# IoT Weather Reporting System

## Minor Project Report

Submitted for the partial fulfilment of the degree of

## Bachelor of Technology

In

## Internet of Things (IOT)

**Submitted By**

**Sanskar Puniyani**
**0901IO221062**

**UNDER THE SUPERVISION AND GUIDANCE OF**

**Dr. Nookala Venu**
**Assistant Professor**

## Centre for Internet of Things

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR (M.P.), INDIA
माधव प्रौद्योगिकी एवं विज्ञान संस्थान, ग्वालियर (म.प्र.), भारत
Deemed to be university
NAAC ACCREDITED WITH A++ GRADE

**July-Dec 2024**

# DECLARATION BY THE CANDIDATE

I hereby declare that the work entitled **"IoT Weather Reporting System"** is my work, conducted under the supervision of **Dr. Nookala venu, Assistant Professor,** during the session July-Dec 2024. The report submitted by me is a record of bonafide work carried out by me.

I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Sanskar Puniyani

**Sanskar Puniyani**
**0901IO221062**
**B.Tech. V Sem**

**Date: 20/11/2024**
**Place: Gwalior**

This is to certify that the above statement made by the candidates is correct to the best of my knowledge and belief.

**Guided By:**

**Dr. Nookala Venu**
**Assistant Professor**
Centre for Internet of Things
MITS, Gwalior

**Departmental Project Coordinator**

**Dr. Nookala Venu**
**Assistant Professor**
Centre for Internet of Things
MITS, Gwalior

**Approved by HoD**

20/11/24

**Dr. Praveen Bansal**
**Assistant Professor**
Centre for Internet of Things
MITS, Gwalior

# PLAGIARISM CHECK CERTIFICATE

This is to certify that I/we, a student of B.Tech. in **Internet of Things (IOT)** have checked my complete report entitled **"IoT Weather Reporting System"** for similarity/plagiarism using the "Turnitin" software available in the institute.

This is to certify that the similarity in my report is found to be 14% which is within the specified limit (20%).

The full plagiarism report along with the summary is enclosed.

Sanskar Puniyani

**Sanskar Puniyani**

**0901IO221062**

**Checked & Approved By:**

Soumyajit Ghosh 20/11/24

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

**Dr. Soumyajit Ghosh**
**Assistant Professor**
Centre for Internet of Things
MITS, Gwalior

PAPER NAME

weather report.pdf

| WORD COUNT | CHARACTER COUNT |
| --- | --- |
| 4486 Words | 27468 Characters |
| PAGE COUNT | FILE SIZE |
| 29 Pages | 609.6KB |
| SUBMISSION DATE | REPORT DATE |
| Nov 19, 2024 4:34 PM GMT+5:30 | Nov 19, 2024 4:35 PM GMT+5:30 |

## 14% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 9% Internet database
- 1% Publications database
- Crossref database
- Crossref Posted Content database
- 14% Submitted Works database

## Excluded from Similarity Report

- Bibliographic material
- Small Matches (Less then 10 words)

## ABSTRACT

*The increasing demand for real-time environmental monitoring has led to the development of various IoT-based weather reporting systems. This project presents the design and implementation of an IoT weather reporting system using the NodeMCU (ESP8266) microcontroller and the Blynk platform. The system utilizes a DHT11 or DHT22 sensor to measure temperature and humidity levels, which are then transmitted to the Blynk app via Wi-Fi.*

*In this system, the NodeMCU reads data from the DHT sensor and sends it to the Blynk cloud platform, where it is displayed in real-time on user-friendly graphical widgets. Additionally, the system is configured to trigger email alerts if specific weather conditions are met, such as high temperatures exceeding a predefined threshold. This feature ensures that users are promptly notified of critical weather changes.*



***Keywords- IoT, NodeMCU(ESP8266), DHT11, Email alert***

# ACKNOWLEDGEMENT

The full semester Internship/ Project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology & Science** to allow me to continue my disciplinary/interdisciplinary Internship/ Project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would also like to extend my gratitude to the entire team at **Madhav Institute of Technology and Science**, especially **Dr. Nookala Venu**, for his collaboration and support during my time with the organization. I am grateful for the opportunity to work alongside such talented individuals and to learn from their expertise and experiences.

