International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

IOT-based Smart Weather Monitoring System

Sidam Masanta [1], Anindya Biswas [2], Soumik santra [3], Pritam Mondal [4], Dr. Manash Chanda [5], Dr. Sudipta Ghosh [6]

[1,2,3,4] Student, Department of ECE, Meghnad Saha Institute of Technology, Kolkata
[5] Associate Professor, Department of ECE, Meghnad Saha Institute of Technology, Kolkata
[6] Assistant Professor, Department of ECE, Meghnad Saha Institute of Technology, Kolkata

Abstract

The IoT-based Weather Monitoring and Reporting System developed in this work delivers real-time updates on key environmental parameters such as temperature, humidity, soil moisture, atmospheric pressure, and rainfall. It is particularly useful for scientists, researchers, and environmental analysts who need access to data from remote locations. Traditional SMS-based systems are limited by the number of recipients and slower transmission times. In contrast, the proposed IoT solution uses cloud-based connectivity, allowing instant and global access to data. The system includes both hardware and software components. An Arduino microcontroller forms the core, connected to various sensors including the DHT11 for temperature and humidity, BMP180 for pressure, and other sensors for rainfall and soil moisture. Data from these sensors is sent through the ESP8266 Wi-Fi module, powered by a USB Type-B or 5V DC source. The system is integrated with both the Blynk platform for mobile application and the ThingSpeak website. Blynk offers a user-friendly interface and allows up to 30,000 SMS alerts, while ThingSpeak enables data logging with a daily limit of 8,000 entries. Sensor data is transmitted using authentication tokens for secure communication. This efficient and scalable setup supports climate research, disaster prevention, and smart environmental monitoring

Keywords: Weather Monitoring system, Interne of Things, ThingSpeak

1. Introduction

Climate change and environmental monitoring have gained significant attention in recent years. Individuals and organizations require real-time weather updates for specific locations, such as college campuses or office buildings [1-3]. In response to this need, an IoT-based weather station is presented in this work, designed to provide accurate and real-time environmental data. The system is equipped with various environmental sensors, including the DHT11 sensor, soil moisture sensor, and rain drop sensor, integrated with an Arduino Uno microcontroller. These sensors continuously monitor weather parameters and transmit data to an online web server via a Wi-Fi connection. The collected information is uploaded to the cloud, enabling real-time reporting and analysis. This paper also explores the application of IoT in next-generation environmental monitoring system and its potential role in developing smart cities. The proposed system aims to enhance localized weather tracking, offering valuable insights for urban planning and environmental sustainability.

The Internet of Things (IoT) has emerged as a revolutionary advancement [4-6] in the global information industry, transforming traditional communication paradigms. IoT enables interconnected devices to exchange information, facilitating seamless data acquisition and management. By leveraging IoT technology, the proposed system enhances environmental monitoring, offering a scalable and efficient solution for weather tracking in smart city development. The weather station consists of an Arduino Uno microcontroller connected to multiple environmental sensors. The collected data through the sensors is transmitted to an online web server, where it is stored and processed for real-time access. Users can retrieve weather updates via a cloud-based interface, ensuring timely and accurate information dissemination.

IoT technology enables intelligent sensing, tracking, monitoring, and management of environmental parameters. By integrating microcontrollers, wireless communication modules, and cloud computing, IoT facilitates seamless data exchange between devices and users [7-8]. The proposed weather station exemplifies the potential of IoT in enhancing environmental awareness, contributing to sustainable development initiatives. The IoT-based weather monitoring system can be deployed in various settings, including office buildings, agricultural fields, and educational institutions. Its real-time data capabilities can assist in weather forecasting, disaster management, and resource optimization. Future enhancements may include the integration of machine learning algorithms for predictive analysis, expanded sensor networks for broader coverage, and enhanced connectivity through advanced communication protocols.

