**A REPORT**

**On**

**“Live Weather Monitoring System using Iot”**

**Submitted By**

## ( Dhamane Paresh, Dhole Aditya, Gangurde Aniket, Sayyad Jishan )

**Guided By**

**Dr. P. N. Kalavadekar**



**(Savitribai Phule Pune University, Pune)**

**In the academic year 2022-23 Department of Computer Engineering Sanjivani College of Engineering Kopargaon - 423 603.**

I

# CONTENTS

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr.** | | **Chapter** | **Page No** |
| **1.** |  | **Introduction** | 04 |
| **2.** |  | **Proposed System** | 05 |
| **3.** |  | **Methodology** | 06 |
| **4.** |  | **Modeling and Analysis** | 07 |
| **5.** |  | **Requirements** | 09 |
|  | 5.1 | Hardware Requirements | 09 |
|  | 5.2 | Software Requirements | 10 |
|  | 5.3 | Sample Code | 11 |
| **6** |  | **Results** | 13 |
|  | 6.1 | Test Cases | 14 |
| **7** |  | **Conclusion** | 15 |

**DIAGRAMS**

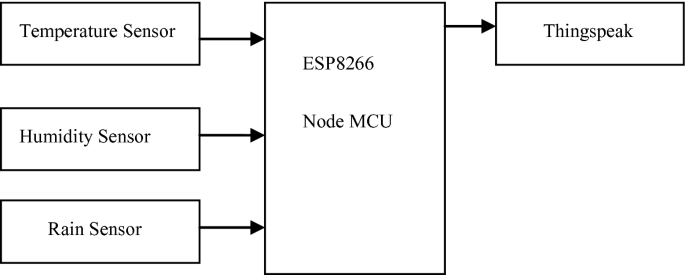
|  |  |  |
| --- | --- | --- |
| **Figure no** | **Diagram**  **Name** | **Page No** |
| a) | Proposed system for Weather Monitoring System | 05 |
| b) | 3D view of building. | 07 |
| c) | Algorithm | 08 |
| d) | Screenshots of Thingspeak | 05 |

# INTRODUCTION

Weather is a daily aspect but climate is the average of atmospheric conditions over a longer period of time. Weather changes with respect to the latitude and longitude of a place so even small changes may lead to large effects on the system. In the past few years, the increase in human activities and growing industries has had a drastic impact on the weather conditions. We need to be aware and ready for the upcoming disasters hence weather forecasting is important. In India, weather forecasting system are setup at a proximity of 32kms, due to their cost constraint. The data gathered by those stations is insufficient, eventually the accuracy is affected to a greater extend. In order to monitor the changes an effective system needs to be designed.

A weather station is used to measure atmospheric conditions at different locations at different periods of time for weather forecasts and to study different climatic conditions. Weather is mainly driven by temperature, humidity and atmospheric pressure. Other parameters like wind speed, wind direction and precipitation amount can also be measured. These parameters can be recorded periodically and a statistical analysis can be obtained which can determine the future conditions. India is an agriculture-based country, almost 17-18% of the country's GDP is contributed by agriculture which accounts to 50 percent of the workforce, but it is totally dependent on weather and rainfall, so weather monitoring needs to be accurate as well as zone specific in order to plan contingency for the upcoming situations.

# PROPOSED SYSTEM



***Figure: a) Proposed system for Weather Monitoring System***

When this system is powered on, the Nodemcu board connects to the algorithm development system called MATLAB through the Thingspeak cloud. Then, values are obtained from the sensors. Also, these values are sent to the thingspeak app using the internet. Then, we can see the values as a visualization on the screen.

