Smart Weather Monitoring System

Submitted in partial fulfillment of the requirements of the degree

BACHELOR OF ENGINEERING

IN INFORMATION TECHNOLOGY

(Subject – Internet of Everything Lab)

By

Komal Rane	**18101B0002**
Nitesh Pednekar	**18101B0026**

Supervisor

Ms. Rasika Ransing

Vidyalankar Institute of Technology

Accredited A+ by NAAC

Department of Information Technology

Vidyalankar Institute of Technology

Vidyalankar Educational Campus,

Wadala(E), Mumbai - 400 037

University of Mumbai (AY 2021-22)

CERTIFICATE

This is to certify that the Mini Project entitled " **Smart Weather Monitoring System** " is a bonafide work of **Komal Rane - 18101B0002** and **Nitesh Pednekar - 18101B0026** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **"Bachelor of Engineering"** in **"Information Technology"** .

Ms. Rasika Ransing

Dr. Vipul Dalal	**Dr. S. A. Patekar**
Head of Department	Principal

Mini Project Approval

This Mini Project entitled **”Smart Weather Monitoring System”** by **Komal Rane - 18101B0002** and **Nitesh Pednekar - 18101B0026** is approved for the degree of **Bachelor of Engineering** in **Information Technology.**

Examiners

1……………………………………….
(Internal Examiner Name & Sign)

2……………………………………….
(External Examiner name & Sign)

Date:

Place:

Contents

Abstract

With the advent of high speed Internet, more and more humans around the globe are interconnected. Internet of Things (IoT) takes this a step further, and connects not only humans but electronic devices which can speak amongst themselves. With falling costs of Wifi enabled devices this trend will only gather more momentum. The main concept behind the Internet of Things(IoT) is to connect various electronic devices through a network and then retrieve the data from these devices (sensors) which can be distributed in any fashion, upload them to any cloud service where one can analyze and process the gathered information. In the cloud service one can utilize this data to alert people by various means such as using a buzzer or sending them an email or sending them an SMS etc.

A weather station can be described as an instrument or device, which provides us with the information of the weather in our neighboring environment. For example it can provide us with details about the surrounding temperature, barometric pressure, humidity, etc. Hence, this device basically senses the temperature, pressure, humidity.There are various types of sensors present in the prototype, using which all the aforementioned parameters can be measured. It can be used to monitor the temperature or humidity of a particular room/place. With the help of temperature and humidity we can calculate other data parameters, such as the dew point. The brain of the prototype is the ESP8266 based Wi-fi module.Various sensors are connected to the Arduino namely temperature and humidity sensor(DHT11), pressure sensor(BMP180).The values are then shown to the user

Acknowledgement

We would like to express our deepest appreciation to all those who provided us the possibility to complete this report. We express our profound gratitude to our **Prof. Rasika Ransing** Ma'am, our respectable project guide, for her gigantic support and guidance. Without her counseling our project would not have seen the light of the day.

We extend our sincere thanks to **Dr. Vipul Dalal**, Head of the Department of Information Technology for offering valuable advice at every stage of this undertaking. We would like to thank all the staff members who willingly helped us. We are grateful to VIDYALANKAR INSTITUTE OF TECHNOLOGY for giving us this opportunity.

The days we have spent in the institute will always be remembered and also be reckoned as guiding in our career.

1. Komal Rane
2. Nitesh Pednekar

List Of Figures

1.Introduction

1.1 Introduction

Weather monitoring system deals with detecting and gathering various weather parameters which can be analyzed or used for weather forecasting.The aim of this system is achieved by technologies such as Internet of Things(IOT) and Cloud. The idea of internet of things is to connect a device to the internet and to other required connected devices.Using Internet the information from the IOT device can easily be transferred to the cloud and then from the cloud to the end user.Weather Monitoring is an essential practical implementation of the concept of Internet of Things,it involves sensing and recording various weather parameters and using them for alerts,sending notifications, adjusting appliances accordingly and also for long term analysis.Also we will try to identify and display trends in parameters using graphical representation.

The devices used for this purpose are used to collect, organize and display information.It is expected that the internet of things is going to transform the world by monitoring and controlling the phenomenon of environment by using sensors/devices which are able to capture,process and transmit weather parameters.Cloud is availability of computer system resources like data storage, computing power without direct active management of user.The data captured is transmitted to the cloud so that the data could be further displayed.Besides this, the system consists of components such as Arduino UNO board which is a microcontroller board consisting of 14 digital pins, a USB connection and everything used to support microcontroller; DHT11 is Temperature and humidity sensor which is used for detecting these mentioned parameters; WIFI module is used to convert the data collected from the sensors and then send it to the web server. So,in this way weather conditions can be monitored from any remote location in the world.

1.2 Motivation

IOT has become a great area of interests for institutes,big tech companies and obviously users or customers also.Many IOT based concepts have gained so much attention like Smart wearable devices,smart home,smart city etc.Almost all the applications based on Internet of things include devices like transducers and sensors attached to the microcontroller with a wireless/wired flow of data to a remote cloud service or a local data storage which converts the raw data to a significant information which can further used in many areas.IOt can help solve real life issues and our smart weather monitoring system will provide a system way to analyze the weather conditions.

1.3 Problem Statement

The main aim of the project is to develop a smart weather monitoring system. The system will then predict the weather conditions accordingly.The parameters which would be monitored by the system will include temperature,pressure and humidity of the surroundings.