I would sincerely like to thank my department, **Centre for Internet of Things**, for allowing me to explore this project. I humbly thank **Dr. Praveen Bansal**, Assistant Professor and Coordinator, Centre for Internet of Things, for his continued support during the course of this engagement, which eased the process and formalities involved. I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Dr. Nookala Venu**, Assistant Professor, Centre for Internet of Things, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

**Sanskar Puniyani**

**0901IO221062**

# CONTENT

## ACRONYMS

1. IoT - Internet of Things
2. Wi-Fi - Wireless Fidelity
3. GPIO - General Purpose Input/Output
4. IDE - Integrated Development Environment

## NOMENCLATURE

1. IoT (Internet of Things): A network of interconnected devices that communicate and exchange data with each other over the internet, enabling remote monitoring and automation.
2. ESP8266: A low-cost microcontroller with built-in Wi-Fi capabilities, used in IoT applications for connecting devices to the internet and facilitating communication between sensors and cloud platforms.
3. NodeMCU: A development board based on the ESP8266 microcontroller, commonly used for IoT projects due to its ease of use and Wi-Fi capabilities.
4. DHT11: A low-cost sensor used for measuring temperature and humidity, commonly used in weather monitoring systems.
5. DHT22: A more accurate version of the DHT11 sensor, used for measuring temperature and humidity.
6. Blynk: A cloud-based platform for building mobile applications to control and monitor IoT devices. It allows users to create custom dashboards for visualizing data from connected devices.
7. Wi-Fi: A wireless networking technology that allows devices to communicate with each other and the internet without physical connections.
8. GPIO (General Purpose Input/Output): A type of pin on a microcontroller that can be configured as either an input or an output, used for various functions like reading sensor data or controlling devices.
9. Arduino IDE: An integrated development environment (IDE) used to write and upload code to Arduino and compatible microcontrollers, like the ESP8266.
10. BME280: A sensor used to measure temperature, humidity, and atmospheric pressure, providing more precise environmental data compared to DHT11 and DHT22.
11. Cloud Platform: A remote system or server (such as Blynk or ThingSpeak) that stores, processes, and makes IoT data accessible from anywhere over the internet.
12. API (Application Programming Interface): A set of protocols and tools that allow different software applications to communicate with each other (not explicitly mentioned in the provided text, but relevant to IoT systems that interact with platforms like Blynk).

13. Thresholds: Predefined limits for temperature or humidity levels, used to trigger notifications or alerts in the system when certain conditions are met.
14. Notification System: A mechanism that sends alerts or updates to users based on predefined conditions, such as when temperature or humidity exceeds a certain threshold.
15. Real-Time Monitoring: The continuous and live tracking of environmental conditions like temperature and humidity, displayed on a user interface (e.g., Blynk app).

# LIST OF FIGURES

# List of Tables

# CHAPTER 1: INTRODUCTION

Currently, IoT technology has expanded in different fields and offered better features for monitoring and controlling processes in real-time. Environmental monitoring is one of the fields that has experienced a positive impact of this technological innovation. Weather information is crucial for many applications ranging from simple human comfort as well as climate control to farming and disaster response. While current conventional weather stations are very precise, they are large, expensive, and rigid for small-scale or individual usage. While the use of barometers provides the user with real time weather information the use of IoT technology is less expensive and can be used from a distance.

The anticipated project of this work is to design an IoT weather reporting system that utilizes the NodeMCU (ESP8266) microcontroller and the Blynk. The system is type for sensing and indicating temperature and humidity utilizing a DHT11 or DHT22 humidity sensor. With this, the system gets connected to the Blynk cloud platform over Wi-Fi and controls and monitors the data acquisition system. I think the system can track environmental conditions from nearly every corner with ease. In contrast with most of the current solutions, this project features Blynk's Event and Notification, which will enable users to set the desired temperature and humidity levels and receive a notification when these parameters are crossed.