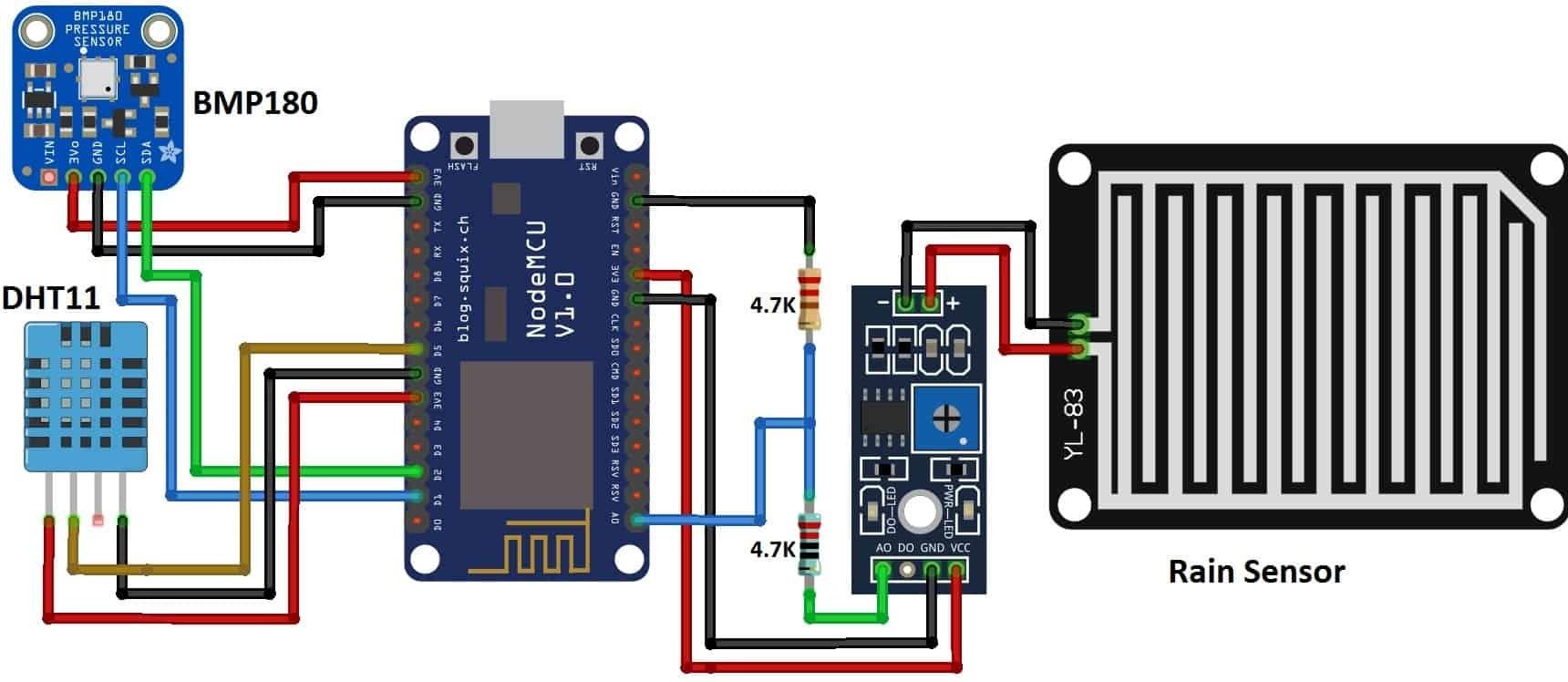
# METHODOLOGY

This project will focus on development of the Thing Speak an IoT platform that to show the data of the sensor. The method divided into two parts which are hardware and software development part. The hardware development involves the circuit construction and develops the prototype. Meanwhile, the software part involves the IoT coding, circuit schematic diagram, circuit simulation and data acquisition. By using three (3) types of sensor to monitoring the weather parameter that are temperature, humidity, rain, and atmospheric pressure.

The system will be able to display the weather condition by an analysis about the current weather with the sensor value data. All the data will be control by a microcontroller ESP8266 and NODEMCU as the client that will receive the sensor data from ESP8266. This system will be seen on ThingSpeak channel that has been created to simplify user to check online. The data collected will be analyse and compare it with any weather station to ensure the precise of data and weather condition on current condition. The Internet of Things (IoT) will connect the system with the user wireless and online without the need of checking manually

# MODELING AND ANALYSIS

The circuit diagram consists of the components that are utilized in this project. There are two modes available in this project working operation. Firstly, controlling mode will involve ESP8266 and monitoring mode will involve NODEMCU. This two-microcontroller board will communicate each other in order the monitoring mode get sensor data from controlling mode via wireless communication and hotspot Wi-Fi. Controlling mode will collect all the sensor data then send to the ThingSpeak website and monitoring mode to display on thingspeak display page. The client will display the sensor data on Thingspeak. The data collected will be analyzed to configure the actual condition and the current condition by using simple formula in Equation 1. The result of this data analysis then will be made the weather state for this system to tell the user about the rain and air quality condition is it good or bad in actual condition.