Objectives

- Detect Temperature ,Pressure and Humidity
- Develop a smart weather station
- Develop a model to predict the weather conditions

1.4 Organization Of Report

The report is divided into 3 parts. The very first part is about the introduction.It covers all the information about the introduction to the project,the aim behind developing the idea,the motivation of working on the idea and explanation of the problem statement. The second part is based on the literature survey. It explains details about the existing systems with the similar idea, few shortcomings in the research gap and the contribution of each member towards the development of the mini project.The last part focuses on the actual proposed system.It tells about the proposed solution ,explains the architecture and working of the model ,details of the hardware and software used,the outputs of the project and the final conclusions and future scope.

2.Literature Survey

2.1 Survey of Existing/Similar System

IOT has become a great area of interests for institutes,big tech companies and obviously users or customers also.Many IOT based concepts have gained so much attention like Smart wearable devices,smart home,smart city etc.Almost all the applications based on Internet of things include devices like transducers and sensors attached to the microcontroller with a wireless/wired flow of data to a remote cloud service or a local data storage which converts the raw data to a significant information which can further used in many areas.While working on this project we came across some works that International Journal of Advanced Science and Technology Vol. 29, No. 12s, (2020), pp. 2473-2479 ISSN: 2005-4238 IJAST Copyright ⓒ 2020 SERSC 2474 have been accomplished in making smart applications using either Raspberry Pi boards or arduino board which are economical.Most of the applications were built using these boards for example smart city and other automation projects. In some journals it was said that for a smart city “Places can be equipped with sensors and monitor environmental conditions, cyclists or athletes can find the most"healthy" trips and the city can respond by adjusting the traffic orby planting more trees in some areas.

The data will be accessible to all citizens to promote the creation of applications using real-time information for residents.”So we can say that this weather monitoring system will be helpful in some smart city projects also. In one of the technical papers online the authors chose a single sensor ie. composite DHT11 sensor for reading both temperature and humidity. Earlier people staying at home and busy in their household works or people who work in offices had no idea about the environmental parameters outside their home or office.

People have no idea if the temperature outside is quite low or high or normal or if it is raining or not or the value of the humidity in the environment. According to some reports, the monitoring systems can provide self-protection to our environment such as protecting public health from the pollution or at least reducing the effects of pollution on the public. It will notify us whenever the temperature is lower than it should be or is higher than normal. It will also automatically notify whether it is raining so one can carry an umbrella or a raincoat. There is a system where the weather monitoring system was designed using a particle photon which is an Arduino Compatible Iot board. It will also give us morning,evening and night wish messages as it has a Light Sensor attached. The authors in one of the papers have mentioned a great thing that “By deploying sensor devices in the environment, we can bring the environment into real life.”

2.2 Limitation Existing/Similar system or research gap

- The Bulky Machinery Of Conventional Weather Monitoring Systems Requires Constant Monitoring And Should Be Changed Routinely.
- This Bulky Machinery Also Increases The Cost Of Installation.
- The existing system Also Has A High Power Consumption due to a lot of sensors Which Increases Its Cost.Data Is Transferred Manually.
- The Existing Weather System Predicts The Weather And The Sudden Change In The Forecast With Some Delay.
- The systems are also complex to understand by the naive users

2.3 Mini Project Contribution

The project work was equally divided in group members.The hardware connections of the model was performed by all the project members. The implementation of the hardware and connections to system, connecting the code to the hardware and report formation all was done with contributed efforts from all team members

3.Proposed System

3.1 Introduction

Global warming has led to unpredictable climates; researchers around the world are using weather stations to observe, record and analyze weather patterns to study climate changes and provide weather forecasts. These Weather stations normally consist of a few sensors to measure environmental parameters and a monitoring or logging system to analyze these parameters. In this tutorial, we will learn how to build a wireless IoT-based weather station that can measure critical environmental parameters like Temperature, Humidity, and Pressure. Also since our weather station is IoT enabled, we can send these parameters to a ThingSpeak channel (IoT cloud) where we can store, analyze, and access the data remotely.

We will be using the Arduino board along with the DHT11 sensor, BMP180 sensor, and an ESP8266 wifi module. The DHT11 sensor senses the temperature and humidity, while the BMP180 sensor calculates the pressure, and ESP8266 is used for internet connectivity. In our previous project, we already learned to use the DHT11 sensor to monitor temperature and humidity with Arduino, here in this project, we are adding another sensor (BMP180) to make a complete weather station using Arduino. Sending these data to ThingSpeak enables live monitoring from anywhere in the world and we can also view the logged data which will be stored on their website and even graph it over time to analyze it.