2. Literature Review

To predict weather conditions accurately has become a crucial challenge due to extreme climate states that adversely affect lives and property. Precise weather data is essential for enhancing weather prediction capabilities and building resilience against detrimental weather conditions. Developing countries like Uganda face challenges in acquiring timely and accurate weather data due to the scarcity of weather observation stations [9], primarily caused by the high cost of implementing automatic weather monitoring systems and limited funding for national meteorological services. To address these issues, an automatic weather monitoring station based on a wireless sensor network has been proposed. The plan involves developing three generations of automatic weather stations (AWS) prototypes, with the first-generation AWS being evaluated to improve subsequent versions. The study suggests enhancements in power consumption, data accuracy, reliability, and data transmission, ensuring a robust and cost-effective AWS that can be adopted by developing countries to improve weather forecasting.

Another study presents an IoT-based weather monitoring system [10] that retrieves environmental parameters using sensors such as temperature, humidity, pressure, and rain sensors, along with an LDR sensor to measure light intensity. The system calculates dew point values and integrates an SMS alert mechanism that notifies users when sensing parameters exceed predefined thresholds. Additional functionalities include email and social media alerts. The system employs Node MCU ESP8266 along with various sensors for data collection. The research paper [11] introduces a low-cost live weather monitoring system utilizing an OLED display to showcase real-time weather conditions. The system is built on an ESP8266-EX microcontroller-based We Mos D1 board, programmed using Arduino, which retrieves data from the cloud. The use of the Thing Speak platform for cloud storage ensures efficient data visualization on both OLED displays and the cloud.

Further research proposes a system [12] that not only monitors but also predicts weather conditions, allowing individuals to plan their daily activities accordingly. This system integrates sensor data, bus mobility data, and deep learning technology to provide real-time weather updates at stations and buses. Weather forecasting is achieved using a friction model that processes data from vehicles such as buses, incorporating local information processing for improved accuracy. An IoT-based weather monitoring system [13] highlights how IoT technology facilitates real-time climate monitoring, providing accurate and efficient weather data. This system uses swarm algorithms to enhance prediction accuracy, making it more effective for climate awareness and forecasting. The project employs various sensors to collect climate-related information, which is then stored in the cloud and accessed via an Android mobile application using an API key.

A system is designed [14] to keep an eye on weather conditions in industries using sensors for temperature, gas, and humidity. It uses an LPC1768 microcontroller to gather sensor data and sends it to a program called LABVIEW, which stores it in an Excel sheet. Plus, it can send alerts straight to your phone via SMS with a GSM module. The sensors track temperature, gas, and humidity and the mmicrocontroller sends data to LABVIEW and mobile via GSM. Embedded C programming is used with Keil uVision4 and JTAG for uploading code. The proposed work [15] introduces a weather monitoring system that tracks temperature, pressure, humidity, rainfall, wind speed, and wind direction using various sensors like rotary encoders and tipping buckets. The data is processed by a microcontroller and then shared wirelessly via Wi-Fi for continuous storage and access. This fully automated setup reduces human error and can be applied in fields such as agriculture, meteorology, and industry. The paper [16] reported the design and implementation of a low-cost, efficient, and accessible weather monitoring system using Internet of Things (IoT) technology. The system employs various sensors—such as DHT11 for temperature and humidity, MQ135 for air quality, and rain sensors—integrated with microcontrollers like Arduino UNO and NodeMCU ESP8266/ESP32. Data collected from these sensors is transmitted via Wi-Fi to cloud platforms such as ThingSpeak or Blynk, where it can be monitored remotely through web dashboards or mobile applications.