***Figure: b) 3D view of building.***

Sending DHT11 Sensor data

# Algorithm Used :

START

Sending DHT Sensor data



DHT 11

Sensor

Rain

Sensor

Gathered Sensor Data

Sending Rain Sensor data

Temperature, Humidity, Rain Sensor

END

Cloud Thingspeak

***Figure c): Algorithm***

# REQUIREMENTS

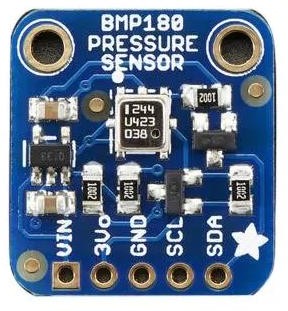
The requirements are divided into two parts which are hardware and software requirements. The hardware development involves the circuit construction and develops the prototype. Meanwhile, the software part involves the IoT coding, circuit schematic diagram, circuit simulation and data acquisition.

# Hardware Requirements:

## Node MCU ESP8266

The NodeMCU (***N***ode ***M***icro***c***ontroller ***U***nit) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) called the ESP8266.

## BMP 180

BMP180 is an **atmospheric pressure sensor**. The BMP180 sensor is mainly used to measure atmospheric pressure or biometric pressure. The working principle of the air pressure sensor is very simple, it works based on the weight of air.

Because the air around us has a certain weight, and this weight has a specific pressure.

## DHT 11

The **DHT11** is a commonly used **Temperature and humidity sensor that** comes with a dedicated NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data.

## Rain Sensor

The rain sensor **detects water that comes short circuiting the tape of the printed circuits**. The sensor acts as a variable resistance that will change status : the resistance increases when the sensor is wet and the resistance is lower when the sensor is dry.

# Software Requirements ThingSpeak

ThingSpeak is an open-source software

written in Ruby which allows users to communicate with internet enabled devices. It facilitates data access, retrieval and logging of data by providing an API to both the devices and social network websites.

## Sample Code:

#include <SFE\_BMP180.h> #include <Wire.h> #include <ESP8266WiFi.h> #include "DHT.h"

DHT dht(D3, DHT11); SFE\_BMP180 bmp;

double T, P; char status;

WiFiClient client;

String apiKey = ""; const char \*ssid = ""; const char \*pass = "";

const char\* server = "api.thingspeak.com";

void setup() { Serial.begin(115200); delay(10); bmp.begin();

Wire.begin();

dht.begin(); WiFi.begin(ssid, pass);

while (WiFi.status() != WL\_CONNECTED) { delay(500);

Serial.print(".");

}

Serial.println(""); Serial.println("WiFi connected");

}

void loop() {

//BMP180 sensor

status = bmp.startTemperature(); if (status != 0) {

delay(status);

status = bmp.getTemperature(T);

status = bmp.startPressure(3);// 0 to 3 if (status != 0) {

delay(status);

status = bmp.getPressure(P, T); if (status != 0) {

}

}

}

//DHT11 sensor

float h = dht.readHumidity(); float t = dht.readTemperature();

if (isnan(h) || isnan(t)) {

Serial.println("Failed to read from DHT sensor!"); return;

}

//Rain sensor

int r = analogRead(A0);

r = map(r, 0, 1024, 0, 100);

if (client.connect(server, 80)) { String postStr = apiKey; postStr += "&field1=";

postStr += String(t); postStr += "&field2="; postStr += String(h); postStr += "&field3="; postStr += String(P, 2); postStr += "&field4="; postStr += String(r);

postStr += "\r\n\r\n\r\n\r\n";

client.print("POST /update HTTP/1.1\n"); client.print("Host: api.thingspeak.com\n"); client.print("Connection: close\n");

client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n"); client.print("Content-Type: application/x-www-form-urlencoded\n"); client.print("Content-Length: ");

client.print(postStr.length()); client.print("\n\n\n\n"); client.print(postStr);

Serial.print("Temperature: "); Serial.println(t); Serial.print("Humidity: "); Serial.println(h); Serial.print("absolute pressure: "); Serial.print(P, 2); Serial.println("mb"); Serial.print("Rain"); Serial.println(r);

}

client.stop(); delay(1000);