3.2 Architecture /Circuit Diagram

The circuit diagram of the project is s follows

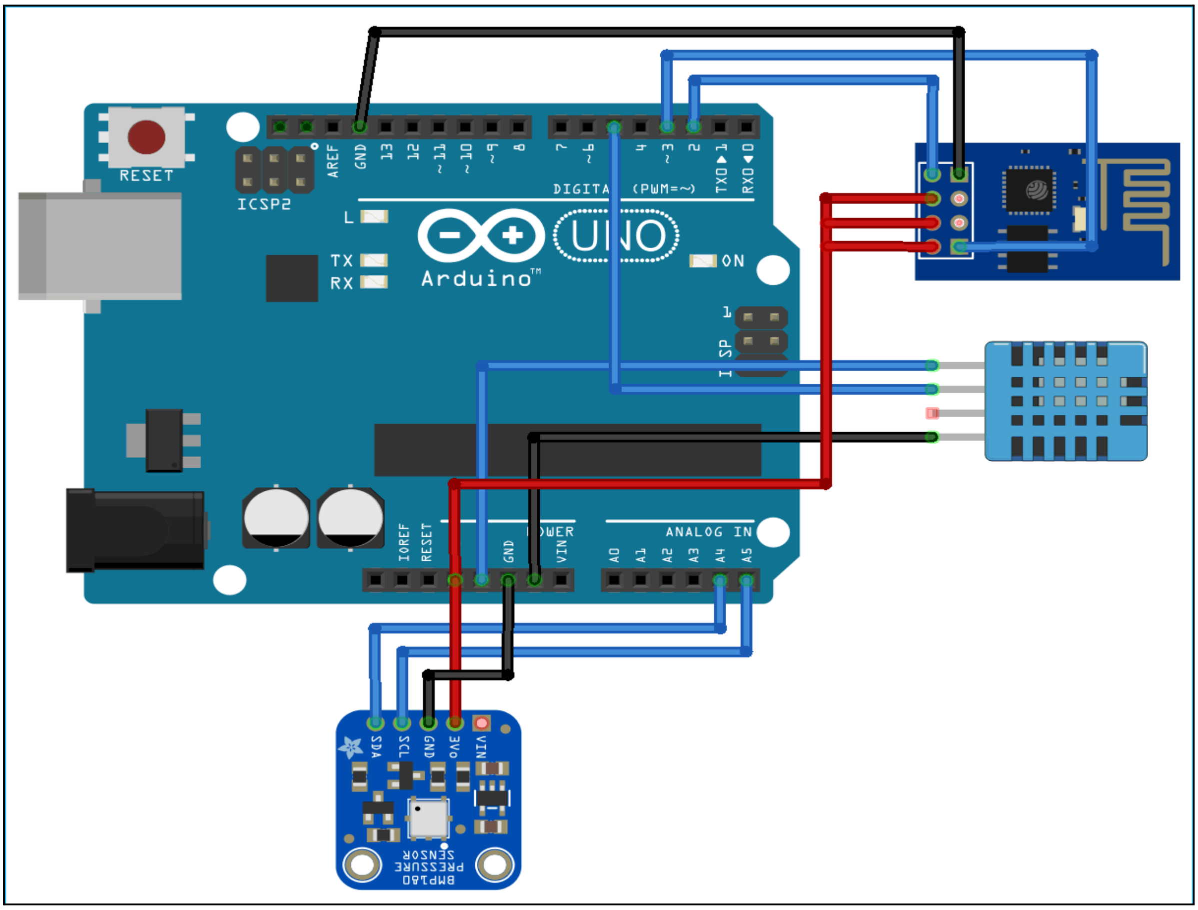


Fig 3.1 Circuit Diagram

The DHT11 sensor is powered by the 5V pin of the Arduino and its data pin is connected to pin 5 for one-wire communication. The BMP180 sensor is powered by the 3.3V pin of Arduino and its data pins SCL (Serial Clock) and SDA (Serial Data) are connected to the A4 and A5 pins of Arduino for I2C communication.The ESP8266 module is also powered by the 3.3V pin of the Arduino and its Tx and Rx pins are connected to Digital pins 2 and 3 of Arduino for serial communication. You can use the below table as a reference for making your connections.

S.NO.	Pin Name	Arduino Pin
1	ESP8266 VCC	3.3V
2	ESP8266 RST	3.3V
3	ESP8266 CH-PD	3.3V
4	ESP8266 RX	TX
5	ESP8266 TX	RX
6	ESP8266 GND	GND
7	BMP180 VCC	5V
8	BMP180 GND	GND
9	BMP180 SDA	A4
10	BMP180 SCL	A5
11	DHT-11 VCC	5V
12	DHT-11 Data	5
13	DHT-11 GND	GND

Fig 3.2 Connection description table

3.3 Algorithm And Process Design

Arduino Code

```
#include <WiFi.h>
#include <DHT.h>
#include <Wire.h>
#include <SoftwareSerial.h>
#include <stdlib.h>
#include <SFE_BMP180.h>
SFE_BMP180 pressure;
#define DHTPIN 5
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
#define TEMPTYPE 0
#define ALTITUDE 160 // Altitude from Bussero (MI) Italy
#define ssid  "Enter Your WiFi Name Here "     // "WiFi Name"
#define pass "WiFi Password"       // "Password"
#define server = "api.thingspeak.com";
String apiKey ="Enter the API Key";
char buffer[10];
char t_buffer[10];
char h_buffer[10];
char P_buffer[10];

SoftwareSerial ser(2, 3); // RX, TX
void setup() {
  Wire.begin();
  pressure.begin();
  // enable debug serial

  Serial.begin(9600);
  Serial.println("AT");
  delay(5000);
  if(Serial.find("OK")){
    connectWiFi();
}
void loop()
{
```

```
  delay(60000); // 60 seconds

void Trsmission()
{
  int8_t h = dht.readHumidity();
  int16_t t = dht.readTemperature(TEMPTYPE);
  char status;
  double T,P,p0,a;
  status = pressure.startTemperature();
  if (status != 0)
  {
    delay(status);
    status = pressure.getTemperature(T);
    if (status != 0)
    {
      status = pressure.startPressure(3);
      if (status != 0)
      {
        // Wait for the measurement to complete:
        delay(status);
        status = pressure.getPressure(P,T);
        if (status != 0)
        {
          p0 = pressure.sealevel(P,ALTITUDE); // we're at 1655 meters (Boulder, CO)
          a = pressure.altitude(P,p0);
        }
        else Serial.println("error retrieving pressure measurement\n");
      }
      else Serial.println("error starting pressure measurement\n");
    }
    else Serial.println("error retrieving temperature measurement\n");
  }
  float temp = t;
  float humidity = h;
  float Pression = p0;
  String strTemp = dtostrf(temp, 4, 1, t_buffer);
  String strHumid = dtostrf(humidity, 4, 1, h_buffer);
```

```
  String strPres = dtostrf(Pression, 4, 2, P_buffer);
  Serial.print("Temperature: ");
 Serial.println(strTemp);
Serial.print("Humidity: ");
  Serial.println(strHumid);
  Serial.print("Pression: ");
  Serial.println(strPres);
  String cmd = "AT+CIPSTART=\"TCP\",\"";
  cmd += "184.106.153.149"; // api.thingspeak.com
  cmd += "\",80";
  ser.println(cmd);
  if(ser.find("Error")){
    Serial.println("AT+CIPSTART error");
    return;
  }
  if(ser.find("Error")){
    Serial.println("AT+CIPSTART error");
    return;
  }
  // prepare GET string
  String getStr = "GET /update?api_key=";
  getStr += apiKey;
  getStr +="&field1=";
  getStr += String(strTemp);
  getStr +="&field2=";
  getStr += String(strHumid);
  getStr +="&field3=";
  getStr += String(strPres);
  getStr += "\r\n\r\n";
  // send data length
   cmd = "AT+CIPSEND=";
  cmd += String(getStr.length());
  ser.println(cmd);
  //ser.print(getStr);
if(ser.find(">")){
    ser.print(getStr);
  }
  else{
```

```
  ser.println("AT+CIPCLOSE");
  // alert user
  Serial.println("AT+CIPCLOSE");
  ser.println("AT+RST");
 }

 char buffer[10] = "";
}
```

