SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

WINTER SEMESTER - 2022

IoT Based Weather Monitoring System

A Report

submitted by

Shreyas Singh
20BCE2182
Shivam Singhal
19BCE2112
Prasanna Kumar
20BCE2121

CSE3009 – Internet of Things – J Component

B2 Slot

Faculty Name – Dr.A.Saraswathi Priyadharshini,

Department of Analytics,

SCOPE

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

TABLE OF CONTENTS

Introduction

The Weather Monitoring system plays a large utility in varied areas from agricultural growth and development to industrial development.

Sensing the weather has been important to man over the centuries. The winds and other weather variables are of equal concern and can have an even greater impact on our modern, high-tech lifestyle.

A weather station is that facility on land or sea, which has instruments and devices for observing and measuring atmospheric parameters to provide the information for weather forecasts. Modern weather monitoring systems and networks are designed to make the measurements necessary to track these movements in a cost-effective manner.

LITERATURE SURVEY

Paper	Objective	Advantages	Disadvantages
WEATHER MONITORING SYSTEM FOR EFFECTIVE ANALYTICS	To implement and data visualization of weather parameters like temperature, humidity, PM 2.5 and PM 10 concentrations and AQI using the Raspberry Pi as server and data accessed over the intranet or internet in a specified subnet or world wide web.	The proposed system is the most compact unit for measuring weather parameters in regions suffering from PM 2.5 pollution. Helps to monitor the weather metrics corresponding to pollution over a centralized data analytics server.	The proposed system can be connected to the internet for the study from various locations.
WEATHER MONITORING SYSTEM FOR PRECISION AGRICULTURE	To intelligently select low cost sensors (without compromising with the accuracy and precision), collect data in real time, store the data in cloud, perform visualization of data and to carry out analytics on the data	The proposed system would also aid the farmers to carry out the agricultural tasks on real-time bases, which in turn helps them to use the agricultural resources in an efficient way and at the time when needed by the crops.	More sensors can be used to sense wind speed and direction and deploying the prototype into the agricultural field and monitor weather parameters and convert this prototype in a full-fledged Precision Weather Station
RASPBERRY PI BASED WEATHER REPORTING OVER IOT	A portable module that can produce accurate results up to a height of 3000m and a temperature range between 00C to 500C.	A portable module that can produce accurate results up to a height of 3000m and a temperature range between 00C to 500C.	Periodic data sensing is used. So data is stored or the buzzer is set off only when there is a drastic change from the normal values. Data is available only online which can be inaccurate at times due to time latency. Insufficient data collected that results in low accuracy

INTERNET OF THINGS FOR LOCAL WEATHER MONITORING	This paper proposes the compact Internet of Things (IoT) module for local weather monitoring. This IoT module was designed and developed to collect meteorological data for the purposes of power output prediction from photovoltaic panels and environmental data for the evaluation of the air quality in inhabited areas.	Measures Humidity, temperature, Barometric pressure, Visible spectrum, Infrared spectrum, Air quality accurately.	Needs constant power supply since the current version of the IoT module cannot run on battery because of its significant power consumption
INTERNET OF THINGS FOR LOCAL WEATHER MONITORING	This paper proposes the compact Internet of Things (IoT) module for local weather monitoring. This IoT module was designed and developed to collect meteorological data for the purposes of power output prediction from photovoltaic panels and environmental data for the evaluation of the air quality in inhabited areas.	Measures Humidity, temperature, Barometric pressure, Visible spectrum, Infrared spectrum, Air quality accurately.	Needs constant power supply since the current version of the IoT module cannot run on battery because of its significant power consumption

Proposed Work

The weather station will be used to determine 3 parameters namely temperature and humidity using the DHT11 and rain condition using the KG004 rain drop sensor. NodeMCU will be used to interface the sensors and the WIFI module embedded inside the NodeMCU helps in connecting with the ThingSpeak cloud
Temperature & Humidity will be detected by DHT11 sensor and rain value will be detected by KG004. The collected data is then sent to the system via NodeMCU. NodeMCU sends the data to Think Speak cloud which will graphically represent the data for the analysis.

