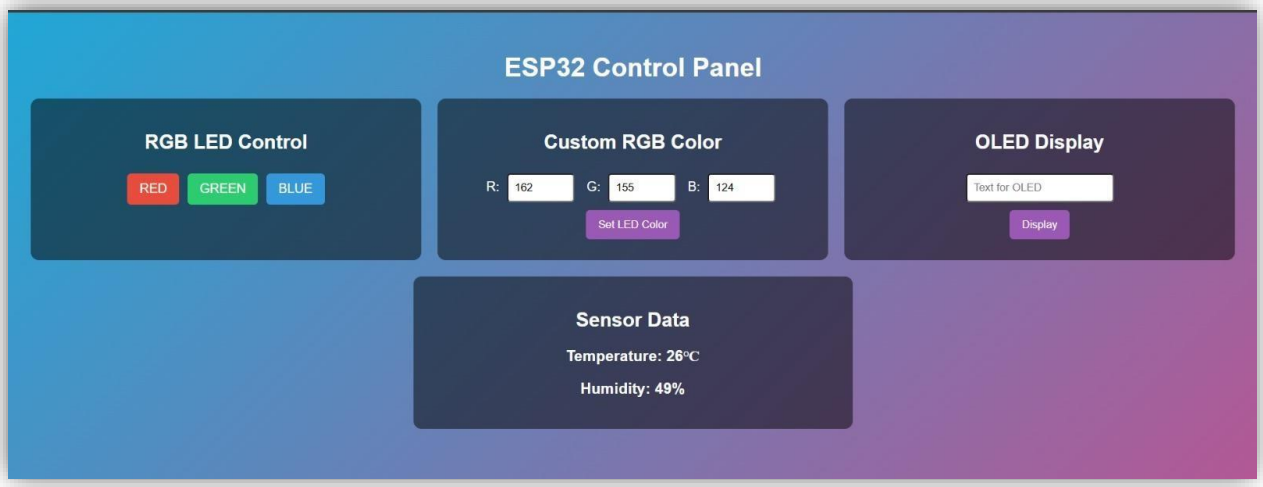
2. System Architecture

2.1 Network Configuration

- **STA Mode**: Allows the ESP32 to connect to a specified Wi-Fi network for external communication.
- **AP Mode**: Provides an access point for direct device communication without an external router.

2.2 Hardware Integration

- **NeoPixel RGB LED:** Controlled via HTTP requests to change colors and intensities.
- **SSD1306 OLED Display**: Displays text messages entered by the user through the web interface.
- **DHT11 Sensor**: Measures temperature and humidity, which is displayed on the web interface.

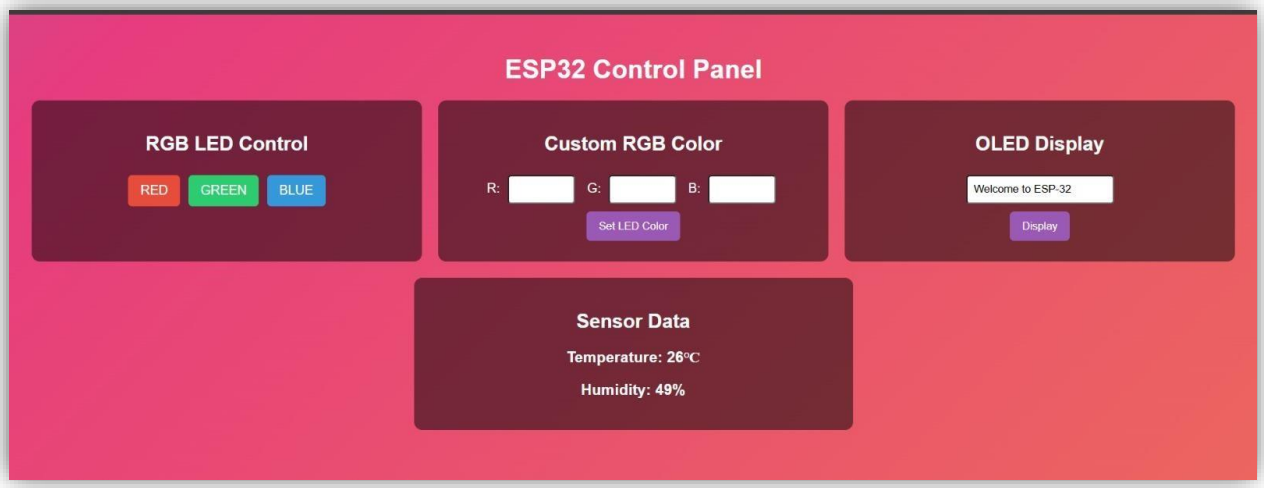


3. Implementation Details

3.1 Web Server Development

The web server runs on port 80 and supports various requests:

- **RGB Control**: Change LED color using predefined buttons or custom RGB values.
- **OLED Display Control**: Accept text input from the user to display messages on the OLED screen.
- **DHT11 Data Handling**: Provides real-time temperature and humidity readings.



