**PROJECT REPORT**

On

***Weather Forecasting System***

*Submitted in partial fulfillment of requirements for the award of*

**Integrated Master of Science (Int. Msc.)**

In the department of

**Information Technology**



*Submitted by*:

**Aryan Raj (CS15MI0396)**

**Aftab Hussian (CS15MI0447)**

*Under the Supervision of*

**PROF. RATAN KUMAR SAHA**

PROFESSOR

**School of Computing Sciences**

**The Assam Kaziranga University, Jorhat, Assam**

**May 2020**

**SCHOOL OF COMPUTING SCIENCES**

**THE ASSAM KAZIRANGA UNIVERSITY**

**JORHAT-785006:: ASSAM :: INDIA**

**CERTIFICATE**

This is to certify that the project report entitled ***“Weather Forecasting System”****,* submitted to the School of Computing Sciences (SCS), **THE ASSAM KAZIRANGA UNIVERSITY, JORHAT, ASSAM,** in partialfulfillment for the completion of **Semester – X** of the degree of **Integrated Master of Science (Int. MSc.)** in the department of **Information Technology**, is a record of bona fide work carried out by **Mr. Aryan Raj**, **Roll No**. CS15MI0396, **Mr. Aftab Hussian**, **Roll No**. CS15MI0447 under my supervision and guidance.

All the help received by us from various sources have been duly acknowledged.

No part of this report has been submitted elsewhere for award of any other degree.

----------------------------- -----------------------------

Dean, SCS, KU **Faculty Supervisor**

**Department of Information Technology**

**Prof. Ratan Kumar Saha**

**Professor**

**ACKNOWLEDGEMENT**

We would like to express our special thanks of gratitude to Prof. Ratan Kumar Saha and Dr. Purnendu Bikash Acharjee who gave us the golden opportunity to do this wonderful project on the topic *“*Weather Forecasting System*”*, which also helped us in doing a lot of inspiration, encouragement and continuous support our Int. MSc. (IT) program.

We express my sincere gratitude to all persons who have helped me during the development of our project.

We are greatly indebted to Prof. Ratan Kumar Saha for his valuable guidance and encouragement which have been really helpful for our work.

The completion of my project work is possible only because of the generous help and guidance received by me from several persons.

Finally, we take our privilege to thank all our parents, seniors and class mates for their kind co-operation and help inside and outside the campus.

**DECLARATION**

We both, Aryan Raj and Aftab Hussian declare that all the materials embodied in the project report entitled **Weather Forecasting System** submitted to the School of Computing Science(SCS), The Assam Kaziranga University, Jorhat, Assam, by me as a 10th semester project in the partial fulfillment of the Integrated Master of Science(Int.M.Sc) in Information Technology(IT) is original and no part of the report submitted to any other institution or organization for the award of any degree or diploma. we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact from any other sources in my project report. For any violation of the above facts we shall remain solely responsible.

**Full Name Enrollment No. Signature of Student**

Aryan Raj CS15MI0396

Aftab Hussain CS15MI0447

**CONTENTS**

Certificate

Acknowledgement i

Declaration ii

1. Introduction 1

2. Literature Review 2

2.1 Literature Review#1 2

2.2 Literature Review#2 2

2.3 Literature Review#3 3

2.4 Literature Review#4 3

2.5 Literature Review#5 4

3. Project Details 5

4. Result and Discussion 24

5. Conclusion 27

6. Future work 28 Bibliography 29-30

**LIST OF FIGURES**

**Figure name Figure Number Page No**

|  |  |  |
| --- | --- | --- |
| NodeMCU | 3.1 | 6 |
| NodeMCU Circuit | 3.2 | 6 |
| DHT11 | 3.3 | 9 |
| Connection Between NodeMCU And DHT11 | 3.4 | 10 |
| Jamper Cable | 3.5 | 11 |
| Breadboard | 3.6 | 12 |
| Blynk Connection | 3.7 | 13 |
| Create Project On Blynk | 3.8 | 14 |
| Setting of Gauge | 3.9 | 15 |
| System Block Diagram Logical | 3.10 | 16 |
| System Block Diagram Physical | 3.11 | 16 |
| Output in Blynk | 3.12 | 19 |
| Voice Recognition | 3.13 | 20 |
| Output in Voice Recognition app | 3.14 | 21 |
| Remotely Accessible | 3.15 | 22 |
| System Flow Chat | 3.16 | 22 |
| Connection Between NodeMCU and DHT11 | 4.1 | 24 |
| Circuit After Getting Power Supply Through Battery | 4.2 | 25 |
| Voice Command App | 4.3 | 25 |
| Output of Temperature And Humidity | 4.4 | 26 |

**ABSTRACT**

The proposed system is one of the solution that has been planned for monitoring the weather conditions at a particular place and makes the information visible anywhere in the world. The technology behind this will be Internet of Things (IoT). This system will deal with monitoring the environmental conditions temperature and humidity. These values are to be captured using a microcontroller interfaced with sensors. In this IoT enabled weather forecasting system, NodeMCU measures 2 weather parameters using two sensors. These sensors are temperature sensor and humidity sensor. These two sensors are directly connected to NodeMCU since it has inbuilt Analog to digital converter. NodeMCU calculates and displays these weather parameters on LCD display. Then it sends value of these parameters to Internet using IOT techniques.

**Chapter 1**

**Introduction**

In IoT enabled weather monitoring system project, NodeMCU measures 2 weather parameters using respective 2 sensors. These sensors are temperature sensor and humidity sensor. These 2 sensors are directly connected to NodeMCU since it has inbuilt Analog to digital converter. NodeMCU calculates and displays these weather parameters on LCD display. Then it sends these parameters to Internet using IOT techniques. The process of sending data to the internet using Wi-Fi is repeated after constant time intervals. Then the user needs to visit a particular website to view this data.

The project connects and stores the data on a web server. Thus user gets live reporting of weather conditions. Internet connectivity with Wi-Fi is compulsory in this IoT weather monitoring project.

Through weather monitoring system we can collect the information about humidity and temperature and according to current and previous data we can produce the results in graphical manner in the system. After reviewing many articles, there are presently no papers that mention monitoring the combination of temperature, lighting and humidity in one integrated system and have actuators to modify these settings. In addition to this, there is one research paper that has discussed monitoring these three environmental conditions; however, there has been no mention about having actuators to modify. So our main idea was to coin a system that can sense the main components that formulates the weather and can be able to forecast the weather without human error.