Code

```
#include <ESP8266WiFi.h>                                    // NodeMCU Library
#include <DHT.h>                                    // DHT Sensor Library
#include <ThingSpeak.h>                                    // ThingSpeak Library
DHT dht(D3, DHT11);                                    // Declaring D3 as dht pin
WiFiClient client;                                    // WiFi client object for ThingSpeak
long myChannelNumber = 1700965 ;                                    // ThingSpeak channel
const char myWriteAPIKey[] = "W0CFRCUTCJ0TGQ76";                    // ThingSpeak
write key
int sensorPin = A0;                                    // Raindrop sensor input pin
int enable2 = 13;                                    // Raindrop sensor output led pin
int sensorValue2 = 0;                                    // Variable to store the value coming from
Raindrop sensor
bool flag = false;     // Boolean to save rain status

 void setup() {
    pinMode(enable2, OUTPUT);                                    // Setting led pin as output
    Serial.begin(115200);                                    // Initializing baud rate
    WiFi.begin("OnePlusNord", "prasanna");                    // Connect to WiFi using
given id and password
    while (WiFi.status() != WL_CONNECTED)                    // WiFi getting connected
    {
      delay(500);
      Serial.print("..");
    }
    Serial.println();
    Serial.println("NodeMCU is connected!");
    Serial.println(WiFi.localIP());                                    // Print IP connected
    dht.begin();                                    // Connect to dht sensor
    ThingSpeak.begin(client);                                    // Connect to ThingSpeak
 }

void loop() {
  sensorValue2 = analogRead(sensorPin);                    // Reading Raindrop sensor
value
  sensorValue2 = constrain(sensorValue2, 0, 1023);                    // Limiting the range of
values
  sensorValue2 = map(sensorValue2, 0 , 1023, 100, 0);                    // Invert the range
  if (sensorValue2 >= 20)                                    // If rain sensor value > threshold
  {
    Serial.println("rain is detected ");                    // Print on serial monitor that rain
detected
//    Serial.println(sensorValue2);
    digitalWrite(enable2, HIGH);                                    // Light the LED on board
    flag = true;
                                    // Rain status is true
```

```
 }
 else                                                    // If rain sensor value < threshold
 {
   Serial.println("rain not detected ");                           // Print on serial monitor that rain
detected
   digitalWrite(enable2, LOW);                             // Switch off the LED on board
   flag = false;                                           // Rain status is false
 }
 float h = dht.readHumidity();                             // Reading and saving Humidity value
 float t = dht.readTemperature();                          // Reading and saving Temperature
value
 Serial.println("Temperature: " + (String) t);                    // Printing these values on serial
monitor
 Serial.println("Humidity: " + (String) h);
 Serial.println();
 // SENDING DATA TO THINGSPEAK
 if (flag == true) {                                       // If it is raining
   ThingSpeak.writeField(myChannelNumber, 3, sensorValue2, myWriteAPIKey);      //
Send sensorValue to ThingSpeak in channel 3
   //delay(100);
   ThingSpeak.writeField(myChannelNumber, 2, h, myWriteAPIKey);       // Send humidity
value to ThingSpeak in channel 2
   //delay(100);
   ThingSpeak.writeField(myChannelNumber, 1, t, myWriteAPIKey);  // Send temperature
value to ThingSpeak in channel 1

 }
 else {                                                    // If it is not raining
    ThingSpeak.writeField(myChannelNumber, 3, 0, myWriteAPIKey);        // Send 0 to
ThingSpeak in channel 3
    //delay(100);
    ThingSpeak.writeField(myChannelNumber, 2, h, myWriteAPIKey);       // Send humidity
value to ThingSpeak in channel 2
    //delay(100);
    ThingSpeak.writeField(myChannelNumber, 1, t, myWriteAPIKey);       // Send
temperature value to ThingSpeak in channel 1
  }
 delay(1000);
}
```

Results and Discussion

With this project we were successful in making a small scale weather station that would be able to generate and visualize the local surrounding environmental conditions such as Temperature, Humidity and Rain. The data is stored on cloud and the visualization for the same is carried out on cloud as well.