3.2 Sensor Data Handling

The system reads data from the DHT11 sensor and displays it in real-time on the web interface. Sensor data is fetched using AJAX to ensure smooth and continuous updates.

3.3 RGB LED Control

The NeoPixel LED is controlled by predefined buttons for **Red**, **Green**, and **Blue**. Custom RGB values can also be set via user input, providing flexible control.

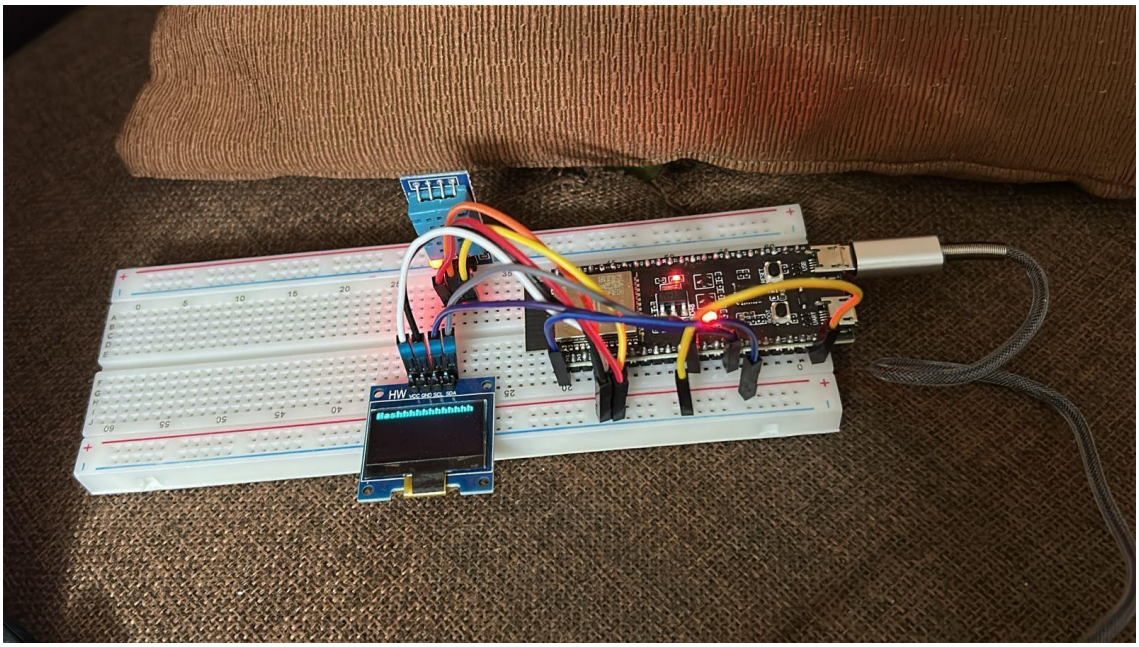
3.4 OLED Display Control

Messages entered via the web interface are displayed on the OLED screen. The display automatically adjusts to fit multiple lines of text.

4. Results & Discussion

The **Environmental Control System** successfully integrates:

- Real-time temperature and humidity monitoring.
- Dynamic control of RGB LED through HTTP requests.
- User-defined message display via OLED.
- Simultaneous AP and STA mode operations.



4.1 Technical Challenges

- Establishing simultaneous Wi-Fi connections using STA and AP modes.
- Managing HTTP requests and handling AJAX calls for sensor data.

4.2 Solutions

- Optimizing socket communication and enhancing compatibility with various devices.
- Implementing proper error handling mechanisms to ensure reliable operation.

5. Conclusion

The **Environmental Control System** demonstrates the versatility of the ESP32 microcontroller for creating interactive IoT projects. Future improvements can include:

- Adding authentication for device security.
- Enhancing user interface design.
- Adding more sensors for expanded monitoring capabilities.

6. References

- **ESP32 Documentation**
- **Socket Programming Fundamentals**
-**W3Schools for Web Design**