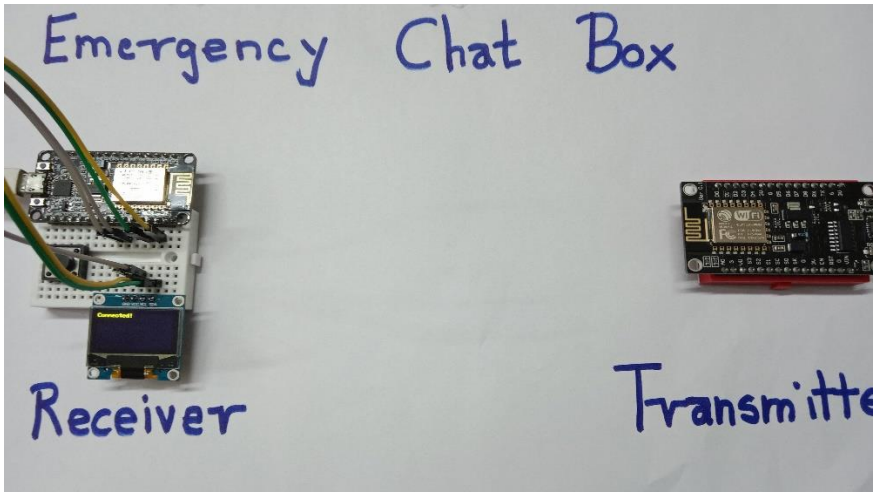
**EMERGENCY CHAT BOX:**

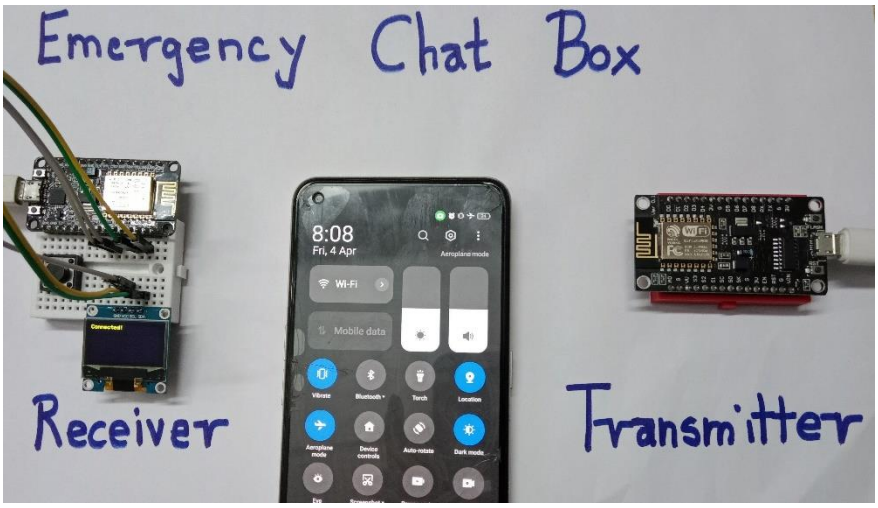
**COMPONENTS:**

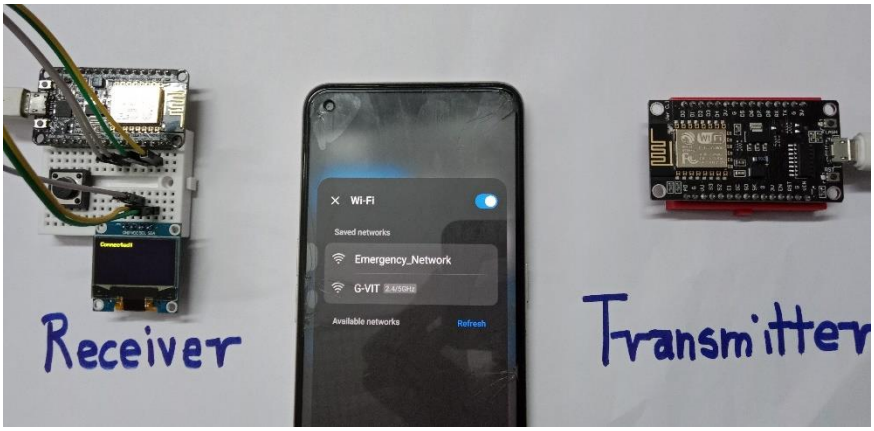
- ESP8266 (NodeMCU) – Used for both Sender & Receiver.
- OLED Display (SSD1306, 128x64) – Displays incoming messages.
- Push Button – Used for message acknowledgment.