**Chapter 2**

**Literature Review**

**2.1 Literature Review#1**.

Title: IoT Based Weather Monitoring System for Effective Analytics

Author: Ferdin Joe and John Joseph

Publication: International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8 Issue-4, April, 2019

**2.1.1 Objective**

This implementation of weather monitoring system using Raspberry Pi is done as per the specifications above and the data insights are generated in web based portal. The access to this data is available in the intranet with the current level of implementation and it could be made public when the data is made to store in cloud servers or other sources in the internet. This proposed system is the most compact unit for measuring weather parameters in regions suffering from the PM 2.5 pollution. This device in multiple nodes can be connected to the internet from various locations of study. This connectivity will aid the user to monitor the weather metrics corresponding to pollution over a centralized data analytics server.

**2.2 Literature Review#2**.

Title: Study of Recent Literature on Weather Monitoring Systems

Author : K. P. Rane, PhD

ETC Department Godavari COE

Jalgaon, MS, India

J. P. Chaudhari, PhD

CHARUSAT Space Research and Technology Center,

Changa, Gujarat, India

S. R. Pachpande Dept. of ETC, J T Mahajan COE Faizpur, MS, India

Madhuri P. Patil Dept. of ETC, Godavari COE, Jalgaon, MS, India

**2.2.1 Objective**

Climate observing has critical influence on mankind. Gathering of the various data of fleeting elements of the climate variations is extremely noteworthy. The essential point of this paper is to build up an installed framework to outline a climate observing framework which empowers the checking of climate parameters. This type of frame work includes various sensors involving temperature, Humidity, wind speed, wind bearing information can be signed into cloud so that any one (validated individual) from wherever can watch the particular information. If there should be an occurrence of any catastrophes like flame, substantial downpour, overwhelming wind, temperature or moistness might be wild, in these cases the moment data can be passed on all through the world utilizing cloud to the verified people, regardless of the fact that his equipment is wrecked in crisis

**2.3 Literature Review#3**.

Title: IoT Based Weather Monitoring System

Author : R Suresh Babu, T Palaniappan,K Anushya, M Kowsalya , M Krishnadevi

Department of Electronics and Communication Engineering, Kamaraj College of Engineering

and Technology, Madurai, Tamil Nadu, India

Publication: *International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), Vol. 5, Special Issue 13, March 2018*

**2.3.1 Objective**

By keeping the embedded devices in the environment for monitoring enables self-protection (i.e., smart environment) to the environment. To implement this need to deploy the sensor devices in the environment for collecting the data and analysis. By deploying sensor devices in the environment, we can bring the environment into real life i.e. it can interact with other objects through the network.

**2.4 Literature Review#4**.

Title: Internet of Things (IOT) based Weather Monitoring System

Author: Girija C, Andreanna Grace Shires, Harshalatha H and Pushpalatha H P, Department of Electronics and Communication, NIEIT, Mysuru

**2.4.1 Objective**

Department of Electronics and Communication, NIEIT, Mysuru. The system proposed in this paper is an advanced solution for monitoring the weather conditions at a particular place and make the information visible anywhere in the world. The technology behind this is Internet of Things (IoT), which is an advanced and efficient solution for connecting the things to the internet and to connect the entire world of things in a network. Here things might be whatever like electronic gadgets, sensors and automotive electronic equipment. The system deals with monitoring and controlling the environmental conditions like temperature, relative humidity and CO level with sensors and sends the information to the web page and then plot the sensor data as graphical statistics. The data updated from the implemented system can be accessible in the internet from anywhere in the world.

**2.4.2 Conclusion**

The implementation of weather monitoring system using Raspberry Pi is done as per the specifications above and the data insights are generated in web based portal. The access to this data is available in the intranet with the current level of implementation and it could be made public when the data is made to store in cloud servers or other sources in the internet. This proposed system is the most compact unit for measuring weather parameters in regions suffering from the PM 2.5 pollution. This device in multiple nodes can be connected to the internet from various locations of study.

**2.5 Literature Review#5**.

Title: A Smart Weather Monitoring System Using Internet of Things

Author: Chaw Myat Nwe1, Zaw Min Min Htun, Electronic Engineering Department, Mandalay Technological University, Mandalay, Myanmar

Publication: International Journal of Scientific Engineering and Research (IJSER), ISSN (Online): 2347-3878, Index Copernicus Value (2015): 56.67 | Impact Factor (2017): 5.156

**2.5.1 Objective**

The proposed system is an advanced solution for monitoring the weather conditions at a particular place and makes the information visible anywhere in the world. The technology behind this is Internet of Things (IoT). The system deals with monitoring the environmental conditions like temperature, humidity with respect to its measured time with a microcontroller interfaced with sensors, and GSM module to sends the information wirelessly to remote server and then plot the sensor data as graphical statistics.

**2.5.2 Conclusion**

This paper presents the research and implementation of a system for monitoring the environmental parameters using IoT scenario is accomplished. As the applications are limitless, other weather parameters can also be monitored easily with the addition of related sensors to the system architecture. 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 is tested in an indoor environment and it is successfully updated the weather conditions from sensor

data. It is also a less expensive solution due to usage of low power wired sensors and GPRS module.

**Chapter 3**

**Project Details**

**3.1 NodeMCU**

NodeMCU is an open source [LUA](https://www.lua.org/start.html) based firmware developed for ESP8266 wifi chip. By exploring functionality with ESP8266 chip, NodeMCU firmware comes with ESP8266 Development board/kit i.e. NodeMCU Development board.

Since NodeMCU is open source platform, their hardware design is open for edit/modify/build.

NodeMCU Dev Kit/board consist of ESP8266 wifi enabled chip. The **ESP8266** is a low-cost [Wi-Fi](https://en.wikipedia.org/wiki/Wi-Fi) chip developed by Espressif Systems with TCP/IP protocol. For more information about ESP8266, you can refer [ESP8266 WiFi Module](http://www.electronicwings.com/sensors-modules/esp8266-wifi-module).

There is Version2 (V2) available for NodeMCU Dev Kit i.e. NodeMCU Development Board v1.0 (Version2)**,** which usually comes in black colored PCB.

NodeMCU Dev Kit has NodeMCUlike Analog (i.e. A0) and Digital (D0-D8) pins on its board.

It supports serial communication protocols i.e. UART, SPI, I2C etc.

Using such serial protocols we can connect it with serial devices like I2C enabled LCD display, Magnetometer HMC5883, MPU-6050 Gyro meter + Accelerometer, RTC chips, GPS modules, touch screen displays, SD cards etc.