The primary objectives of this project are:

i. Real-Time Data Monitoring:

To continuously measure and display temperature and humidity data on the Blynk platform, allowing users to monitor environmental conditions in real-time from anywhere.

ii. Automated Alerts:

To implement automated alerting mechanisms using Blynk's Event and Notification feature, which sends email notifications when specific conditions, such as high temperatures, are detected. This ensures that users are continuously informed of critical changes in weather conditions.

iii. User-Friendly Interface:

To provide a seamless and intuitive interface through the Blynk platform, where users can easily visualize data and manage alerts.

This project shows how NodeMCU microcontroller could be connected to Blynk for the advancement of IoT technology in weather monitoring and control. Its flexibility of application makes this system relevant to different users in both the public and in the private setting for environment management. These objectives form the foundation of appreciable importance of the project, objectives and the technology employed in developing an efficient weather reporting system.

# CHAPTER 2: LITERATURE SURVEY

Late research literature focuses significantly on applying IoT in weather monitoring because IoT helps people obtain real-time, accessible, and accurate environment information in different fields, such as agriculture, urban planning, disaster management, and so on. The conventional weather monitoring systems have been characterized by large, costly fixtures, which need maintenance frequently and operation from qualified personnel. However, IoT makes it possible to install low-cost sensors and Micro controllers in society thus making the whole society equipped with a decentralized, portable, and affordable way of establishing a system for monitoring the environment.

## A. Related Work

A number of prior works have investigated weather monitoring systems based on the IoT borne by microcontrollers such as ESP8266 or Arduino, and sensors including DHT11 and BMP180. For example, Kumar et al.(2021) constructed an IoT weather station that measures temperature, humidity, and air quality Using an ESP8266 microcontroller and passes data to cloud-based application which is available to monitor via a mobile application. The cost effective and simplicity of using IoT devices was also pointed out in this study. Nonetheless, it highlighted some of the issues such as connectivity network and the limitations posed by the less intuitive techniques to make interface as simple as possible to be used by non-researchers.

Sharma et al. (2020) also created a system of the DHT11 sensor and ESP8266 targeting precision agriculture. Their system provided capabilities to monitor weather conditions giving farmers real-time information on when to water crops or embark on cultivation. The following limitations were however perceived with reference to the system; the system scalability issues and the inability of the farmers to set certain preferences on notification depending on the set thresholds such as an extreme weather condition. As highlighted in these studies, there is a need for IoT solutions that not only purchase real time data but also implement a form of alert mechanism that can be customized to suit a user's needs.

## B. Existing Systems

There are many commercial IoT weather stations available, some of them being Netatmo weather station and Ambient Weather station that provide all kinds of weather information through their application. However, these solutions are generally costly and developed for the commercial or residential purpose other than small-scale, focused application such as, smart farming or community surveillance. Available free of charge, such systems pose a challenge in that they may be complex to install and maintain and hence have not achieved mass deployment as would be expected. In this context, ThingSpeak and Blynk are basic platforms for IoT data visualization and remote control. Of those, Blynk has become a favorite for both its friendly interface and the fact it is cloud based so users can view data and receive updates at any time. According to Bhatt, Patel & Mistry, Hsu & Wei, and Habib, Blynk's ability to support the design of widgets makes it ideal for IoT devices that have to engage the user and be configurable. However, even so, there is still a need for practical cost-friendly systems for generating real-time alerts and which does not need high level expertise in installation and application.

### C. Problem Definition

Weather tracking is significant in many industries but current solutions offer reduced affordability, flexibility, or personalization for users in the cases when hyperlocal, precise information is called for. Traditional systems are often expensive, hard to manage and can scale, and not very user friendly limiting the use in L540 for individuals, small organizations and local communities. IoT based systems are a potential solution to this scenario but as we can see many current IoT solutions are still lacking in various aspects such as sophisticated alerting and easy to use interfaces. This does this through this project, through the creation of an affordable and efficient IoT weather monitoring system allowing users to track current weather conditions and receive an alert when certain thresholds are triggered.