Below are screenshots of the graph as visualized on thingspeak.

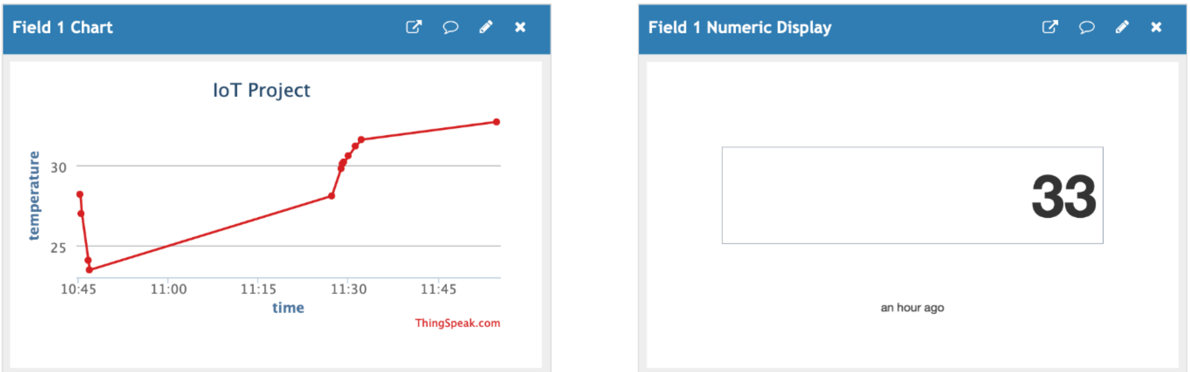

Fig. 1 Temperature graph and current Value

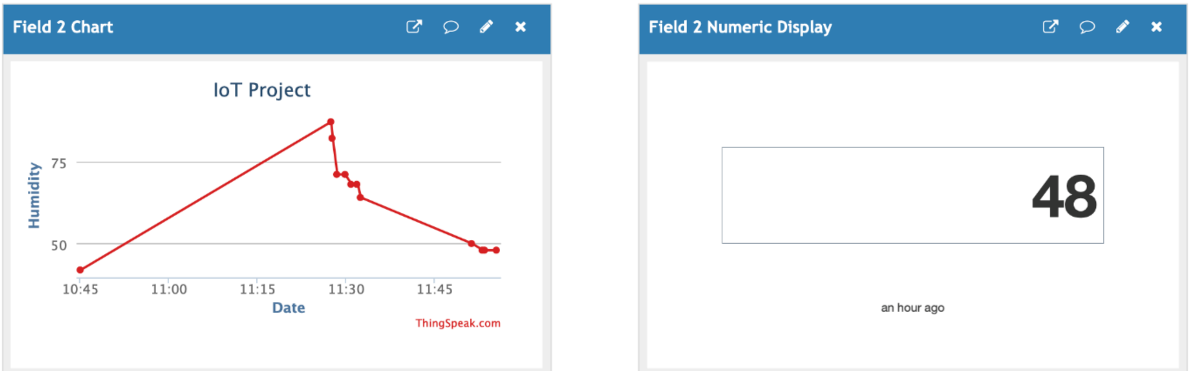

Fig. 2 Humidity graph and current Value

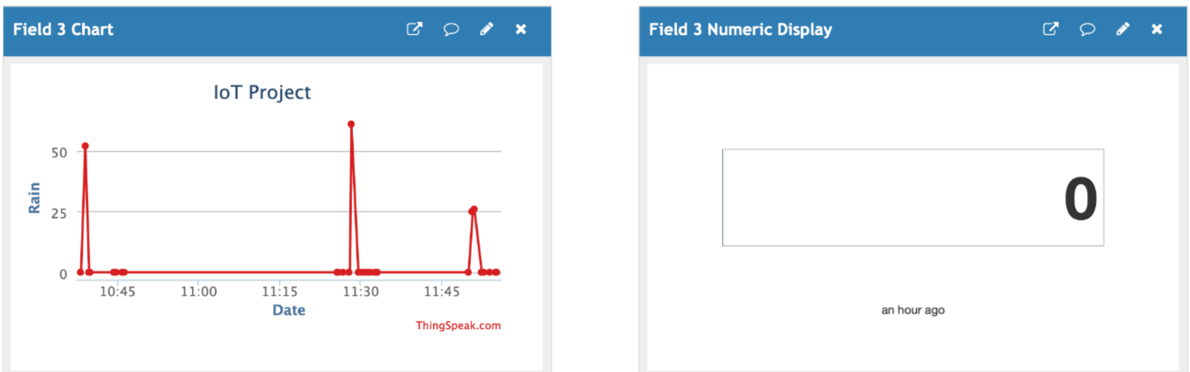

Fig. 3 Rain graph and current Value

Conclusion and Future work

This project aims to measure the various parameters like Temperature, Humidity, Rainfall and continuously monitor. The data can be stored online, which can be used to forecast weather and eventually analyze climate patterns, as well as for other meteorological purposes. The system uses a good combination of analog and digital sensors in wired and wireless modes of operation. Thus, a proof of concept for an Internet of Things device for a remote weather monitoring system has been established.

An alarm can be added to the circuit to notify the user in case of excess smoke conditions i.e. Smoke alarm. An SMS can be sent to clients notifying them with the temperature/humidity/smoke parameters.

References

[1] Karthik Krishnamurthi, Suraj Thapa, Lokesh Kothari, Arun Prakash," Arduino Based Weather Monitoring System", International Journal of Engineering Research and General Science Volume 3, Issue 2, March-April, 2015 ISSN 2091-2730

[2] Montgomery. K, Chiang. K, "A New Paradigm for Integrated Environmental Monitoring", ACM International Conference Proceeding Series, 2010.