The prototyping hardware is a circuit board functioning as a [dual in-line package](https://en.wikipedia.org/wiki/Dual_in-line_package) (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on [breadboards](https://en.wikipedia.org/wiki/Breadboard). The design was initially was based on the ESP-12 module of the [ESP8266](https://en.wikipedia.org/wiki/ESP8266), which is a Wi-Fi SoC integrated with a [Tensilica](https://en.wikipedia.org/wiki/Tensilica" \o "Tensilica) Xtensa LX106 core, widely used in IoT applications.



Fig. -3.1 NodeMCU

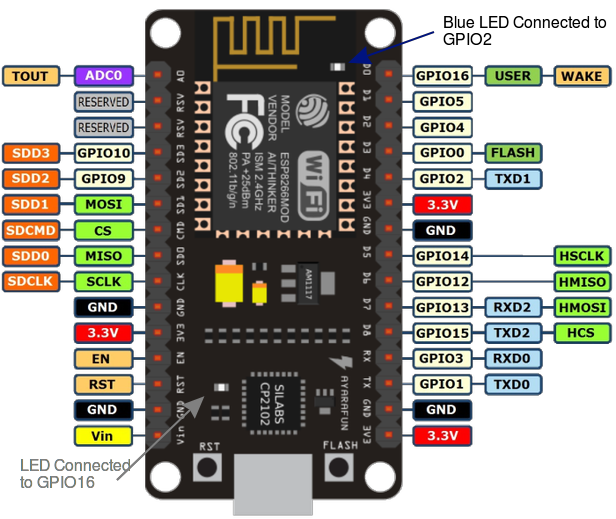


Fig. -3.2 NodeMCU Circuit

**3.1.1 Programming**

Working with ESP modules have become a lot easier since the NodeMCUCommunity has started to support it by providing its own library and board manager. This way you don’t have to use the hard way of learning the instruction documentation of the ESP module and program it using AT commands.

**Set up the NodeMCUIDE for ESP module and program it to blink a LED light**.

**Step 1:** Intall Arduino IDE. (Version is 1.6.5 or above)

**Step 2:** open Arduino IDE and follow the instruction :-

1. Navigate to *File -> Preferences* to open the below dialog box.
2. In the *“Additional Board managers URL”* paste the below link as shown in the image.
3. press OK

**Step 3:** Now, navigate to *Tool -> Boards -> Board Managers*. Search for ESP8266 by esp8266 community and click on install.

**Step 4:**  Now, Navigate to *Tools -> Boards -> Generic ESP8266 modules*.

**Step 5:** Now uploading the Example blink Program. This program can be found in *File -> Examples -> ESP8266 -> Blink*.

**Step 6:** Physical connectivity of module as follows:-

1. Connect physical ESP8266 Module and the FTDI board as shown above circuit and make sure your GPIO0 is pulled to ground.
2. To pull down the GPIO0, set the jumper switch as shown in below figure, we have used jumper switch to select between programming through AT command and through Arduino IDE.
3. Then power up the modules and reset it by pressing the pushbutton on the board.

**Step 7:** Now press the upload button, wait for the upload to complete.

After completing the above 7 steps, the system display the blue LED on blinking ESP module. This confirms that the program has been uploaded successfully.

After, this if we try to use the AT commands of the module it will not work. **To work with AT commands again we have to flash your ESP8266 module** with the firmware.

**3.1.2 Features:**

* 802.11 b/g/n
* Wi-Fi Direct (P2P), soft-AP
* Integrated TCP/IP protocol stack
* Integrated TR switch, balun, LNA, power amplifier and matching network
* Integrated PLLs, regulators, DCXO and power management units
* +19.5dBm output power in 802.11b mode
* Power down leakage current of <10uA
* 1MB Flash Memory
* Integrated low power 32-bit CPU could be used as application processor
* SDIO 1.1 / 2.0, SPI, UART
* STBC, 1×1 MIMO, 2×1 MIMO
* A-MPDU & A-MSDU aggregation & 0.4ms guard interval
* Wake up and transmit packets in < 2ms
* Standby power consumption of < 1.0mW (DTIM3)

**3.1.3 Specification:**

* Wi-Fi Direct (P2P), soft-AP
* Integrated TCP/IP protocol stack
* Integrated TR switch, balun, LNA, power amplifier and matching network
* Integrated PLLs, regulators, DCXO and power management units
* 19.5dBm output power in 802.11b mode
* Power down leakage current of < 1.0mW (DTIM3)

**3.1.4 Application of NodeMCU for Our Project**

* We are using NodeMCU and sensor DHT11 to capture the data of humidity and temperature from the atmosphere.
* Storing data of humidity and temperature of a remote location in a database, where these data will be uploaded in the cloud.

**3.2 Sensor DHT 11**

NodeMCU is used to interface DHT11 for humidity (in %) and temperature (in °C) measurement using single wire serial interface (SPI). The DHT11 is a commonly used as temperature and humidity sensor. The sensor can measure temperature from 0°C to 50°C with error of ± 2 °C and humidity from 20% to 90% with an accuracy of ±1°C and ±1% with RH ± 5% RH error. DHT11 works on 3-5.5V voltage supply and 0.5-2.5mA current supply.

The sensor comes with a dedicated Negative Temperature Coefficient (NTC) component to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity measurement resistive type component is used by this sensor.

The sensor is also factory calibrated and hence easy to interface with other microcontrollers with digital interface.

Output of DHT11 is calibrated digital signal which NodeMCU can understand and no need to have analog to digital converter.

In our project, we are looking for following measurement:

* Humidity of the atmosphere in the range of 20% to 90%.
* Temperature of the atmosphere in the range of 0°C to 50°C.

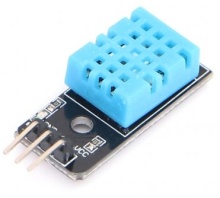


Fig. 3.3 DHT11

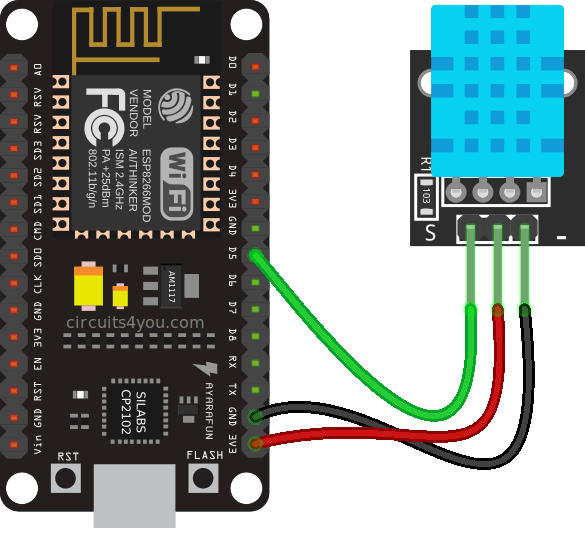


Fig.- 3.4 Connection Between NodeMCUand DHT11

**3.3 Jumper Cable**

Jumpers are like on /off switches they may be removed or added to alternate component performance options. A jumper is made of materials that conduct electricity and is sheathed in a nonconductive plastic covering to prevent accidental short circuit. The jumper’s main advantage is its one-time configuration, which make it less vulnerable to corruption or power failure than firmware.