### D. Problem Explanation

The problems existing in current solutions include impossibility of constant access and limited adaptability coupled with a high price. Advanced systems are more often tailored to be a part of commercial or research infrastructure; they are out of reach for those who may require a relatively basic, localized monitoring system for a farm, conservation project or one's own backyard. Furthermore, although many DIY weather stations have emerged due to IoT, many of these are complex and need a lot of technical input to set up, especially in terms of notifications or configuring the alert level. Most IoT systems do not incorporate customizable notifications further limiting users who require alerts pertaining to certain values.

## E. Limitations of Existing Systems

While there has been some improvement regarding the weaknesses of the traditional weather monitoring systems through IoT weather monitoring solutions, the following challenges are likely to persist. Some of the systems depend on internet connection, accessing which is impossible in some of the rural areas. Furthermore, user interfaces are not designed with the ordinary user in mind; the configurations of these interfaces require technical personnel. The absence of a clear alert generation option implies that users must always go through their systems to watch conditions, consequently limiting the realism of these solutions in dynamic operations where real-time information is desirable. It also means we need improved, cheaper, and accessible solutions; many of the existing IoT weather systems require costly or patented parts.

# Chapter 3: Proposed System

This concept of IoT-based weather monitoring is intended to give users environmental information all the time through a mobile application together with an additional feature of giving alerts depending on a user-defined temperature and humidity level. This paper focuses on design of the proposed weather monitoring and alert system using ESP8266 microcontroller, Blynk platform and DHT11 temperature and humidity sensor microcontroller to meet the demand of the easily implemented and user-friendly displayed weather monitoring and alert systems for the specific locations.

The system works through continuously collecting and transmitting temperature and humidity data through the DHT11 sensor which has the capability to serve IoT applications at a comparatively cheap price. This sensor delivers its information to the ESP8266 microcontroller, which sends it through Wi-Fi to the Blynk cloud platform. When connected to Blynk, users are able to monitor the weather data from any internet connected devices using interface from Blynk mobile application. This interface enables the users not only to check the updating weather information but also to set up the specific notification via Blynk Event and Notification function..

**A. System Architecture:**

i. ESP8266 Microcontroller: The ESP8266 serves as the primary control unit for this project, managing sensor data collection and transmission. Its built-in Wi-Fi module facilitates a direct connection to the internet, enabling seamless data transfer to the Blynk cloud. The ESP8266's low power consumption and high efficiency make it ideal for continuous monitoring applications.

ii. DHT11 Sensor: This sensor measures temperature and humidity with reasonable accuracy for standard weather monitoring needs. Its compact design and ease of integration with the ESP8266 enable efficient and reliable data collection.

iii. Blynk Cloud Platform: Blynk acts as the primary user interface, hosting data and displaying real-time information on the user's mobile device. It also provides tools for setting threshold-based notifications, so users receive alerts if the temperature or humidity surpasses preset levels. This platform facilitates a convenient, interactive way for users to monitor data without needing advanced technical knowledge.

**B. Functionalities:**

i. Real-Time Monitoring: The DHT11 sensor continuously reads temperature and humidity, relaying this data to the ESP8266. The microcontroller processes and sends the data to the Blynk cloud, where users can access it through a smartphone app in real-time.

ii. Customizable Notifications: By using Blynk's Event and Notification feature, users can define specific thresholds for temperature and humidity. If the environmental parameters exceed these thresholds, the system immediately sends a notification to the user's mobile device, allowing prompt action.

iii. Data Storage and Analysis: With Blynk's capabilities, users can track historical weather data for analysis and decision-making. This historical data can help users identify patterns and prepare for conditions based on past trends.