Data Visualization and Prediction Code

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
data = pd.read_csv('C:\\Users\\ranek\\Downloads\\weatherHistory.csv')
data.head()
data.isnull().sum()
col = ['Formatted Date', 'Apparent Temperature (C)', 'Humidity', 'Daily Summary']
c = data
c = c[col]
c.head()
c['Formatted Date'] = pd.to_datetime(c['Formatted Date'], utc=True)
c = c.set_index('Formatted Date')
c= c.resample('M').mean()
plt.figure(figsize=(18,5))
plt.title('Variation of temp with humidity')
plt.plot(c)
plt.figure(figsize=(15, 5))
data_of_april = c[c.index.month==4]
plt.plot(data_of_april, marker='o',label=['Apparent Temperature (C)','Humidity'] );
plt.legend(loc = 'center right',fontsize = 10)
plt.title('Relation between temperature and humidity for the month of April')
plt.show()
correlation = c.corr()
sns.heatmap(correlation)
plt.figure(figsize = (18,5))
sns.barplot(x='Apparent Temperature (C)', y='Humidity', data=data_of_april)
plt.xticks(rotation=-30)
plt.title('Relation between temperature and humidity for the month of April')
plt.show()
data['Precip Type'].fillna(method='ffill',inplace=True,axis=0)
```