Jumper wires typically come in three versions:-

* Male-to-male
* Male-to-female
* Female-to-female

The difference between each is in the end point of the wire. Male ends have a pin protruding and can plug into things, while female ends do not and are used to plug things into. Male-to-male jumper wires are the most common and what you likely will use most often. When connecting two ports on a breadboard, a male-to-male wire is what you’ll need.

What Do the Colors Mean ?

Though jumper wires come in a variety of colors, the colors don’t actually mean anything. This means that a red jumper wire is technically the same as a black one. But the colors can be used to your advantage in order to differentiate between types of connections, such as ground or power.

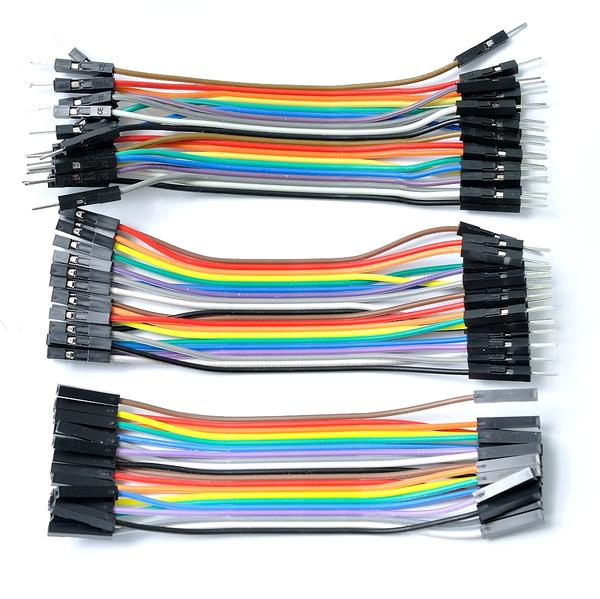


Fig.3.5 Jumper Cable

**3.4 Breadboard**

A **breadboard** is a widely used tool to design and test circuit. You do not need to solder wires and components to make a circuit while using a bread board. It is easier to mount components & reuse them. Since, components are not soldered you can change your circuit design at any point without any hassle. It consist of an array of conductive metal clips encased in a box made of white ABS plastic, where each clip is insulated with another clips. There are a number of holes on the plastic box, arranged in a particular fashion.

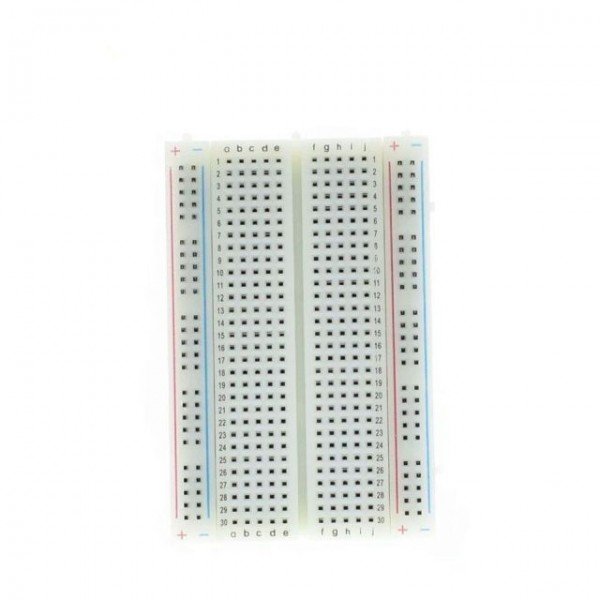
****

Fig. 3.6 Breadboard

**3.5 Blynk**

Blynk is a Platform with iOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet.

Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things.

It's a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets. It's really simple to set everything up and you'll start tinkering in less than 5 mins. Blynk is not tied to some specific board or shield. Instead, it's supporting hardware of your choice. Whether your Arduino or Raspberry Pi is linked to the Internet over Wi-Fi, Ethernet or this new ESP8266 chip, Blynk will get you online and ready for the Internet Of Your Things.

**How does it work?**

There are three major components in the platform:  
**Blynk App** – It allows to create interfaces to the projects using various widgets.

**Blynk Server** - Responsible for all the communications between the smartphone and hardware. we can use Blynk Cloud or run your private Blynk server locally. It’s open-source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.

**Blynk Libraries** - for all the popular hardware platforms - enable communication with the server and process all the incoming and out coming commands.

**Its features:**

* Supports majority of development boards like Arduino ,RPI, esp8266
* Easy to use
* Awesome widgets like LCD, push buttons, labelled value, graphs
* Not restricted to local Wi-Fi network
* Direct pin manipulation with no code writing
* Easy to integrate and add new functionality using virtual pins

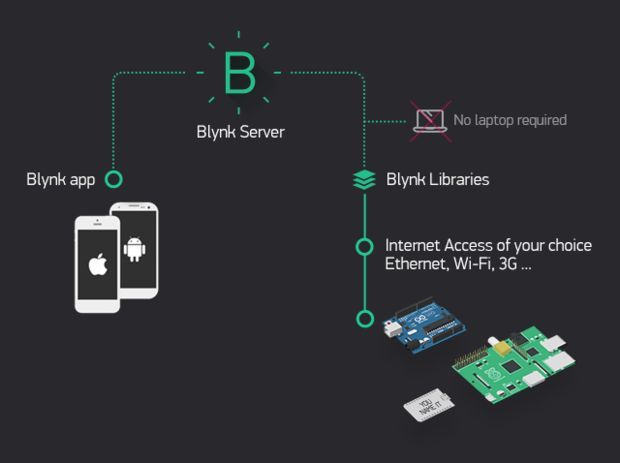


Fig. 3.7 Blynk Connection

**3.5.1 On Blynk App**

We need to perform following steps on Blynk App.