**C. System Flow:**

i. Knowledge wise, the DHT11 sensor acquires the temperature and humidity in the ambient atmosphere.

ii. ii. Therefore, the microcontroller of the ESP8266 reads the data from the DHT11 sensor and then analyzes it.

iii. iii. The final step is to use Blynk cloud platform that the ESP8266 sends the processed data under a Wi-Fi connection.

iv. iv. The user controls the smartphone, and via the Blynk application, they can see the data in real time.

v. v. If the temperature or humidity rises above limits set by the users, the Blynk platform raises a warning message to act quickly.

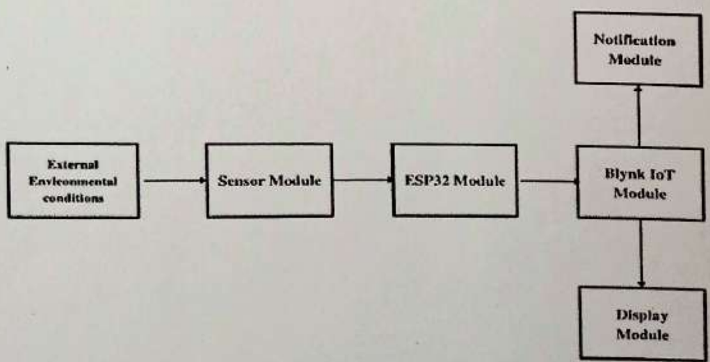


Fig.1

**D. Advantages of the Proposed System:**

This system provides several improvements compared to other weather monitoring systems by being cheaper, more easily usable and more flexible. The access and notification settings are mobile-based, which makes the device useful to a wide range of users such as agriculture, smart homes, environment monitoring, and many more. In addition, the system is portable and extensible in terms of increasing its interacted devices or qualities according to the users' requirement in the future.

Altogether, this proposed system provides a convenient and effective technology to monitor weather data in real-time and on a localized basis, using IoT to enhance the availability of weather data and the provision of useful information to users.

# CHAPTER 4: METHODOLOGY

The process of forming the IoT weather reporting system majorly entails the following tasks; The following methodology outlines the process:

**A. Hardware Setup**

i. Selection of Components:

- NodeMCU (ESP8266): Chosen for its built-in Wi-Fi capabilities, making it ideal for IoT applications.

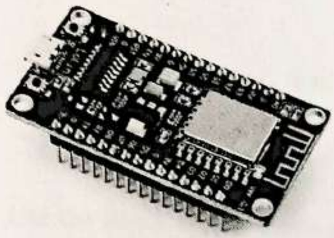

Fig.2

- DHT11 or DHT22 Sensor: Selected for its ability to measure temperature and humidity accurately.

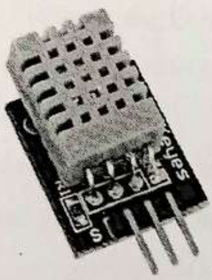

Fig.3

ii. Wiring and Connection:

- DHT Sensor Connection: Connect the sensor's VCC pin to the 3.3V power supply on the NodeMCU, the GND pin to ground, and the data pin to a digital GPIO pin (e.g., D4).
- Power Supply: Ensure that the NodeMCU is powered adequately either via a USB connection or an external power source if using it in a portable setup.

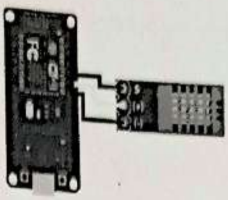

Fig.4

**B. Software Development**

i. Install Required Libraries:

- Arduino IDE Setup: Install the Arduino IDE and configure it for ESP8266 development.
- Library Installation: Install necessary libraries from the Library Manager, including the Blynk library and DHT sensor library.

ii. Configure Blynk Platform:

- Create a Blynk Project: Use the Blynk app to create a new project, select the ESP8266 as the hardware model, and obtain the Auth Token.
- Add Widgets: Set up graphical widgets in the Blynk app for real-time visualization of temperature and humidity data (e.g., Gauges or Labels).
- Event and Notification Setup: Configure the Event and Notification feature to send email alerts when specific conditions are met (e.g., high temperature).

iii. Write and Upload Arduino Code:

- Code Development: Write the Arduino sketch to read data from the DHT sensor, send it to Blynk, and trigger email notifications when conditions exceed predefined thresholds.
- Code Example: Refer to Appendix-1.

iv. Upload Code: Connect the NodeMCU to your computer and upload the code using the Arduino IDE.