```
# Splitting the dataset into train data and test data
# Train dataset is 70% of and Test dataset is 30% of original dataset
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3,random_state=1)
from sklearn.ensemble import RandomForestClassifier
RF = RandomForestClassifier(max_depth=32,n_estimators=120,random_state=1)
RF.fit(x_train,y_train)
y_pred = RF.predict(x_test)
# Finding accuracy of model using test data
from sklearn.metrics import accuracy_score
accuracy_score(y_test, y_pred)
from sklearn.linear_model import LogisticRegression
logisticRegr = LogisticRegression()
logisticRegr.fit(x_train,y_train)
predictions = logisticRegr.predict(x_test)
```

Process Design

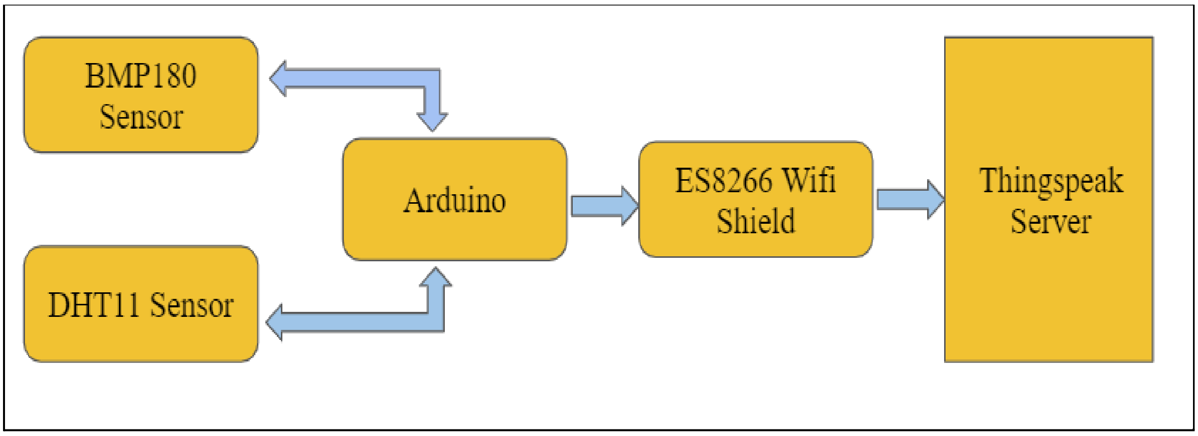


Fig 3.3 Workflow Diagram

The various sensors used will be connected to the arduino uno as per the specific pin connections suitable.Then the arduino will be connected to the wifi shield which will help in sending the data to the cloud platform.The data will be sent to the thingspeak server which will be than displayed on the dashboard.Sending these data to ThingSpeak enables live monitoring from anywhere in the world and we can also view the logged data which will be stored on their website and even graph it over time to analyze it.

3.4 Details Of Hardware and Software

Hardware

1. **Arduino Uno :**

Arduino/Genuino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

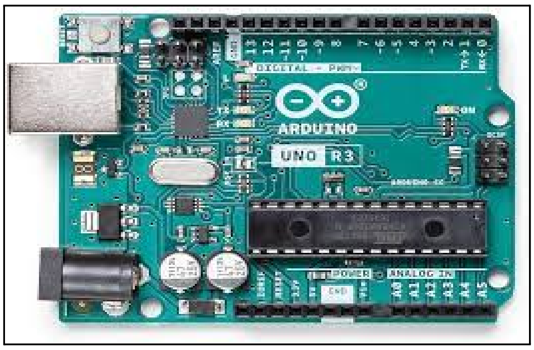


Fig 3.4 Arduino Uno

2. **ESP2866 Wifi Shield**