Step1:- Create a New Project in BLYNK app. Write Project name Temperature Humidity and Select NodeMCU from drop down.

Step2:- An AUTH token will be sent to your registered email, note this down. Tap on the screen and add 2 Gauges.

Step3:- Tap on the Widget and select the respective Virtual pins for temperature and humidity data (V0 for temperature and V1 for humidity).

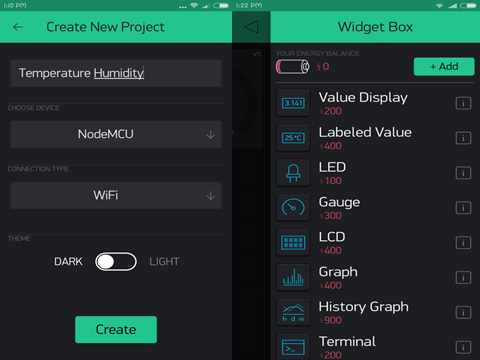
****

Fig. 3.8 Create Project On Blynk

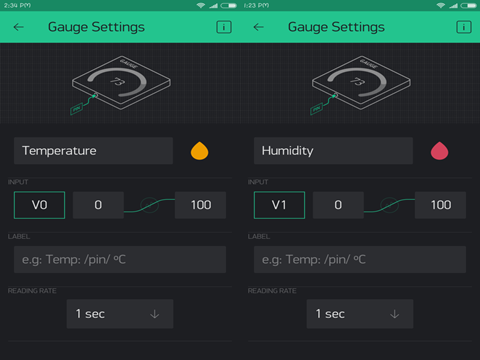
****

Fig. 3.9 Setting Of Gauge

**3.6 Infrastructure required**: To carry out this experiment, you will need:

**3.6.1 Hardware**

* NodeMCU
* Sensor DHT 11
* Jumper Cable
* Bread Board
* Laptop
* Moblie (Internet Connection & Voice Command App)

**3.6.2 Software**

* Arduino IDE
* Android Studio
* Voice Command App
* **3.7 System Block Diagram (Logical)**

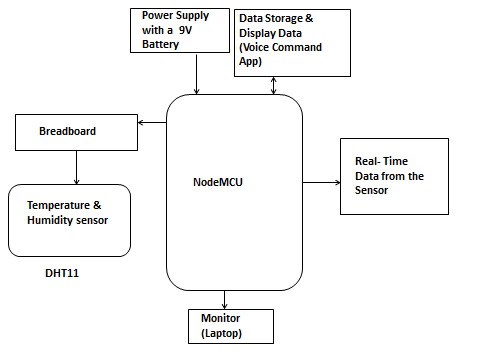


Fig. 3.10 System Block Diagram Logical

**3.8 System Block Diagram (Physical)**

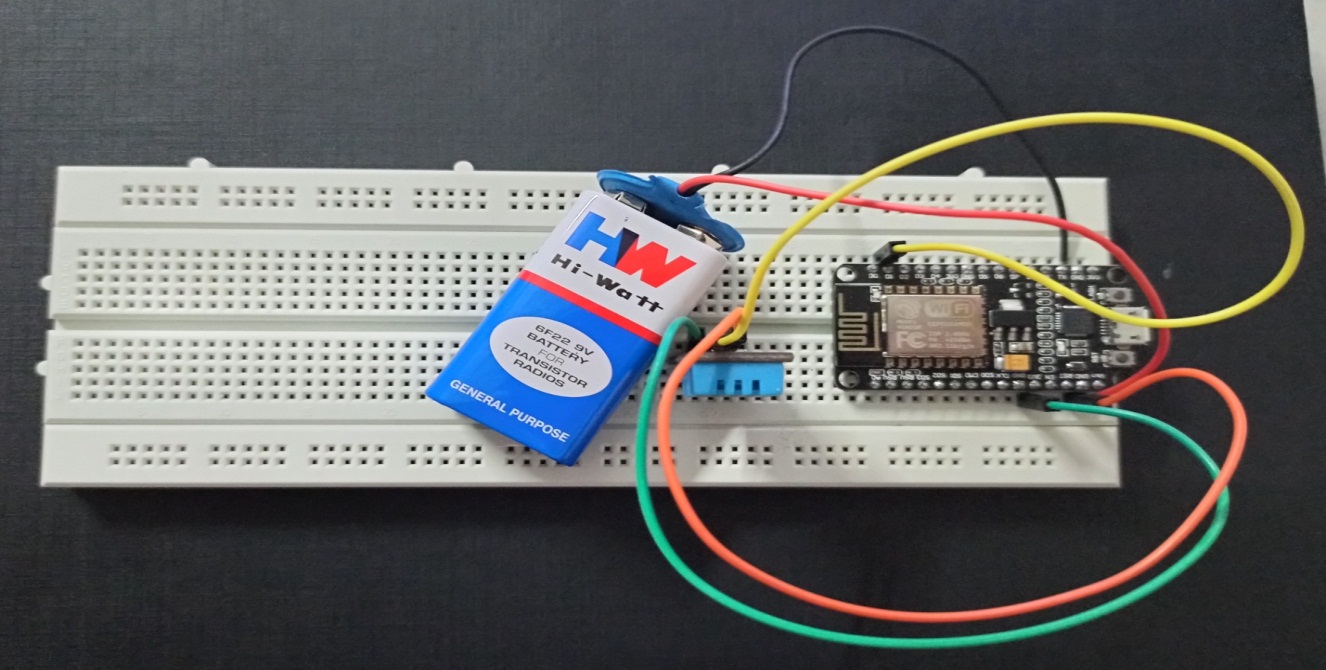
****

Fig. 3.11 System Block Diagram Physical

**3.9 The NodeMCU And DHT11 with the following code**

/\* How to use the DHT11 sensor with NodeMCU

Temperature and humidity sensor

\*/

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

#include "DHT.h"

#define DHTPIN D4

#define DHTTYPE DHT11 // DHT 11

char auth[] = "7EmYhpvu55Ah8ljjwyk3zTz\_sf93JqHD"; // Auth Token

char ssid[] = "OPPO A5 2020"; //Wifi Name

char pass[] = "123456789"; //Wifi Password

BlynkTimer timer;

DHT dht(DHTPIN, DHTTYPE);

void myTimerEvent()

{

float humi = dht.readHumidity();

float temper = dht.readTemperature();

if (isnan(humi) || isnan(temper))

{

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

return;

}

else

{

Serial.print("Humidity: ");

Serial.print(humi);

Serial.print(" %\t");

Serial.print("Temperature: ");

Serial.println(temper);

Blynk.virtualWrite(V1, humi);

Blynk.virtualWrite(V2, temper);

}

}

void setup()

{

// Debug console

Serial.begin(115200);

dht.begin();

Blynk.begin(auth, ssid, pass);

timer.setInterval(2000L, myTimerEvent);

}

void loop()

{

Blynk.run();

timer.run(); // Initiates BlynkTimer

}

**3.9.1 Output**

After Uploading the Ardunio code IDE. Press the play button on blynk app to show the output.

