FLOOD MONITORING AND EARLY WARNING SYSTEM

PHASE 4 : DEVELOPMENT PART 2

TEAM MEMBER

NAME : Narayani C

REG NO : 211121106008

1. Data Collection:

Gather data from various sources, such as weather stations, river sensors, satellite imagery, and historical flood records.

2. Feature Engineering:

Extract relevant features from the collected data, like rainfall, river levels, soil moisture, and more. This helps in building a comprehensive dataset for training.

3. Data Preprocessing:

Clean and preprocess the data, handling missing values, outliers, and normalizing data for consistency.

4. Model Selection:

Choose appropriate machine learning models for flood prediction. Common choices include decision trees, random forests, or deep learning models.

5. Model Training:

Train the selected model using historical data, including both flood and non-flood events. This helps the model learn patterns and make predictions.

6. Validation:

Assess the model's performance using validation data to ensure it's accurate and reliable.

7. Integration of Sensors:

If applicable, integrate sensors and monitoring devices to continually collect real-time data.

8. Threshold Setting:

Define threshold values for different data parameters to trigger flood warnings.

9. Early Warning System:

Develop a system that processes real-time data and activates alerts or warnings when the model predicts an impending flood event based on predefined thresholds.

10. Communication and Alerts:

Implement a communication system to disseminate warnings to relevant authorities and the public through various channels, such as text messages, sirens, or mobile apps.

11. Community Engagement:

Educate and involve local communities in the warning system to ensure a timely and appropriate response.

12. Regular Maintenance:

Continuously monitor and maintain the system, updating it as new data becomes available or as the model's performance changes.

PROGRAM :