**PHOTOS OF WORKING MODEL:**

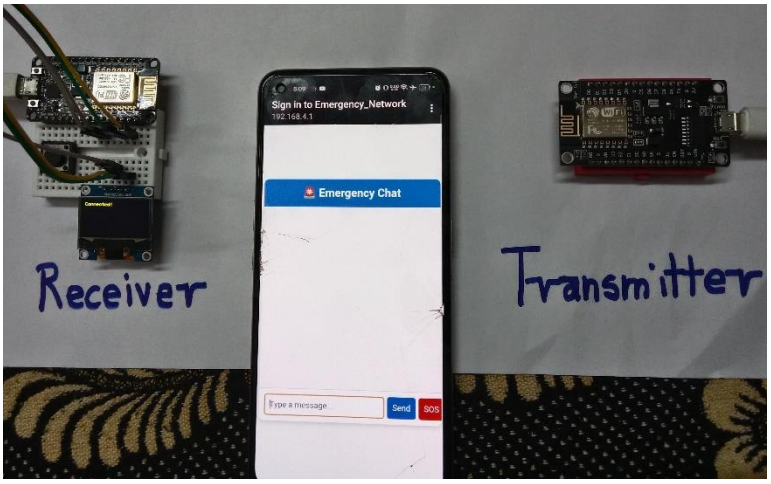
The Transmitter (Sender) ESP8266 creates a local Wi-Fi Access Point named "Emergency_Network", enabling devices—even in Airplane Mode—to connect without internet. Users access a captive web portal to send messages directly to the Receiver via WebSocket communication.

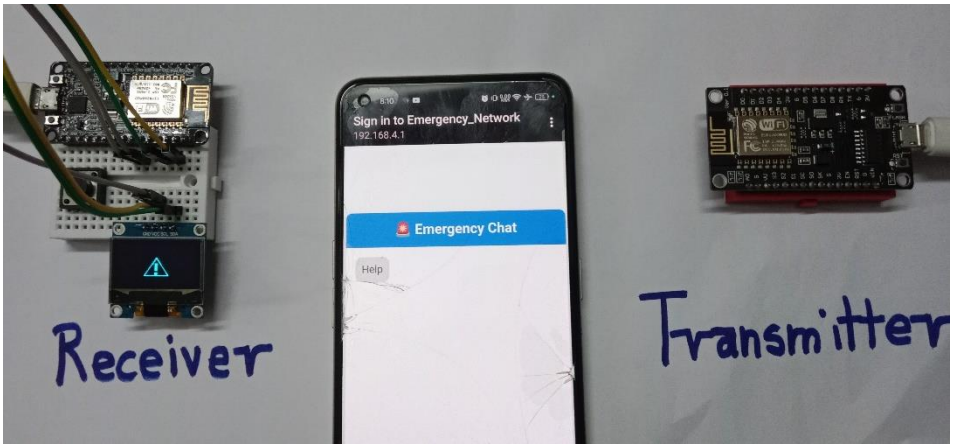


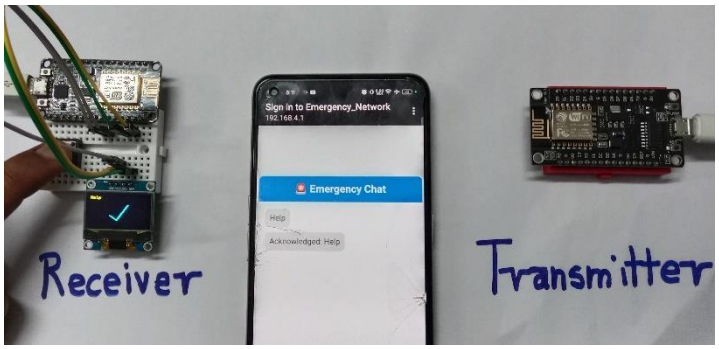



When the user connects to the "Emergency_Network" Wi-Fi, a captive web portal opens automatically for message input. Upon clicking the "Send" button, the message is transmitted to the receiver, which displays an alert symbol on the OLED screen to indicate a new emergency message.




When the receiver presses the push button, an acknowledgment is sent back to the sender confirming that the message has been received. The OLED display shows a tick symbol to indicate successful acknowledgment, followed by displaying the previously received message while the system waits for the next incoming message.