**C. System Testing and Deployment**

i. Initial Testing:

- Monitor Serial Output: Verify the correct operation by checking the serial monitor for data from the DHT sensor.
- Check Blynk App: Ensure that temperature and humidity data are being displayed correctly and that email notifications are functioning as expected.

ii. Calibration and Adjustments:

- Threshold Adjustment: Fine-tune the temperature threshold for email alerts based on testing results.
- Widget Configuration: Adjust widget settings in the Blynk app for optimal display and functionality.

iii. Deployment:

- Field Deployment: Place the weather station in the desired location, ensuring that it has a stable Wi-Fi connection.
- Ongoing Monitoring: Regularly check the system for performance and make any necessary adjustments.

This methodology provides a comprehensive approach to developing, deploying, and maintaining an IoT weather reporting system, ensuring that it meets user needs and performs reliably in various conditions.

# CHAPTER 5: RESULT

The IoT-based weather monitoring system successfully met the project's objectives by providing real-time temperature and humidity data accessible through the Blynk mobile app. Key results of the implementation include accurate sensor data collection, efficient data transmission, and timely notifications for predefined environmental thresholds.


Fig.5

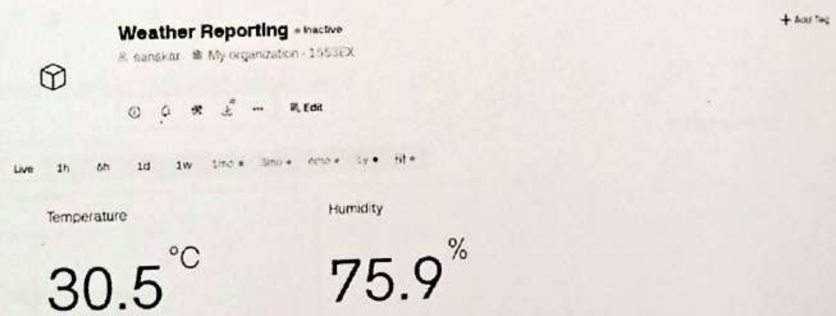