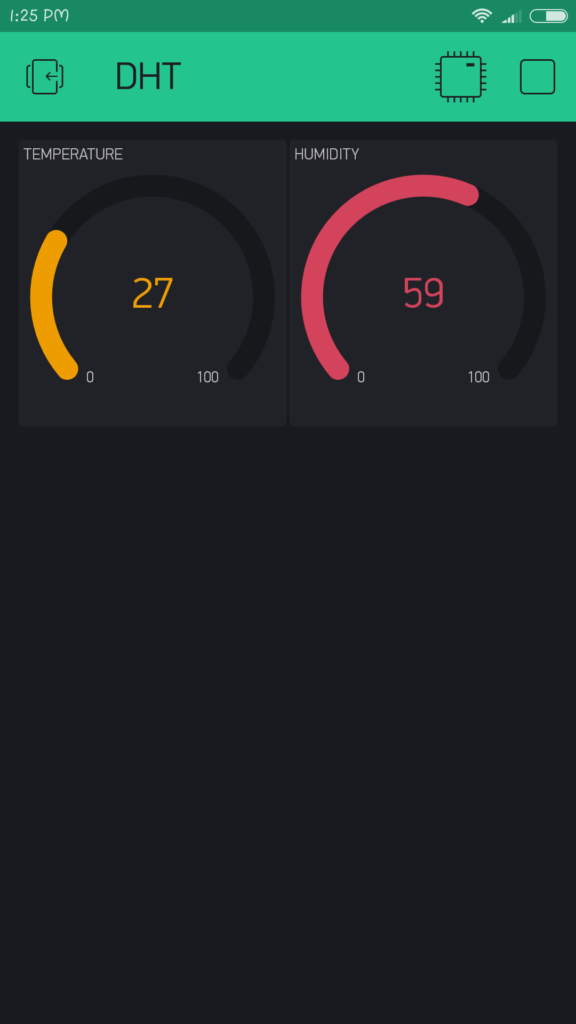


Fig. - 3.12 Output of humidity and temperature in Blynk

Our foremost objective was to display the data set in the appropriate app, here we have used the Blynk App for data display function.

**Current status**

1. After the data display, the function to access voice recognition was worked upon and is presently being carried out using a self-made app. (Voice Command )