## DESCRIPTION:

This project introduces an **Emergency Chat Box** using **ESP8266 Wi-Fi modules**, which enables **offline** communication between a **Sender** and a **Receiver** using a **Captive Portal and WebSockets**. The system operates **without an internet connection** and can function even in **Aeroplane Mode**, making it ideal for emergency scenarios. The sender establishes a **Wi-Fi Access Point (AP)** and hosts a **captive portal** that redirects users to a **chat interface**, allowing them to send messages to a receiver ESP8266. The receiver **displays incoming messages on an OLED screen** and provides an acknowledgment system through a **push button**.

## WORKING:

### Sender ESP8266: Message Transmission & Web Hosting

The Sender ESP8266 serves as a Wi-Fi Access Point (AP) and runs a web server that hosts the chat interface.

- **Wi-Fi Network Creation:** The sender ESP8266 operates in AP mode, creating a Wi-Fi network (e.g., "Emergency _ Network").
- **Captive Portal Setup:** A DNS server is configured to redirect all domain requests to the ESP8266's IP address. When a user connects to the network, their browser is automatically redirected to the chat interface.
- **Webpage Hosting:** The ESP8266 serves an HTML webpage containing a text input box and a send button.
- **Message Transmission Using WebSockets:** When the user enters a message and clicks "Send," the message is sent to the receiver ESP8266 via WebSockets .WebSockets ensure low-latency, real-time communication between the sender and receiver.
- **Acknowledgment Handling:** The sender listens for acknowledgment messages from the receiver. Upon acknowledgment, the webpage updates the status to **"Acknowledged : Message"**.

### Receiver ESP8266: Message Display & Acknowledgment

The Receiver ESP8266 connects to the sender's Wi-Fi network, listens for messages, and displays them on an OLED screen.

**Wi-Fi Connection:** The receiver ESP8266 connects to the sender's Wi-Fi AP.

- **Listening for Messages:** The receiver establishes a WebSocket connection with the sender. It continuously listens for incoming messages.

- **Displaying Messages on OLED Screen:** When a message is received, a warning / danger symbol will blink on the OLED display, until the receiver presses the push button to view and acknowledge the message.

- **User Acknowledgment:** The receiver has a push button for message acknowledgment. When the button is pressed, a tick symbol is displayed along with the received message.

- **Clearing the Display:** Once acknowledged, the message on the OLED screen is cleared, ready to receive the next message, and the previously acknowledged message will also display on the top of the Screen.

The Receiver ESP8266 handles multiple messages using a **queueing system**, ensuring messages are processed in the order they arrive. The first message is displayed on the **OLED screen**, while others wait in the queue. When the **push button is pressed**, the current message is removed, and the next message is displayed. This ensures **sequential acknowledgment** without overwriting, making the system reliable for emergency communication.

## CONCLUSION:

This project presents a self-contained emergency chat system using ESP8266 modules, designed for situations where internet or cellular networks are unavailable—such as in disaster zones, remote areas, or airplane mode. The sender ESP8266 creates a Wi-Fi Access Point with a captive portal, allowing up to four users simultaneously to connect via their mobile devices and send emergency messages through a web interface. These messages are transmitted in real-time using WebSockets to the receiver, where they are queued, displayed on an OLED screen, and acknowledged via a push button. This ensures reliable, sequential handling of multiple messages, enabling efficient communication when it matters most.

## LIMITATIONS AND IMPROVEMENTS NEEDED:

 **Limited Range**: The system currently operates over short-range Wi-Fi.
→ *Improvement*: Integration of ESP-NOW or LoRa to extend communication coverage in outdoor or large-area deployments.

 **Local-Only Communication**: The system cannot transmit messages beyond the local Wi-Fi network.
→ *Improvement*: Adding GSM or satellite communication modules for long-distance or global connectivity.

 **Manual Acknowledgment Required**: Messages must be acknowledged by pressing a physical button.
→ *Improvement*: Implementation of haptic or audio feedback and consider automatic acknowledgment features to improve responsiveness.

 **User Capacity Limited to Four Devices**: Although four users can connect simultaneously, handling more users may lead to latency.
→ *Improvement*: Optimize server-side code and memory management to scale for more concurrent users or staggered message submission.

 **Power Dependency**: Continuous power is required for operation.
→ *Improvement*: Use deep sleep modes, energy-efficient OLEDs, and battery packs to support longer operation in the field.