Fig.6

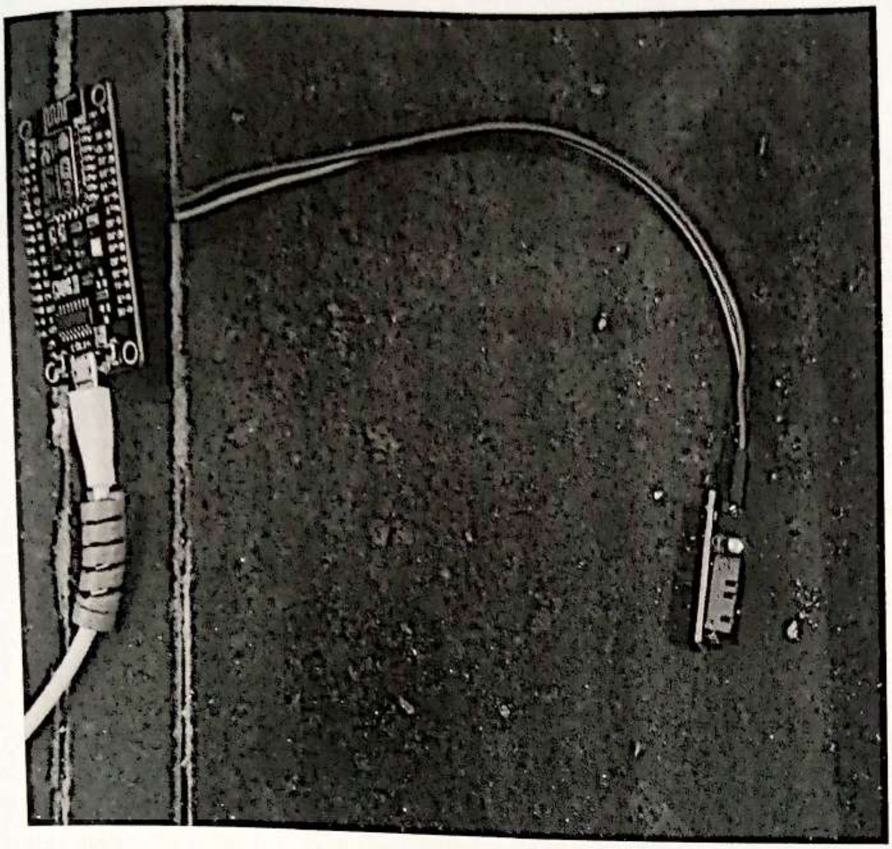

Fig.7

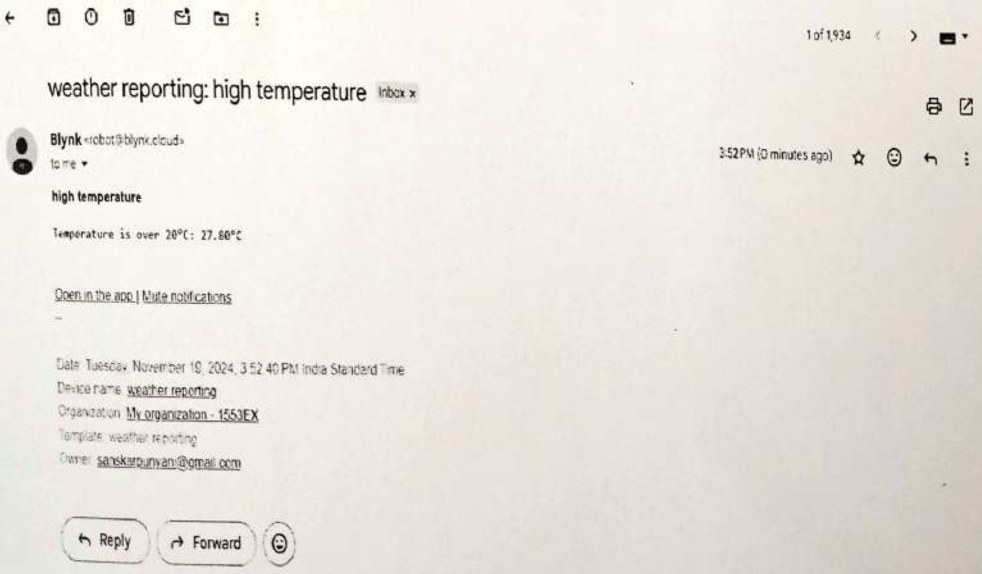
1 of 1,934
weather reporting: high temperature
Inbox x
Blynk <robot@blynk.cloud>
to me
3:52 PM (0 minutes ago)
high temperature
Temperature is over 20°C: 27.80°C
Open in the app | Mute notifications
Date: Tuesday, November 19, 2024, 3:52:40 PM India Standard Time
Device name: weather reporting
Organization: My organization - 1553EX
Template: weather reporting
Owner: sanskarpunyani@gmail.com
Reply
Forward

Fig.8

| Date, Time | Temperature | Humidity |
|---|---|---|
| 19/11/2024, 4:00p.m. | 25.8 | 41.3 |
| 19/11/2024, 4:01p.m. | 25.7 | 41.4 |
| 19/11/2024, 4:02p.m. | 25.4 | 42.1 |
| 19/11/2024, 4:03p.m. | 25.5 | 44.4 |
| 19/11/2024, 4:04p.m. | 25.4 | 42.7 |

Table.1

# CHAPTER 6: CONCLUSION

This project has proposing an IoT based weather monitoring system which is affordable and easy for real-time temperature and humidity checking using ESP8266, DHT11 sensor and Blynk application. IoT has been integrated successfully in the system, the user can get the weather updates through Blynk mobile app continuously and they can set up the alert notification for the threshold they set. This makes it convenient and responsive to timely decision making relative to localized weather information in different use cases such as farming, climate change studies, and intelligent housing.