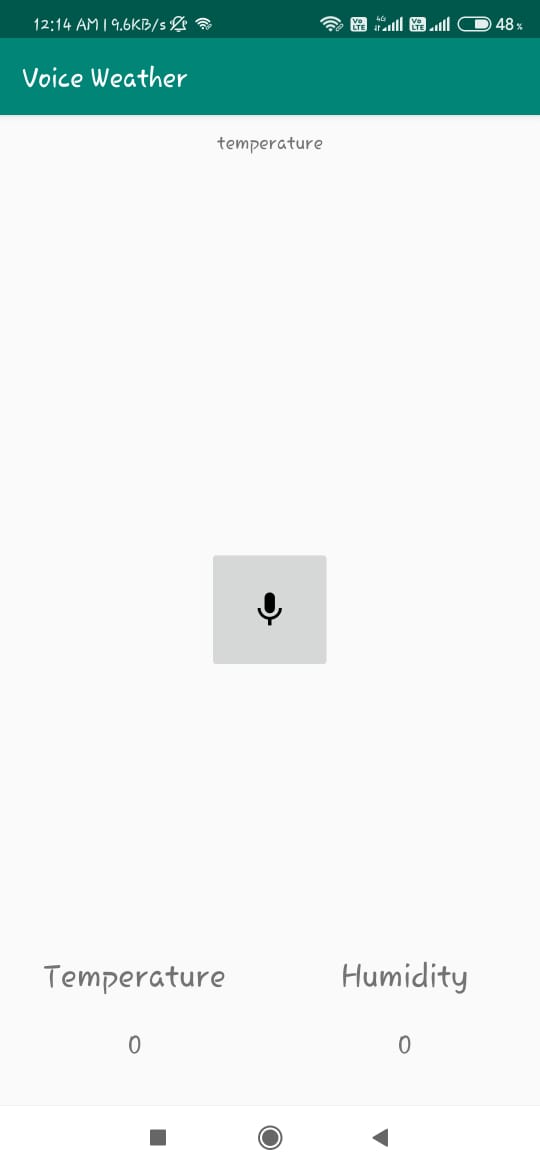


Fig.- 3.13 Voice recognition

2. The main objective to predict the current temperature and humidity has been analyzed so far.

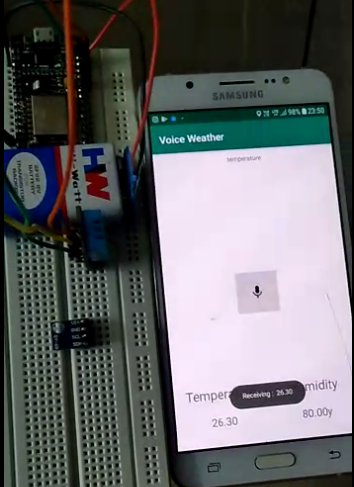


Fig.- 3.14 Output of temperature and humidity in voice recognition app

3. It's remotely accessible feature has also been satisfactory.

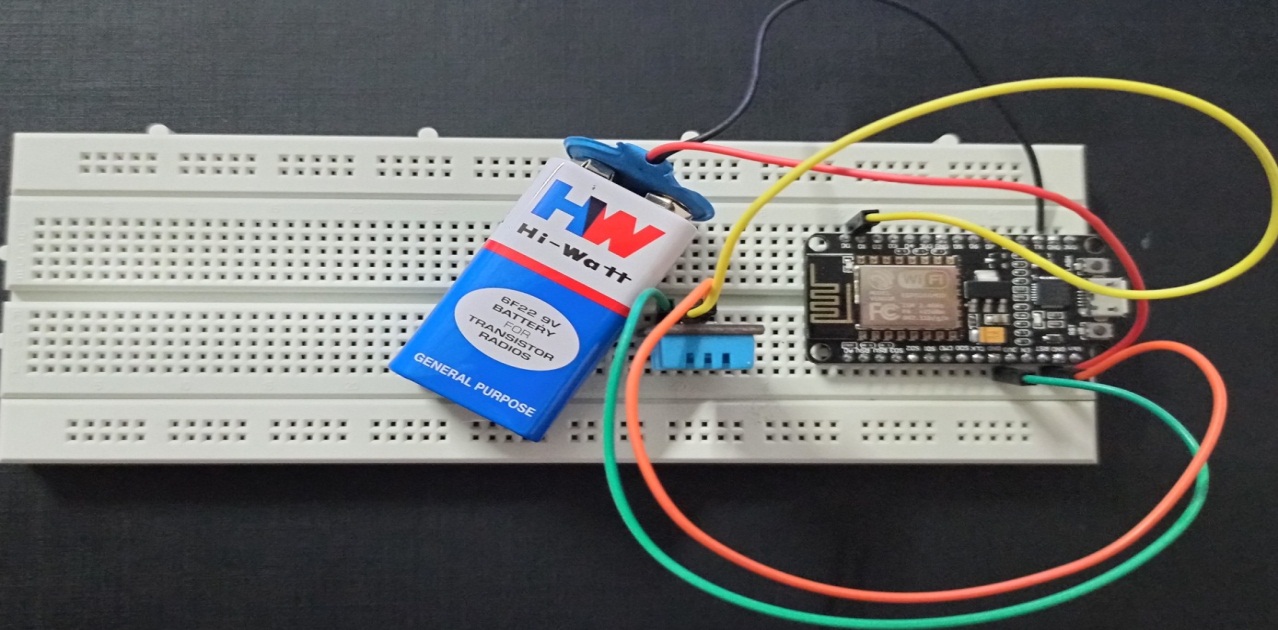


Fig.- 3.15 Remotely Accessible

**3.10 System Flow Chart**

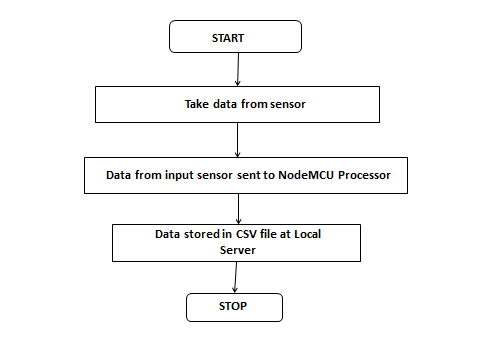
****

Fig. 3.16 System Flow Chart

**3.11 Input**

* Humidity and Temperature from the atmosphere
* Voice Recognize

**3.12 Output**

* Digital value for Humidity and Temperature as data to be stored in Local Server
* Also data to be uploaded to the cloud.

**3.13 Feature**

* Current Temperature and Humidity Predictability
* Portable
* Pocket Friendly
* User friendly
* Precise and Accurate Information

**Chapter 4**

**Result and Discussion**

The Connection between NodeMCU and DHT11 with Breadboard:

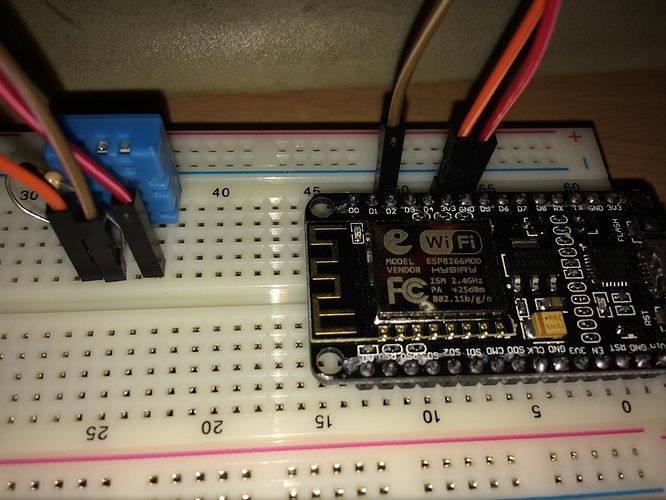
****

Fig.- 4.1 Connection Between NodeMCU and DHT11

Circuit after getting power supply through Battery:

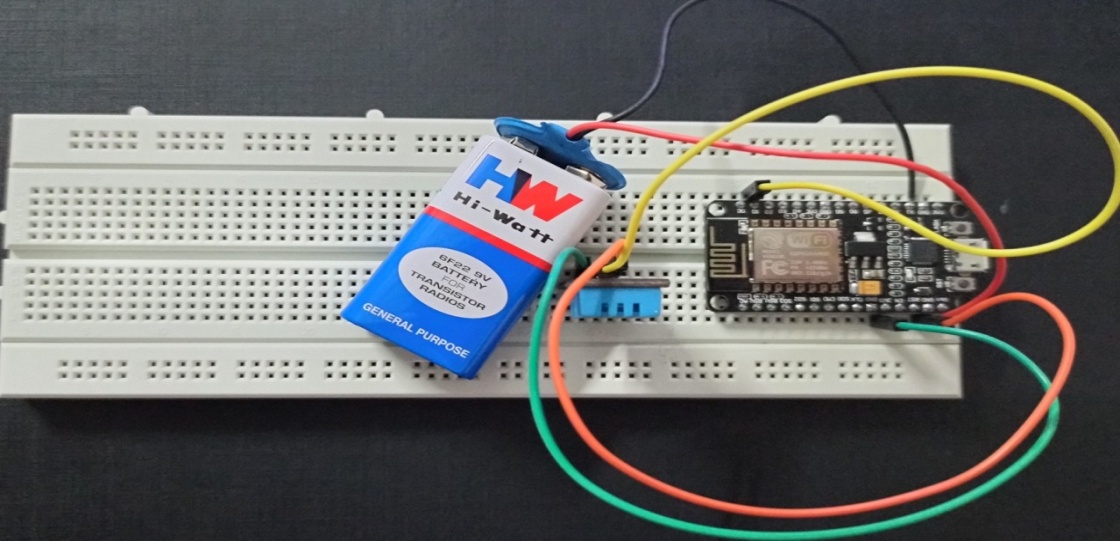
****

Fig.- 4.2 Circuit after getting power supply through Battery

A self-made Voice Command app:

