**FLOOD MONITORING AND SYSTEM**

**Team** **Member**: Weis Meuriel Gifta

**Exam** **Number**: au210521106059

**Domain**:IOT

**INTRODUCTION**:

Flood monitoring with ESP32 is a critical application that leverages the capabilities of this versatile microcontroller to detect and respond to potential flooding events. By utilizing various sensors and communication modules, the ESP32 can collect real-time data on water levels, weather conditions, and more. This data can be analyzed and shared to provide early warnings and aid in flood mitigation efforts, making it an invaluable tool for disaster management and public safety. In this context, I can provide more information on how to set up flood monitoring using ESP32 if you’d like.

**Phase** 2- **Innovation**

Designing and implementing a flood monitoring system using an ESP32 involves several steps. Here’s a high-level overview of the process:

Step1: **Hardware** **Selection**:

Choose appropriate sensors for flood monitoring, such as water level sensors, rain gauges, and weather sensors. Select an ESP32 development board with built-in Wi-Fi or LoRa capabilities for data transmission.Consider a power source, such as a battery or solar panel, to ensure continuous operation.

Step 2: **Sensor** **Integration**:

Connect the selected sensors to the ESP32 using the appropriate interfaces (analog, digital, I2C, etc.).Write code to read data from these sensors. Ensure accuracy and reliability.

Step 3: **Data** **Collection**:

Program the ESP32 to collect data from the sensors at regular intervals. Implement error handling and data validation to ensure data accuracy.

Step 4: **Data** **Transmission**:

Use Wi-Fi or LoRa (Long-Range) communication to transmit data to a central system or the cloud.Implement secure data transmission protocols if necessary.

Step 5:**lasso**

LASSO (Locally Adaptive Sparsity prior for Structured Optimization) is primarily a machine learning algorithm used for regression and feature selection. It may not be directly applied in a straightforward manner on a microcontroller like the ESP32 due to computational and memory constraints. However, machine learning models can be trained using more powerful hardware (e.g., servers, cloud services) and then deployed onto the ESP32 for inference.

Step 6: **Data** **Storage** **and** **Analysis**:

Set up a central system or cloud platform to receive and store the data. Implement data analysis algorithms to detect flood conditions or trends in the data.

Step 7: **Alerting** **System**:

Develop an alerting mechanism that triggers when flood conditions are detected Notifications can be sent through SMS, email, or other means.

Step 8: **Power** **Management**:

Ensure efficient power management to extend the ESP32’s battery life or use alternative power sources like solar panels.

Step 9: **User** **Interface** :

Create a user interface for monitoring the system and viewing historical data.

Step 10: **Testing** **and** **Calibration**:

Thoroughly test the system under different conditions to ensure its reliability and accuracy.Calibrate the sensors if needed to improve data accuracy.

Step11: **Deployment**:

Install the flood monitoring system in flood-prone areas, ensuring proper mounting, waterproofing, and protection from environmental factors.

Step 12: **Logistic**

Implementing a logistic regression model on an ESP32 for flood monitoring involves several steps. Logistic regression is commonly used for binary classification problems, making it suitable for predicting whether a flood event is likely or not based on input features. Here’s a general guide:Data Collection: Gather historical data related to flood events and relevant environmental parameters. This dataset should include instances labeled with whether a flood occurred or

Step13**: Maintenance** **and** **Monitoring**:

Regularly maintain and monitor the system to ensure it continues to operate effectively.

Step 14: **Data** **Visualization**:

Create dashboards or reports to visualize the data, making it easier to interpret and respond to flood conditions.Remember that designing and implementing a flood monitoring system is a complex task. Depending on your specific requirements and environmental conditions, the design may need to be customized. Additionally, ensure compliance with any local regulations and consider working with experts in the field of flood monitoring for a comprehensive and effective system.

**Program**

#include <WiFi.h>

Const char\* ssid = “YourWiFiSSID”;

Const char\* password = “YourWiFiPassword”;

Const char\* serverIP = “YourServerIP”;

Const int serverPort = 80;

Const int sensorPin = 2; // Replace with the GPIO pin connected to your water level sensor

Bool floodDetected = false;

Void setup() {

Serial.begin(115200);

WiFi.begin(ssid, password);

While (WiFi.status() != WL\_CONNECTED) {

Delay(1000);

Serial.println(“Connecting to WiFi…”);

}

Serial.println(“Connected to WiFi”);

}

Void loop() {

Int waterLevel = digitalRead(sensorPin);

If (waterLevel == HIGH) {

If (!floodDetected) {

floodDetected = true;

sendFloodAlert();

}

} else {

floodDetected = false;

}

Delay(1000); // Check the sensor every second

}

Void sendFloodAlert() {

WiFiClient client;

If (client.connect(serverIP, serverPort)) {

Client.print(“GET /flood\_alert HTTP/1.1\r\n”);

Client.print(“Host: “);

Client.print(serverIP);

Client.print(“\r\n\r\n”);

Delay(10);

Client.stop();

Serial.println(“Flood Alert Sent”);

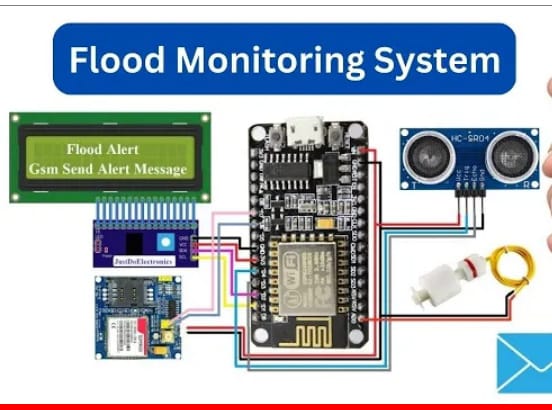
} else {

Serial.println(“Connection failed”);

}

}

**Diagram:**



**Conclusion**

In conclusion, flood monitoring using an ESP32 can be a valuable and potentially life-saving application. This system allows for the real-time monitoring of water levels and the detection of potential floods.Flood monitoring systems can significantly contribute to early warning and disaster management, helping to mitigate the impact of floods on communities and infrastructure. Customization and scalability are key, allowing the system to be tailored to the specific needs of the environment in which it’s deployed.