FINAL PROJECT REPORT

IOT WEATHER AND SAFETY MONITOR

By

Umama Bin Rashid (07)

Abdul Haseeb Nasar (05)

Abdul Rafey (01)

Abdul Samad Khan (08)

From

NAVTCC

SSUET

Internet Of Things Application and Development

Batch 04 Section A

(Sir Mooazzam)

(Sir Saad)

IOT WEATHER AND SAFETY MONITOR

Introduction

IOT Weather And Safety Monitor is an IoT device which can be use to monitor environment of a particular place or space. It's function is to monitor various environmental changes and perform different functions against these changes accordingly. It consists of Weather and humidity sensor (DHT11) through which theweather and humidity of a particular place is sensed in the microcontroller and then the data is sent to cloud where the analysis of the given data takes place, the other important sensor is the tilt sensor (SW-520D) through which we can detect the earthquake and it also consists of Water sensor through which rain can be sensed, flame sensor have also included in the respected device which allow this device to detect fire in the surroundings . After the collection of all the data from the above mentioned sensors in the Microcontroller the data is then send to cloud or web API i.e. ThingSpeak.

Components

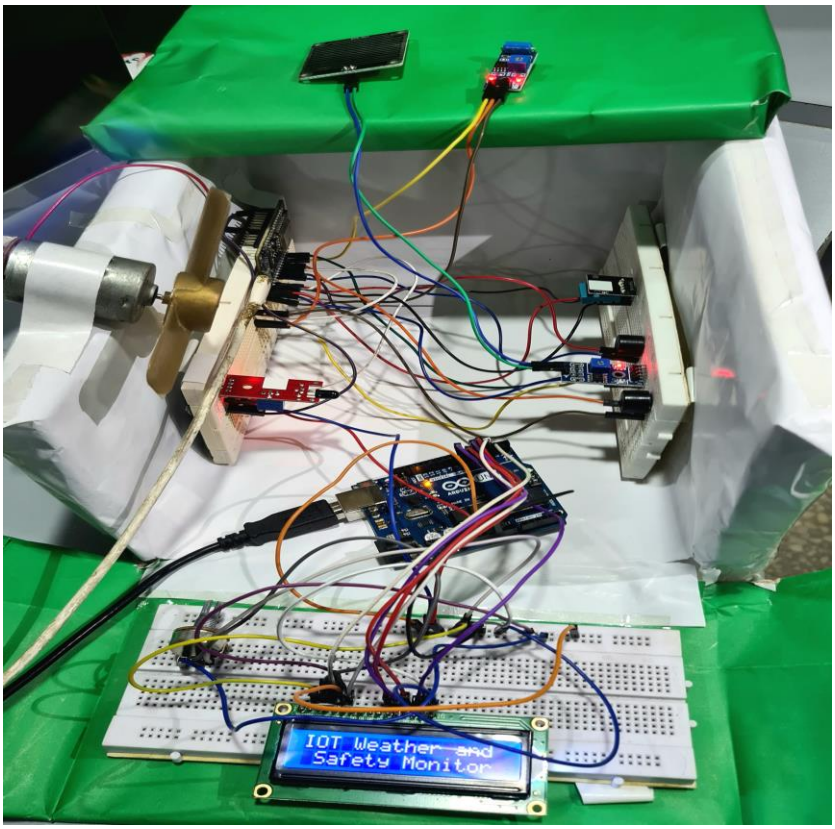
- DHT11 (Temperature and humidity sensor)
- SW-520D (tilt sensor)
- AB007 (flame sensor)
- LM393 (Rain drop sensor)
- ESP 8266 Microcontroller
- Arduino UNO
- Breadboard
- Jumper wires
- LCD
- LEDs
- Buzzers

Code

GIT HUB LINK:

https://github.com/HaseebNasar/Final-Project/tree/main/IOT%20PROJECT

Project Picture



Arduino Code Explanation: Sensor Data Monitoring via ThingSpeak

1) Start

2) Initialize WiFi and connect to the network.

3) Initialize DHT sensor and pins for various sensors and alarms.

4) Loop Start

5) Read temperature and humidity from DHT sensor.

6) Update ThingSpeak field 1 with temperature value.

7) Update ThingSpeak field 2 with humidity value.

8) Read value from flame sensor.

9) If flame is detected (value <= 100):

- Turn on the flame alarm (D7).
- Update ThingSpeak field 3 with flame sensor value.

10) If flame is not detected:

- Turn off the flame alarm (D7).
- Update ThingSpeak field 3 with flame sensor value.

11) Read earthquake detection signal.

12) If earthquake detected:

- Turn on the earthquake alarm (alarm).
- Update ThingSpeak field 4 with earthquake detection value.

13) If earthquake not detected:

- Turn off the earthquake alarm (alarm).
- Update ThingSpeak field 4 with earthquake detection value.

14) Read rain detection signal.

15) If rain detected (rainState == 0):

- Turn on the rain alarm (alarm).
- Update ThingSpeak field 5 with rain detection value.

16) If rain not detected (rainState != 0):

- Turn off the rain alarm (alarm).
- Update ThingSpeak field 5 with rain detection value.

17) Loop End

18) Delay for stability and to avoid rapid changes.

19) End

WORKING

1. The code sets up the necessary libraries, pins, and objects.
2. It establishes a WiFi connection to your network.
3. It reads temperature and humidity data from the DHT11 sensor and sends it to ThingSpeak fields using HTTP requests.
4. It reads data from the flame sensor, earthquake sensor, and rain sensor, and sends these values to ThingSpeak fields as well.
5. The code includes appropriate delay intervals to avoid rapid changes and provide stability.

Advantages

1. Remote Monitoring:

The system enables remote monitoring of multiple environmental factors in real-time.

2. Early Warning:

It can provide early warnings for dangerous situations like fires, earthquakes, and heavy rain.

3. Data Analysis:

Data sent to ThingSpeak can be analyzed and visualized for trends and patterns.

4. Automation:

The system can trigger alarms and other actions based on sensor readings.

5. IoT Learning:

Building this project provides hands-on experience with IoT, sensors, and data communication.

Conclusion

We are building a multi-sensor monitoring system that communicates data to ThingSpeak for remote monitoring and analysis. The system can monitor temperature, humidity, flame presence, earthquake detection, and rain detection. The conclusion would summarize the purpose of your project, its potential applications, and the functionality demonstrated in the code.