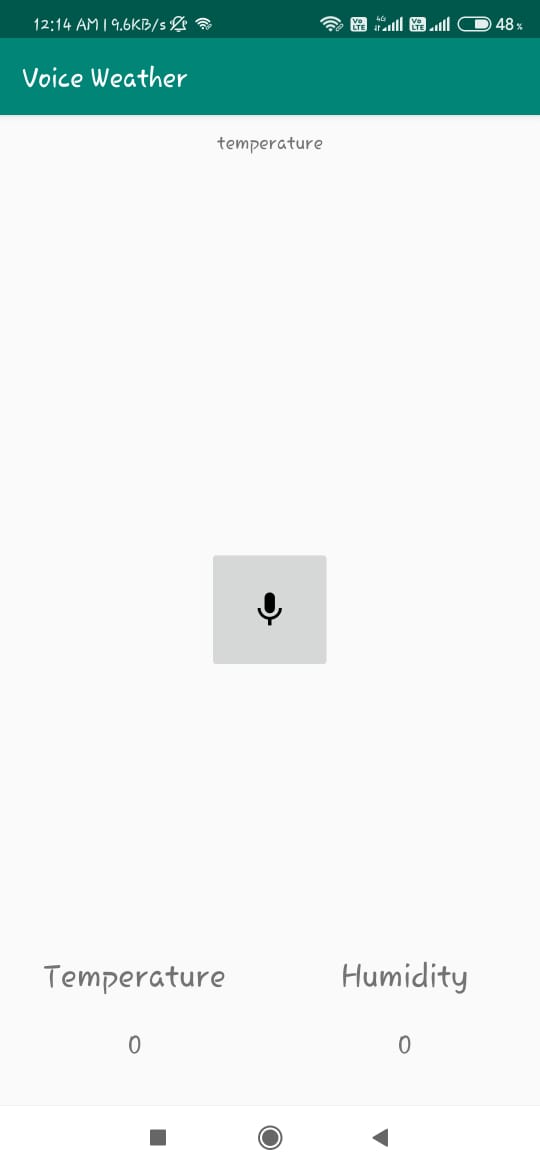
****

Fig.- 4.3 Voice Command app

Output of temperature and humidity in voice recognition app:

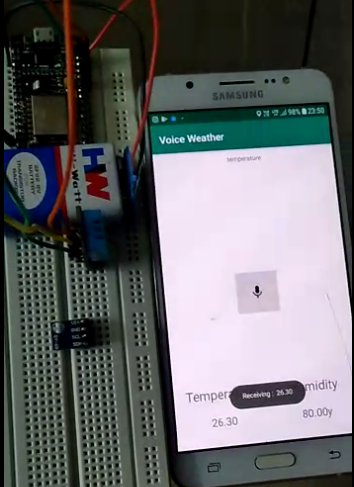
****

Fig.- 4.4 Output of temperature and humidity

**Chapter 5**

**Conclusion**

* Work Completed
* Arrangement of required infrastructure has been completed
* Testing of data collection process has been completed
* It is remotely accessible
* It operates through voice recognition

**Chapter 5**

**Future Work**

This model has been design for different features to be added to different sensor without changing any existing hardware modules. The addition of air pressure sensor, light sensor and rain level sensor which will be carried out as the scope of future work.

This model has been designed in such a way that it can cater to the need of the general public as well as the business class.

Air pressure sensor light sensor and rain level sensor can be helpful in weather forecast and can be studied further to get climate details.

The temperature sensor can be used to measure the temperatures in different automobiles to keep a track of the engine oil temperature, exhaust temperature and so on.

The humidity sensor can measure the humidity in the air along with temperature to maintain the comfort of people be it in the cabin, restaurants or office.

This design also support being eco-friendly and non-polluting, it can be the most popular sensors to be used in the day to day life of the people.

**Bibliography**

**Journal References**

[1] **“IoT Based Weather Monitoring System for Effective Analytics”** written by Ferdin Joe and John Joseph, *International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8 Issue-4, April, 2019*.

[2] **“Study of Recent Literature on Weather Monitoring Systems”** writeen by K. P. Rane, PhD ETC Department Godavari COE Jalgaon, MS, India, J. P. Chaudhari, PhD CHARUSAT Space Research and Technology Center, Changa, Gujarat, India, S. R. Pachpande Dept. of ETC, J T Mahajan COE Faizpur, MS, India Madhuri P. Patil Dept. of ETC, Godavari COE, Jalgaon, MS, India.

[3] **“IoT Based Weather Monitoring System”** writeen by R Suresh Babu, T Palaniappan,K Anushya, M Kowsalya , M Krishnadevi Department of Electronics and Communication Engineering, Kamaraj College of Engineering and Technology, Madurai, Tamil Nadu, India.

*International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), Vol. 5, Special Issue 13, March 2018*

[4] **“Internet of Things (IOT) based Weather Monitoring System”** writeen by Girija C, Andreanna Grace Shires, Harshalatha H and Pushpalatha H P, Department of Electronics and Communication, NIEIT, Mysuru

[5] **“A Smart Weather Monitoring System Using Internet of Things”** writeen by Chaw Myat Nwe1, Zaw Min Min Htun, Electronic Engineering Department, Mandalay Technological University, Mandalay, Myanmar. *International Journal of Scientific Engineering and Research (IJSER), ISSN (Online): 2347-3878, Index Copernicus Value (2015): 56.67 | Impact Factor (2017): 5.156.*

**URL References**

[1] <https://www.hackster.io/amruthp/nodemcu-based-iot-project-connecting-temperature-sensor-4f20d5> October 7, 2019 at 2:50 PM

[2] <https://iotdesignpro.com/projects/iot-based-esp32-wi-fi-weather-station-using-dht11-and-bmp-180-sensor>October 18, 2019 at 3:35 PM

[3]<https://circuitdigest.com/microcontroller-projects/esp12-nodemcu-based-iot-weather-station> November 4, 2019 at 4:20 PM

[4] <https://www.instructables.com/id/DIY-Weather-Station-Using-DHT11-BMP180-Nodemcu-Ove/> November 24, 2019 at 12:50 PM

[5] <https://www.youtube.com/watch?v=0dbws1i2GoE> December 7,2019 at 2:15 PM

[6] https://www.blynk.cc/ January 8, 2020 at 9:30 AM

[7] <https://github.com/blynkkk/blynk-library/releases/tag/v0.5.2>January 10, 2020 at 6:00 AM

[8] <https://www.youtube.com/watch?v=HFUR9uSZFQY> February 03, 2020 at 1:00 PM