FLOOD MONITORING AND EARLY WARNING

DATE	**25-10-2023**
TEAM ID	**539**
PROJECT NAME	**Flood Monitoring And Early Warning using IOT**

PHASE 3: Development Part -1 – Building Flood Monitoring And Early Warning using IOT project.

PROJECT OVERVIEW:

In regions susceptible to flooding, the need for a reliable Flood Monitoring & Early Warning System is undeniable. The central challenge of this project is to design and implement an IoT-based system that can accurately monitor water levels, weather conditions, and historical data to issue timely flood warnings. This system must contend with the complexity of flood prediction, taking into account numerous variables, and deliver early alerts that empower communities and authorities to mitigate the devastating effects of flooding.

BUILDING A FLOOD MONITORING PROJECT WITHSENSORS:

REAL TIME FLOOD MONOTORING INFORMATION:

1.Water level Monitoring :

- Water level monitoring with an ultrasonic sensor involves measuring liquid levels by sending and receiving sound waves. The sensor emits sound pulses and calculates the distance to the liquid's surface based on the echo. This non-contact method provides accurate and efficient monitoring for various applications, such as tanks, reservoirs, and water systems**.**

2. Location Monitoring :

- In flood monitoring systems, GPRS sensors are used to track the location of monitoring stations and transmit data over cellular networks. These sensors enable real-time data collection, helping authorities respond promptly to flood events, enhance disaster management, and protect vulnerable areas by providing critical location-based information.

3. **Thingspeak cloud:**

    - IoT devices use Wi-Fi connectivity to send real-time data to the ThingSpeak cloud platform. Through secure wireless connections, sensor information is transmitted and visualized on ThingSpeak dashboards, enabling remote monitoring and analysis for applications like environmental sensing, home automation, and industrial control.

BENEFITS:

Implementing IoT for environmental monitoring offers several benefits:

- **Real-time Data:** Instant access to data for timely decision-making.
- **Data Accuracy:** High precision and reliability in data collection.
- **Sustainability:** Facilitating eco-friendly practices and reducing environmental impact.
- **Public Awareness:** Engaging the public in environmental issues throughaccessible data.
- **Research and Policy Support:** Supporting research and government policies with reliable environmental data.

PROJECT PLANNING:

Define the scope, objectives, budget, and timeline for the project.

PROJECT COMPONENTS:

SENSOR SELECTION:

- We have selected a range of sensors, including ultrasonic sensor, GPRS sensor, GSM Module.
- These sensors will provide comprehensive data on environmental conditions within the park.

DEPLOYMENT OF IOT:

- The deployment of the Internet of Things (IoT) in flood monitoring and early warning in reserve areas is a powerful strategy to enhance the management, conservation, and enjoyment of natural resources and public spaces.

MICROCONTROLLER SELECTION:

- The ESP32 as is used in this project. The ESP32 offers robustprocessing power and GPIO capabilities, allowing us to collect, process, and transmit data effectively.

SENSOR INTEGRATION:

- Sensor integration in flood monitoring in water prone areas is essential for collecting accurate and comprehensive data to better manage and preservenatural resources.

- Integrating a variety of sensors can provide a holistic view of the nature ecosystem.

DATA COLLECTION AND PROCESSING:

- Data collection and preprocessing are crucial steps in flood monitoring and early warning.
- Accurate and reliable data collection and careful preprocessing ensure that the data is of high quality, which is essential for informed decision-making and effective management.

DATA TRANSMISSION:

- Data transmission in flood monitoring and early warning is a critical aspect of the monitoring process.

- Data is transmitted to the thingspeak cloud through Wi-Fi
- It involves the transfer of collected environmental data from sensors and monitoring devices to a central data repository or control center where it can be processed, analyzed, and utilized for various purposes, such as decision-making, research, and public information.

REAL TIME MONITORING:

Real-time monitoring in Flood monitoring and early warning involves the continuous and immediate collection, analysis, and dissemination of data on various natural parameters

The real time processed data is shown as the continuous graph in the thingspeak cloud platform channel.

DATA VISUALIZATION:

Data visualization in Flood monitoring and early warning is a crucial aspect of turning complex and voluminous data into clear and meaningful insights.

Effective data visualization helps officers, rescue team, and the public better understand the environmental conditions.

TESTING AND CALIBRATION:

- Testing and calibration are essential processes in in Flood monitoring and early warning to ensure that the data collected is accurate, reliable, and consistent.
- Proper testing and calibration procedures help maintain the integrity of monitoring.

PYTHON SCRIPT DEVELOPMENT

- Develop a Python script that will run on the IoT for in Flood monitoring and early warning
- This script should be responsible for the following tasks:
- Formatting the collected data for transmission to the IoT platform.
- Managing secure device communication with the IoT platform.
- Implementing security measures, such as data encryption and device authentication.

PYTHON PROGRAM:

```
import machine
import time
import urequests

TRIGGER_PIN = 18  # GPIO 4 for the trigger pin
ECHO_PIN = 5     # GPIO 5 for the echo pin
MAX_DISTANCE = 200
WIFI_SSID = "logesh 5G"
WIFI_PASSWORD = "logesh0404"
THINGSPEAK_API_KEY = "KHGNROUJVWTLMCHH"
THINGSPEAK_CHANNEL_NUMBER = 2303636  # Replace with your ThingSpeak channel number

duration = 0
distance = 0

# Set up Wi-Fi connection
import network
sta = network.WLAN(network.STA_IF)

sta.active(True)
sta.connect(WIFI_SSID, WIFI_PASSWORD)

while not sta.isconnected():
    time.sleep(1)
    print("Connecting to WiFi...")

print("Connected to WiFi")

# Function to measure distance using the ultrasonic sensor
def measure_distance():
    trigger = machine.Pin(TRIGGER_PIN, machine.Pin.OUT)
    echo = machine.Pin(ECHO_PIN, machine.Pin.IN)

    trigger.off()
    time.sleep_us(2)
    trigger.on()
```

```
time.sleep_us(10)
    trigger.off()

    while echo.value() == 0:
        pulse_start = time.ticks_us()

    while echo.value() == 1:
        pulse_end = time.ticks_us()

    pulse_duration = time.ticks_diff(pulse_end, pulse_start)
    distance_cm = (pulse_duration / 2) / 29.1  # Calculate distance in
centimeters

    return distance_cm

# Main loop
while True:
    distance = measure_distance()

      if (distance < 100.0):
    print("Water level is low.")

  else if ( distance >100 && distance < 250.0):
   print("flood alert")


  else
  print("FLOOD!!.")



        # Update ThingSpeak with the distance data
        data = {'api_key': THINGSPEAK_API_KEY, 'field1': distance}
        response = urequests.post("https://api.thingspeak.com/update",
json=data)

        if response.status_code == 200:
            print("Data sent to ThingSpeak successfully.")
        else:
            print("Error sending data to ThingSpeak. HTTP error code:",
response.status_code)

        response.close()
    else:
        print("Sensor error - distance not obtained")

    time.sleep(1)  # Adjust this delay as needed
```