[3] Satoh. F, Itakura. M, "Cloud-based Infrastructure for Managing and Analyzing Environmental Resources", SRII Global Conference, pp.325- 334, 201.

[4] K. Krishnamurthi, S. Thapa, L. Kothari, A. Prakash, "Arduino Based Weather Monitoring System" International Journal of Engineering Research and General Science Volume 3, Issue 2, March, April, 2015.

[5] D. V. Sose, A. D. Sayyad, "Weather Monitoring Station: A Review" Int. Journal of Engineering Research and Application, ISSN : 2248- 9622, Vol. 6, Issue 6, (Part -1) June 2016, pp.55-60.

[6] F. Hahn, M. Pablo, J. Reyes "Solar Driven Wind Speed Monitoring System Using Wireless or Wired Sensors" SciRes, Energy and Power Engineering, Vol 6, PP.213-221, 2014.

[7] Sharmila Borah," Temperature Monitoring of Server Room Using Matlab and Arduino",International Journal of Engineering Research & Technology (IJERT) IJERT ISSN: 2278-0181 IJERTV2IS90390 www.ijert.org Vol. 2 Issue 9, September – 2013

[8] Forat Falih Hasan, Sarmad Nozad Mahmood," Design of Weather Monitoring System Using Arduino Based Database Implementation", Journal of Multidisciplinary Engineering Science and Technology (JMEST) ISSN: 2458-9403 ,Vol. 4 Issue 4, April – 2017

[9] Kiranmai Nandagiri, Jhansi Rani Mettu," Implementation of Weather Monitoring System", International Journal of Pure and Applied Mathematics Volume 118 No. 16 2018, 477-493.

Annexure- Code/Implementation details if any

Github Link: https://github.com/shivam24-2000/Weather-Montoring-Station