



Class	BSAI – 6th
Course Name	Intro to IOT
Submission Date	12 th March,2025.
Submitted To	Sir Nasir

Group Members:

Subaina Norab 22-NTU-CS-1374

Shaham Hijab 22-NTU-CS-1373

Hadia Alvi 22-NTU-CS-1343

Homepage

The Homepage of the ESP32-S3 Web Server provides an intuitive interface for controlling and monitoring the device. It offers essential functionalities like RGB LED control, OLED display text input, and real-time sensor data monitoring.

Functionalities:

The homepage includes the following key features:

- ❖ **RGB LED Control:** Allows users to adjust the color of the ESP32's NeoPixel LED in real time.
- ❖ **OLED Display Input:** Enables users to send text to be displayed on the ESP32's OLED screen.
- ❖ **Real-Time Sensor Data:** Displays temperature and humidity readings from the connected DHT11 sensor.

Features:

1. RGB LED Control

Purpose:

- Provides users with the ability to change the color of the ESP32's NeoPixel LED.
- Helps in debugging and visual feedback for different states of the device.

Implementation:

- Uses a range slider interface for selecting Red (R), Green (G), and Blue (B) values.
- Sends an HTTP request (GET /setRGB?r=R&g=G&b=B) to update LED color in real time.
- The ESP32 processes the request and updates the NeoPixel LED accordingly.

2. OLED Display Input

Purpose:

- Allows users to enter custom text and display it on the OLED screen connected to the ESP32.
- Provides real-time feedback and debugging support for developers.

Implementation:

- The homepage includes a text input field where users can type messages.
- Upon submission, the text is sent to the ESP32 via (GET /displayText?text=your_message).
- The ESP32 updates the OLED screen to show the received message.

3. Real-Time Sensor Data (DHT11)

Purpose:

- Displays live temperature and humidity data from the DHT11 sensor.
- Helps users monitor environmental conditions remotely.

Implementation:

- The ESP32 reads the DHT11 sensor periodically.
- Data is updated on the web interface using an HTTP request (GET /sensorData).
- The JSON response is processed to display real-time values for temperature and humidity.

Developer Page:

The Developer Page is designed to provide advanced functionalities for ESP32-S3 developers. It includes key features such as file system exploration, system monitoring, encryption-decryption.

Functionalities:

We are using in this page are:

- Files Display: With the help of this we can easily see which files are stored in esp32
- System Info: It will help us to know how much ram is unused and how much its utilizing
- Encryption-Decryption: For safety reasons, it will display encrypted data on oled. Which we can also display in decrypted form

Explanation:

❖ Files Display:

- Purpose:

Displays all files stored in the ESP32's flash memory.

Help developers track available resources and manage storage efficiently.

- **Implementation:**

Uses the os module to list files saved in the ESP32.

Provides an HTTP route (GET /files) to display file names.

❖ System Info

- Purpose:

Help developers track ESP32's memory usage in real time.

Displays total RAM, used RAM, and free RAM.

- **Implementation:**

Uses esp32.idf_heap_info (0) to get accurate ESP32 heap memory details.

Calculates used RAM by subtracting free RAM from total RAM.

Encryption-Decryption:

- **Purpose:**

Provides secure data transmission by encrypting sensitive information.

Displays encrypted data on the OLED screen for additional security.

- **Implementation:**

Uses XOR-based encryption for simplicity.

Stores encrypted text and allows users to decrypt it later.

Conclusion

Together, the **Developer Page** and **Homepage** provide a comprehensive environment for interacting with ESP32-S3. The Developer Page is tailored for advanced debugging, file management, and security, while the Homepage offers intuitive controls for LED adjustments, OLED text display, and real-time sensor monitoring. These features enhance usability, making ESP32-S3 an efficient and versatile development platform for IoT applications.