Hence, through application of cheap hardware and a user friendly software the system is made available to the general populaion thereby increasing its need. When testing the possibility of the application monitoring real-time data it was noted that the notification system reacted instantly when pre-set boundary conditions for temperature or humidity were crossed.

Of course, certain restrictions were identified: namely, the Signal on Wi-Fi and a number of shortcomings of the DHT11 sensor – basically, it restricts the number of significant digits for data. Subsequent versions of this system can consider different sensors which are more precise and the continually improved offline data storage and manipulation ability of the system.

Therefore, this IoT-based weather monitoring system is among the best technological innovations that make environmental monitoring more practical, user-friendly, cheap, easily deployable in most complex and dynamic environments, and readily adaptable. This system forms a useful prototype for any person or firm in need of routine localized weather information and underlines the potential of IoT in practical environmental sensing.

# CHAPTER 7: FUTURE SCOPE

The possibilities for the further development of this IoT-based weather monitoring system are: the use of extra sensors for the IoT weather station (air humidity and pressure, wind speed and direction) for gathering, the usage of improved sensors such as the BME280, and the option to set up offline logging in case of no internet connection. The use of machine learning for predictive analysis and long-range communications such as LoRaWAN will further the applicability of the solution generally in rural and remote locations and solar power would further increase sustainability. Moreover, enhancing mobile application functionality towards data management, including visualization and trend analysis would benefit the user, while an open-source architecture would allow integrating the system into different fields.

# References

[1]. Raj, S., & Mishra, P. (2020). "IoT-based Weather Monitoring System." International Journal of Computer Science and Information Security, 18(7), 155-160.

[2]. Alarifi, A., Al-Salman, A., & Alsheikh, M. (2017). "Internet of Things (IoT) in Environmental Monitoring: A Review." Wireless Communications and Mobile Computing, 17(16), 2350-2373.

[3]. Patel, P., & Patel, V. (2020). "Wireless Weather Monitoring System Using IoT." International Journal of Engineering Research and Technology, 9(7), 1124-1130.

[4]. Bihani, V., & Patil, R. (2018). "IoT-Based Weather Monitoring System for Smart Agriculture." International Journal of Engineering & Technology, 7(3), 2951-2954.

[5]. Zhang, W., Yu, H., & Chen, X. (2020). "Development of IoT-based environmental monitoring system for smart cities." Sensors, 20(6), 1632.

[6]. Nayak, S. R., & Agarwal, S. (2019). "IoT-enabled weather monitoring system using ESP8266 and Blynk for agriculture." International Journal of Computer Science and Mobile Computing, 8(4), 48-54.

## APPENDIX-1

**Source Code:**

```
#define BLYNK_TEMPLATE_ID "YourTempelateID"
#define BLYNK_TEMPLATE_NAME "YourTempelateName"
#define BLYNK_AUTH_TOKEN "YourAuthToken"
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>
char auth[] = "YourAuthToken";
char ssid[] = "YourSSID";
char pass[] = "YourPassword";
#define DHTPIN D4
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
BlynkTimer timer;
void sendSensorData() {
 double h = dht.readHumidity();
 double t = dht.readTemperature();
 if (isnan(h) || isnan(t)) {
  Serial.println("Failed to read from DHT sensor!");
  return;
 }
```

```
Blynk.virtualWrite(V1, t); // Send temperature to Virtual Pin V1
Blynk.virtualWrite(V2, h); // Send humidity to Virtual Pin V2
// Temperature alert using Blynk notification
if (t > 20.0) {
  Blynk.logEvent("temp_alert", String("Temperature is over 20°C: ") + t + "°C");
 }
}
void setup() {
 Serial.begin(9600);
 Blynk.begin(auth, ssid, pass);
 dht.begin();
 timer.setInterval(5000L, sendSensorData);
}
void loop() {
  Blynk.run();
  timer.run();
}
```