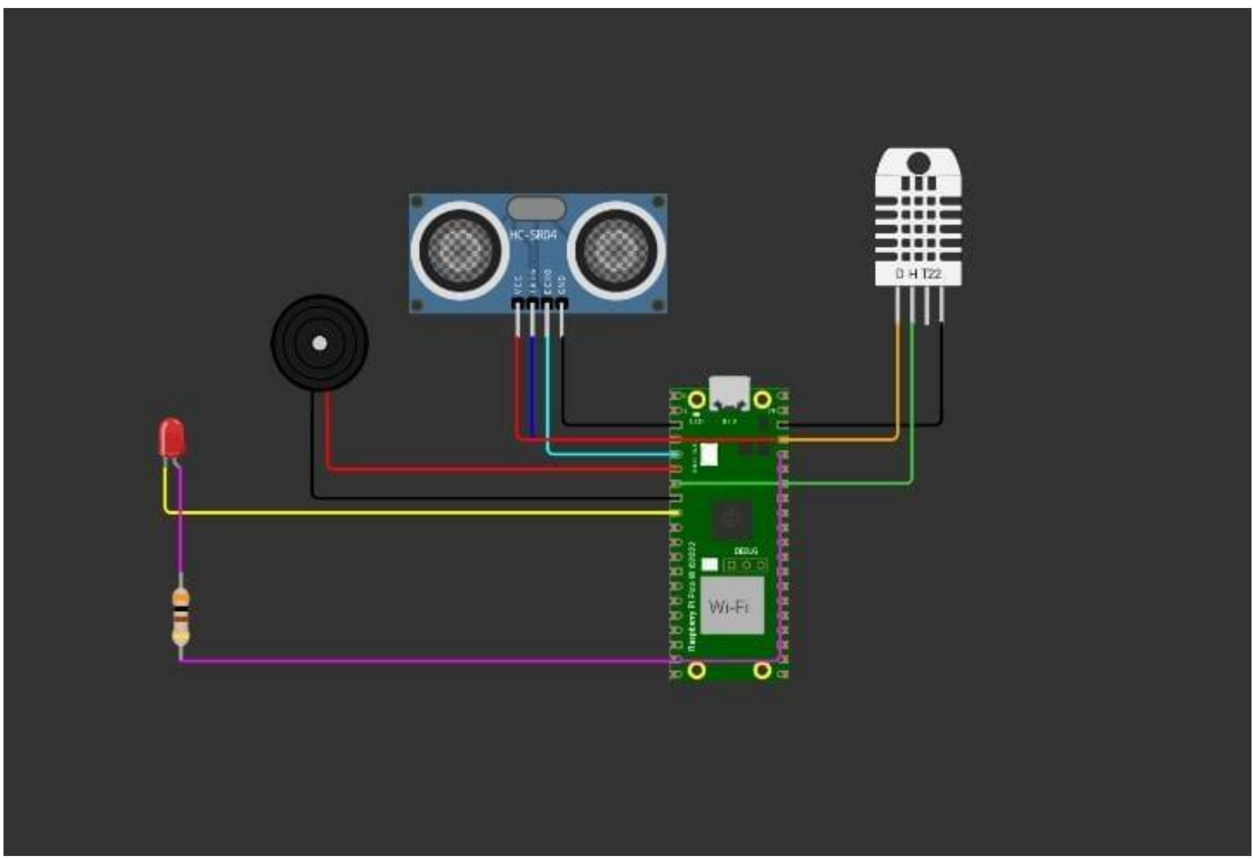
```
import time

import machine

import dht
```

```
# Define GPIO pins
TRIG_PIN = machine.Pin(2, machine.Pin.OUT)
ECHO_PIN = machine.Pin(3, machine.Pin.IN)
BUZZER_PIN = machine.Pin(4, machine.Pin.OUT)
DHT_PIN = machine.Pin(5)
LED_PIN = machine.Pin(6, machine.Pin.OUT)

def distance_measurement():
    # Trigger ultrasonic sensor
    TRIG_PIN.on()
    time.sleep_us(10)
    TRIG_PIN.off()

    # Wait for echo to be HIGH (start time)
    while not ECHO_PIN.value():
        pass
    pulse_start = time.ticks_us()

    # Wait for echo to be LOW (end time)
    while ECHO_PIN.value():
        pass
    pulse_end = time.ticks_us()

    # Calculate distance
    pulse_duration = time.ticks_diff(pulse_end, pulse_start)
```

```
    distance = pulse_duration / 58  # Speed of sound (343 m/s) divided by 2

    return distance

def read_dht_sensor():
    d = dht.DHT22(DHT_PIN)
    d.measure()
    return d.temperature(), d.humidity()

buzz_start_time = None  # To track when the buzzer started

while True:
    dist = distance_measurement()
    temp, humidity = read_dht_sensor()

    # Check if the distance is less than a threshold (e.g., 50 cm)
    if dist < 50:
        # Turn on the buzzer and LED
        BUZZER_PIN.on()
        LED_PIN.on()
        status = "Flooding Detected"
        buzz_start_time = time.ticks_ms()
    elif buzz_start_time is not None and time.ticks_diff(time.ticks_ms(), buzz_start_time) >= 60000:  # 1 
minute
        # Turn off the buzzer and LED after 1 minute
        BUZZER_PIN.off()
```

```
        LED_PIN.off()
        status = "No Flooding Detected"
    else:
        status = "No Flooding Detected"

    print(f"Distance: {dist:.2f} cm")
    print(f"Temperature: {temp:.2f}°C, Humidity: {humidity:.2f}%")
    print("Status:", status)

    time.sleep(2)
```

SCHEMATIC DIAGRAM:



CONCLUSION AND RECOMMENDATIONS :

FWMS has been developed to decrease the rate of property destructions and deaths due to flood disaster. The system is able to call and send warning SMS to the user and Fire and Rescue Department when the flood water level has reached warning and danger levels. This system provides two-way communication system. The user can send a request SMS to monitor the flood water level and control the system by turning on or off the system. It is embedded with a buzzer that will be activated when the flood water level has reached over the warning and danger levels.

For future works, it is highly recommended for the future researchers to add a functional GPS sensor for locating the flood-plain area and inform to the Fire and Rescue Department when the flood water level reaches dangerous level. This is because; the current developed project does not embed with the GPS sensor to locate the location of the system.

Furthermore, the future researchers can also build a simple mobile application for the user to monitor the real-time flood water level easily. The mobile application can include the graph of ultrasonic sensor reading when the flood water level increases and record the date and time of the occurred event.