The WiFI shield connects to an Arduino board using long wire-wrap headers which extend through the shield. This keeps the pin layout intact and allows another shield to be stacked on top. The WiFi Shield can connect to wireless networks which operate according to the 802.11b and 802.11g specifications.

Fig 3.5 Wifi Shield

3.DHT11 Sensor

The DHT11 is a basic, ultra low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). It's fairly simple to use, but requires careful timing to grab data.

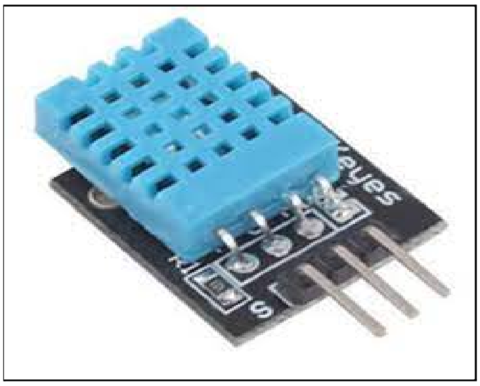


Fig 3.6 DHT11 Sensor

4.BMP180 Sensor

The BMP180 measures both pressure and temperature, because temperature changes the density of gasses like air. At higher temperatures, air is not as dense and heavy, so it applies less pressure on the sensor. At lower temperatures, air is more dense and weighs more, so it exerts more pressure on the sensor.

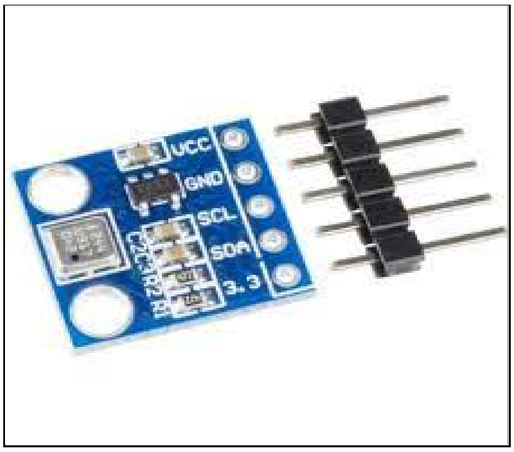


Fig 3.7 BMP180 Sensor

5.BreadBoard

A breadboard is used to build and test circuits quickly before finalizing any circuit design. The breadboard has many holes into which circuit components like ICs and resistors can be inserted.The inside of a breadboard has rows of metal clips that can fit into the leads. When the clips are removed from a breadboard, they look like this. When you press a component's lead into a hole, one of the clips grabs onto it.

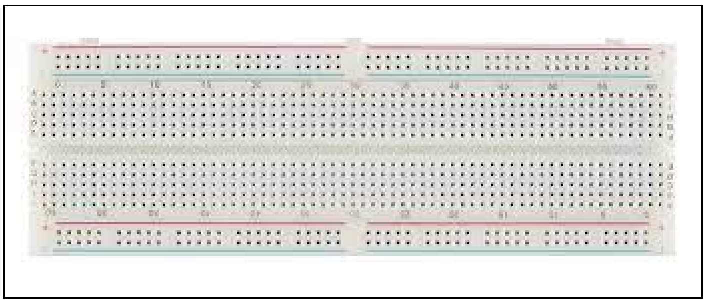


Fig 3.8 BreadBoard

6. Connecting Wires

Used to make connections to components and breadboard

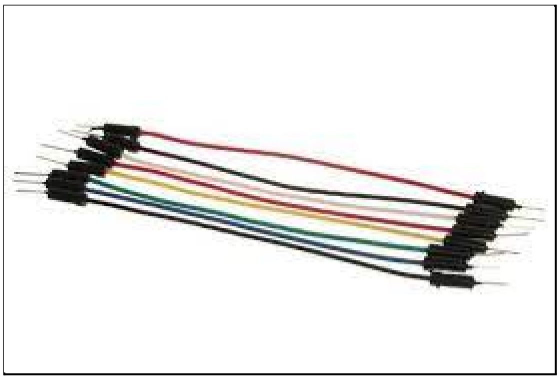


Fig 3.9 Connecting Wires

Software

1. **Arduino IDE**

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++.