}

## RESULT

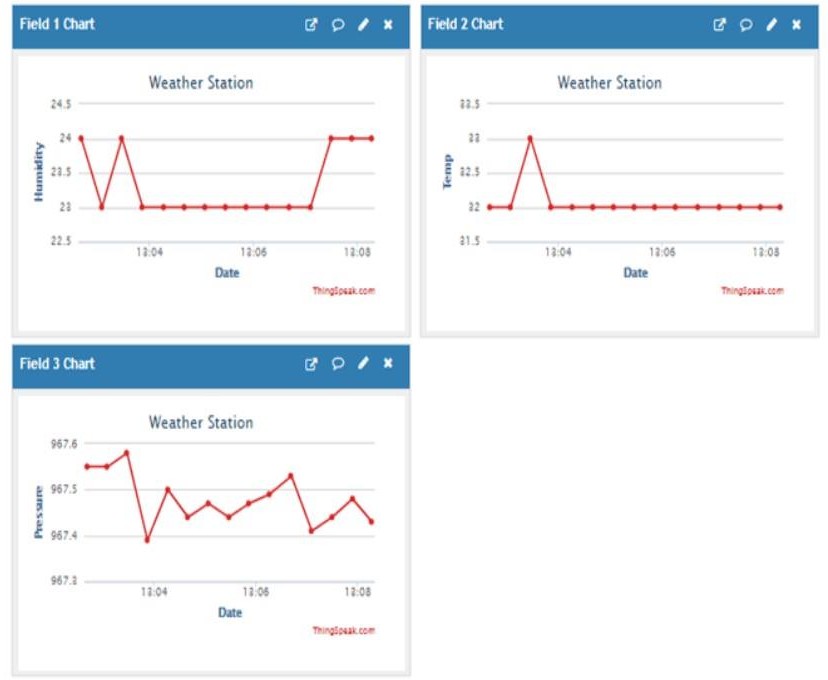
First the circuit of control unit system have been made that ESP8266 microcontroller control all weather parameters sensor, that are DHT11(Temperature, Humidity) sensor, BMP180 (Atmospheric pressure), Rain sensor. Then it powered by USB cable also to upload the sketch of coding in ESP8266 microcontroller. The sensor data can be display on serial monitor in Arduino IDE software. ESP8266 will connect with the Wi-Fi hotspot that have applied to this system so that the web server can be create to display all the sensor data. Data that received by weather station will be displaying on thingspeak demonstrate the communication of both sensor station and weather station by using Wi-Fi hotspot. The communication is successfully established. The web server contains html that can display the sensor data by simple coding and connection where the IP address of the ESP8266 are needed to complete this action are shown in Figure . After that it will read all the sensor value and then send to the cloud data where ThingSpeak has been uses for this. ThingSpeak will stored the sensor value and display that data to the channel create there. The user can check the weather parameter via ThingSpeak websites. The data has collected from the reading of ESP8266 for all sensor and send that data to the ThingSpeak as the results of this project objective.

***TABLE 1: DHT11***

|  |  |  |
| --- | --- | --- |
| **PARAMETER** | **TIME** | **VALUE** |
| TEMPERATURE | 10AM | 24~28 C |
| TEMPERATURE | 1PM | 29~31C |
| TEMPERATURE | 10PM | 27~29C |
| HUMIDITY | 10AM | 78 |
| HUMIDITY | 10PM | 77 |

***TABLE 2: BMP180***

|  |  |  |
| --- | --- | --- |
| **PARAMETER** | **TIME** | **VALUE** |
| PRESSURE | 10AM | 1001~1800 |
| PRESSURE | 1PM | 1006~1829 |
| PRESSURE | 10PM | 997~1560 |



***Figure d): Screenshot of the Thingspeak***

# CONCLUSION

As the conclusion this project have cleared the objective that to build a system that can monitored weather parameter by wireless system and IoT. The Sensor station and Weather station will be communicated by hotspot Wi-Fi and it is limited in areas covered but still better in communication via wireless. The value that been recorded from google sheet and Table 1, 2 it seen that the weather at particular place has different condition from the exact condition with the accuracy of weather reporting system and forecast system data has been compared. It says that weather reporting system is more accurate than forecast system. This weather reporting system will display the sensor data to ThingSpeak and IFTTT to save the data into google sheet.

# REFERENCES

1. M. H. Asghar, A. Negi, and N. Mohammadzadeh, “Principle application and vision in internet of things (iot),” in International Conference on Computing, Communication Automation, May 2015, pp. 427–431.
2. A. Gheith, R. Rajamony, P. Bohrer, K. Agarwal, M. Kistler, B. L. W. Eagle, C. A. Hambridge, J. B. Carter, and T. kaplinger, “Ibm bluemix mobile cloud services,” IbM Journal of research and Development, vol. 60, no. 2-3, pp. 7:1–7:12, March 2016.
3. S. Gangopadhyay and M. k. Mondal, “A wireless framework for environmental monitoring and instant response alert,” in 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Jan 2016, pp. 1–6.
4. H. Saini, A. Thakur, S. Ahuja, N. Sabharwal, and N. kumar, “Arduino based automatic wireless

weather station with remote graphical application and alerts,” in 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), Feb 2016, pp. 605–609.

**THANK YOU…!**