Fig 3.10 Arduino IDE Logo

2. **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. ThingSpeak was originally launched by ioBridge in 2010 as a service in support of IoT applications.ThingSpeak has integrated support from the numerical computing software MATLAB from MathWorks, allowing ThingSpeak users to analyze and visualize uploaded data using MATLAB without requiring the purchase of a MATLAB license from MathWorks.

Fig 3.11 Thingspeak logo

3.5 Experiments and Results

Data Prediction and Data visualization is performed. The algorithms used for prediction are Random Forest and Logistic Regression .The accuracy of random forest is 86% and that of logistic regression is 82%

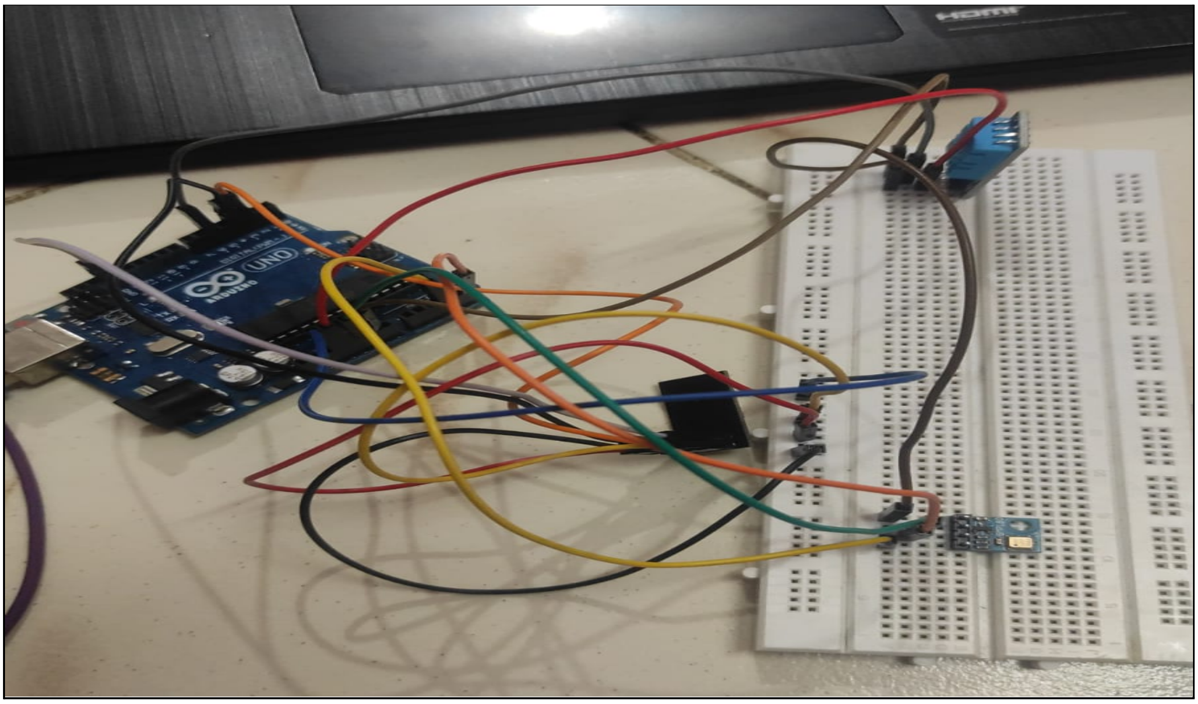


Fig 3.12 : Actual Hardware Model

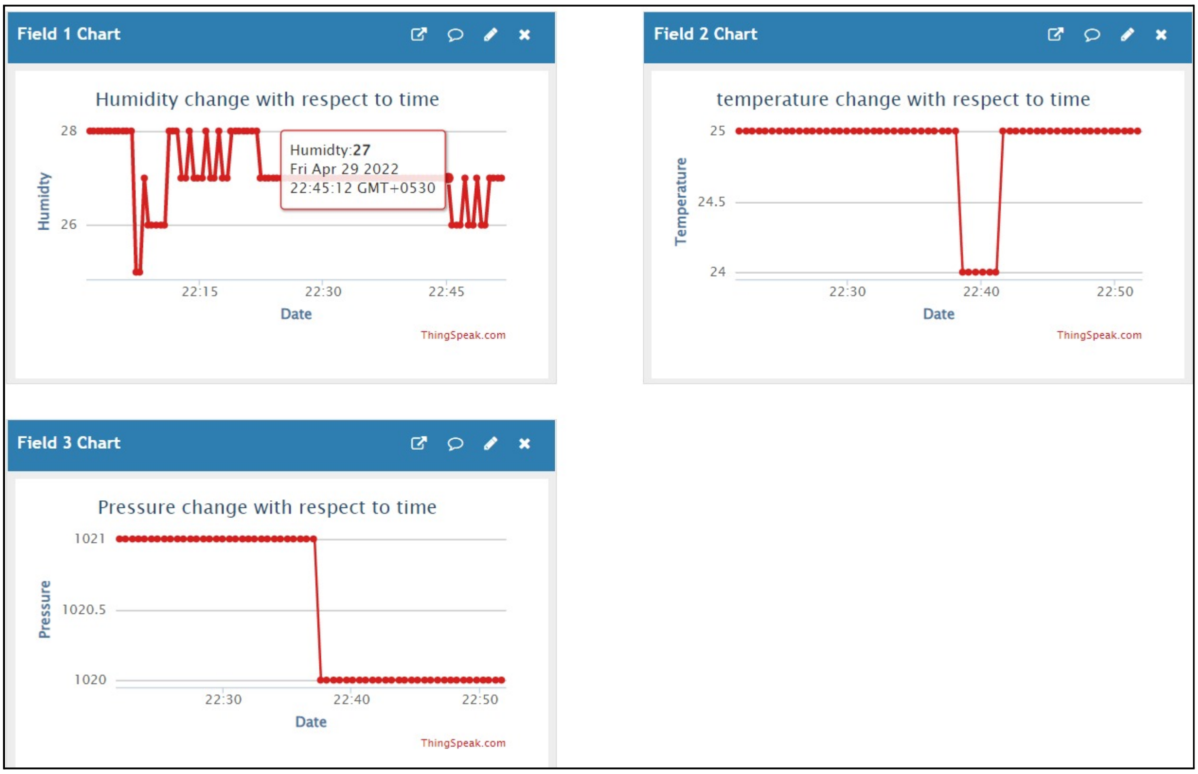
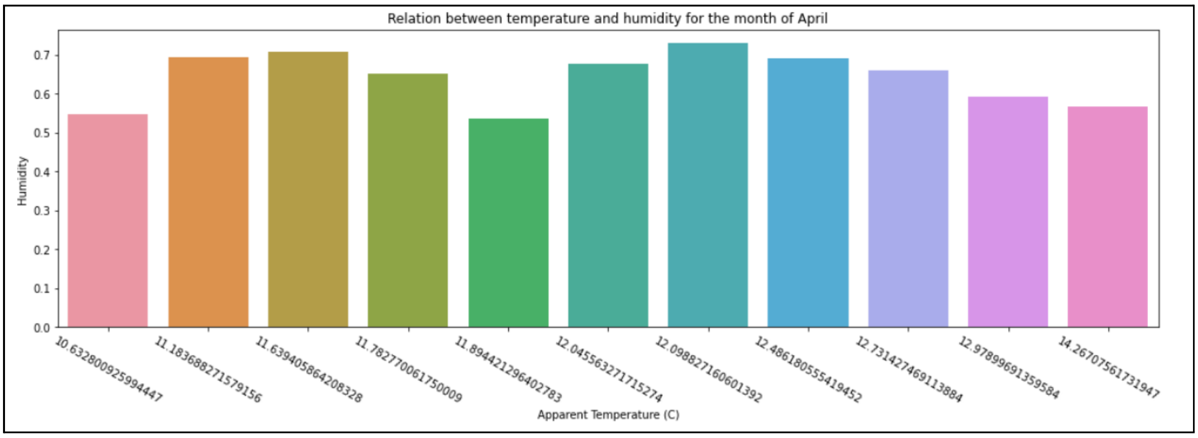


Fig 3.13 Dashboard of live data

Data Visualization



3.14 Temperature ranges in April

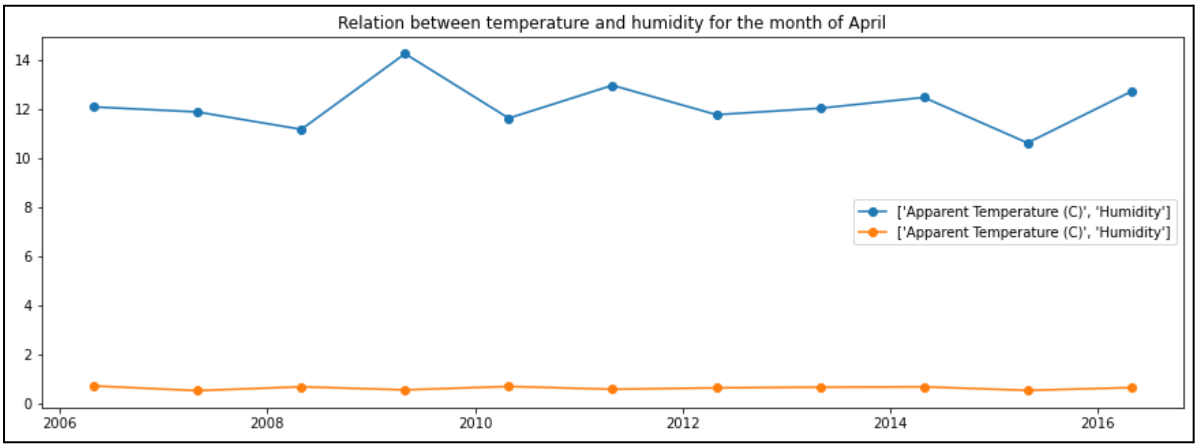


Fig 3.15 Apparent temperature and humidity

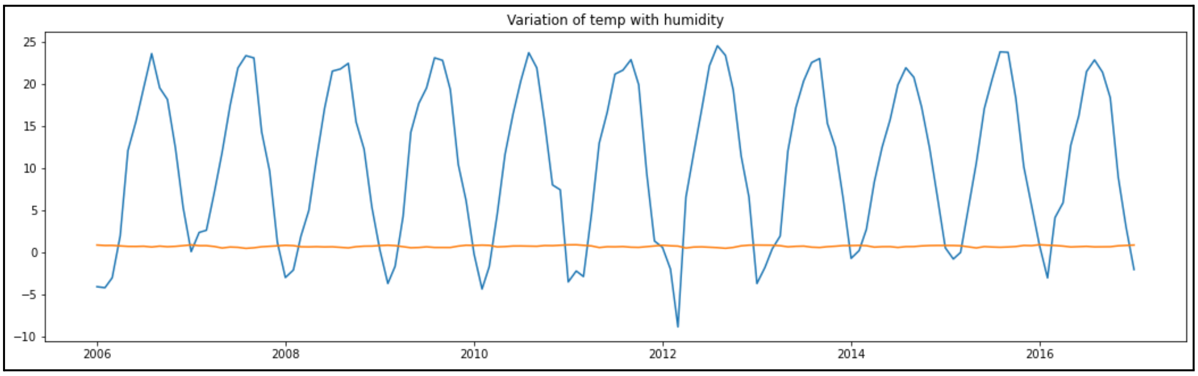


Fig 3.16 Variation of temp with humidity

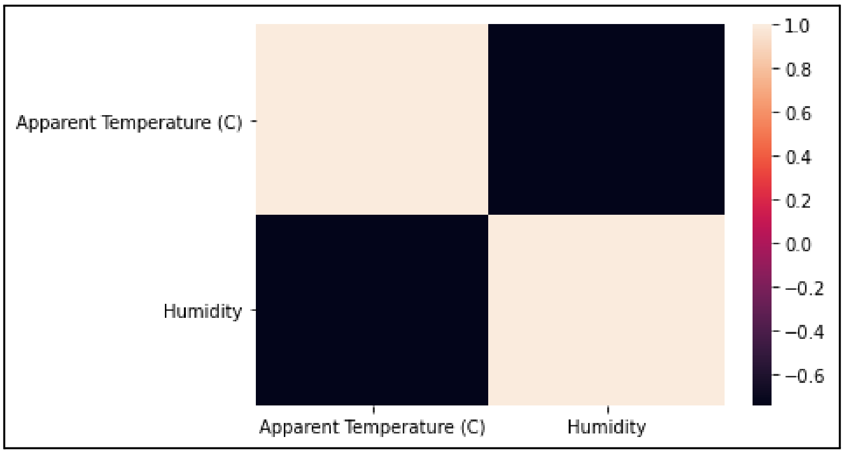


Fig 3.17 Correlation Matrix

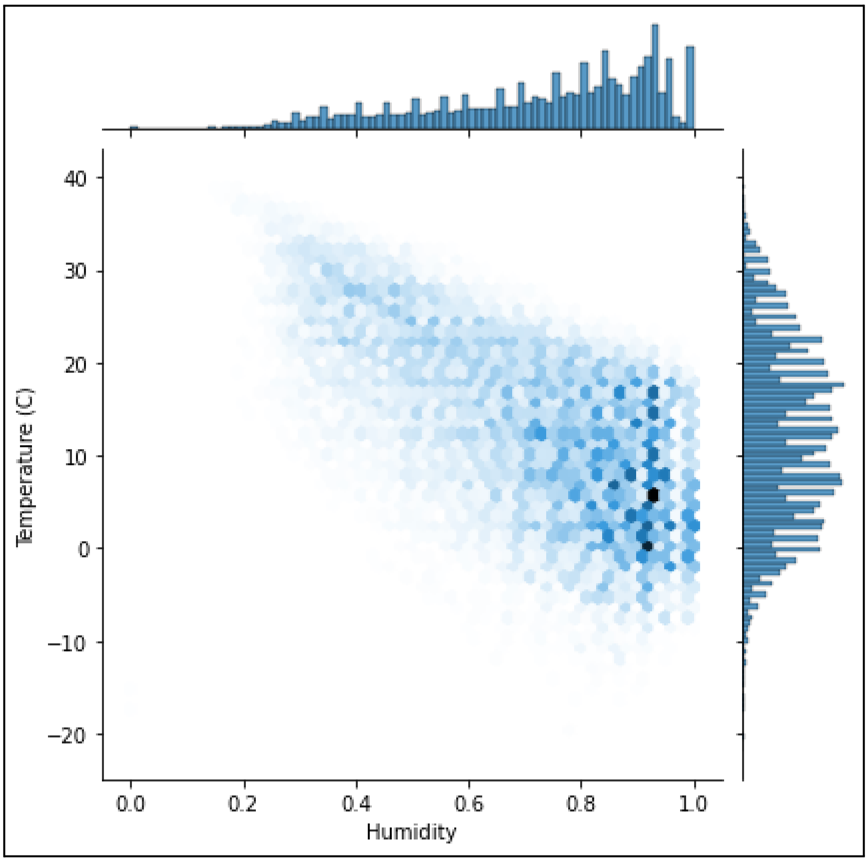


Fig 3.18 Humidity Vs Temperature

3.6 Conclusion

In this project, a smart weather monitoring system and a model is developed that allows users to see the temperature ,pressure and humidity. Also the parameters can be further used to analyze the weather conditions and predict the rains. The model is trained and the weather is predicted using machine learning.Also the analysis of the live data is seen on the dashboard of thingspeak.The main aim of the model is to provide the values which can be further used for prediction,classification or many other things and can help solve issues like global warming ,unnatural rains and so on.

3.6 Future Scope

The proposed IoT based weather station can be modified to incorporate many more features. We can add an OLED display to display the surrounding parameters into it. We can also add a GPS module in the design so that the location of the surroundings will also be mailed or messaged to the user along with the surrounding parameters,like, temperature, humidity, pressure, light intensity etc. It can also be modified such that whenever a message or email is sent from a particular phone number or email id to the server, all the environmental parameters of the device along with its location will be delivered to that phone or email id. This device can also be used to monitor a particular room or place whose environmental parameters are required to be monitored continuously**.**

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.

[5] A. Lage and J. C. Correa, "Weather station with cellular communication network," in 2015 XVI Workshop on Information Processing and Control (RPIC), Oct 2015, pp. 1–5.

[6] T. Thaker, "Esp8266 based implementation of wireless sensor network with linux based web-server," March 2016.

[7] Y. Zhou, Q. Zhou, Q. Kong, and W. Cai, "Wireless temperature amp; humidity monitor and control system," in 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), April 2012, pp. 2246–2250.