The work reported in [17] presents a cost-effective and efficient approach to monitor weather and environmental conditions using IoT technology integrated with machine learning. It addresses the limitations of existing systems that provide generalized data and proposes a localized monitoring setup using sensors like DHT11 for temperature and humidity, BMP180 for pressure, and MQ135 for air quality. These sensors are connected to a NodeMCU microcontroller, which collects data and uploads it to cloud platforms like ThingSpeak. The data can then be accessed globally via web and mobile applications. The system also uses machine learning algorithms, such as linear and multiple regression, to analyze and predict weather patterns. The collected data is processed using tools like Arduino IDE and R Studio. The paper [18] discusses the development of an IoT-based weather monitoring system that provides real-time data on temperature, humidity, and rain probability. The system uses a NodeMCU ESP8266 microcontroller integrated with DHT11 temperature and humidity sensor, a rain sensor, and an OLED display to collect and show environmental data. The information is also uploaded to the Cayenne IoT platform, which allows for remote visualization and alerts when thresholds are exceeded.

3. Objective of the Work

The objective of this work is to design and develop an IoT-based smart weather monitoring system that ensures real-time tracking of environmental parameters such as temperature, humidity, pressure, rainfall etc. The system aims to enhance the accuracy of weather predictions, address challenges faced by developing countries in acquiring reliable weather data, and provide an efficient, cost-effective solution through cloud integration. By leveraging advanced sensors, and wireless communication, the proposed system seeks to improve weather monitoring capabilities, enable proactive decision-making to support various platforms, including agriculture, industry, and disaster management.

3.1. Problem Statement

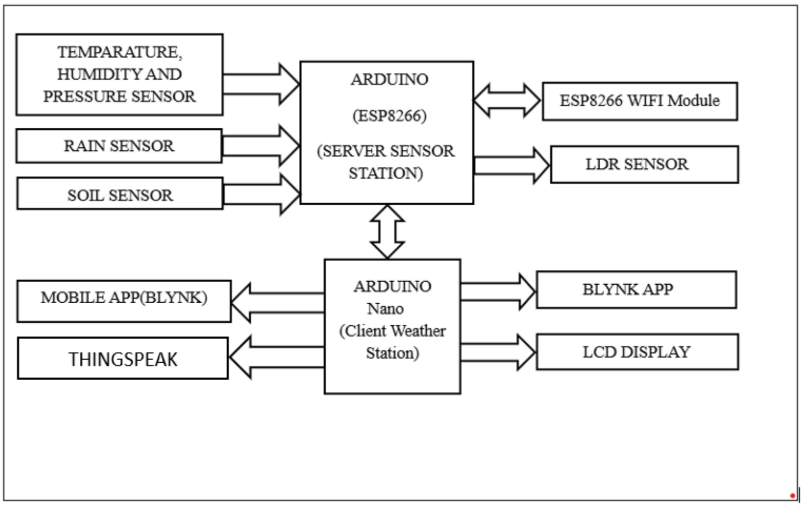
To design an IoT-based smart, semi-personalized weather monitoring system incorporating temperature, humidity, pressure, rainfall, soil moisture and LDR sensors to overcome the limitations of conventional satellite weather reporting systems, which lacks of specific location accuracy, real-time visualization, and automated alert generation mechanism for detecting weather abnormalities.

3.2. Block Diagram of Proposed Hardware

This project outlines a weather monitoring system with both server and client components. At the server side, the SERVER SENSOR STATION utilizes an ESP8266 Wi-Fi module to collect data from several sensors. These sensors include a temperature and humidity sensor (DHT11), pressure sensor (BMP180), a rain sensor, a soil moisture sensor, and a LDR (LM393) sensor for light intensity. The ESP8266 then transmits these data.

On the client side, popularly known as "Client Weather Station", an Arduino Nano receives the data. This received data is then displayed on an LCD screen. Additionally, the system interacts with Blynk cloud (for webpage) and the Blynk app (for mobile access) suggesting online data access and remote monitoring capabilities. The diagram, shown in Fig.1.1 illustrates a dual Arduino setup, with data flowing from the server-side ESP8266, to the client-side Arduino Nano for local display, and potentially to online platforms.

Figure 1.1: Block diagram of proposed hardware of smart weather monitoring system.

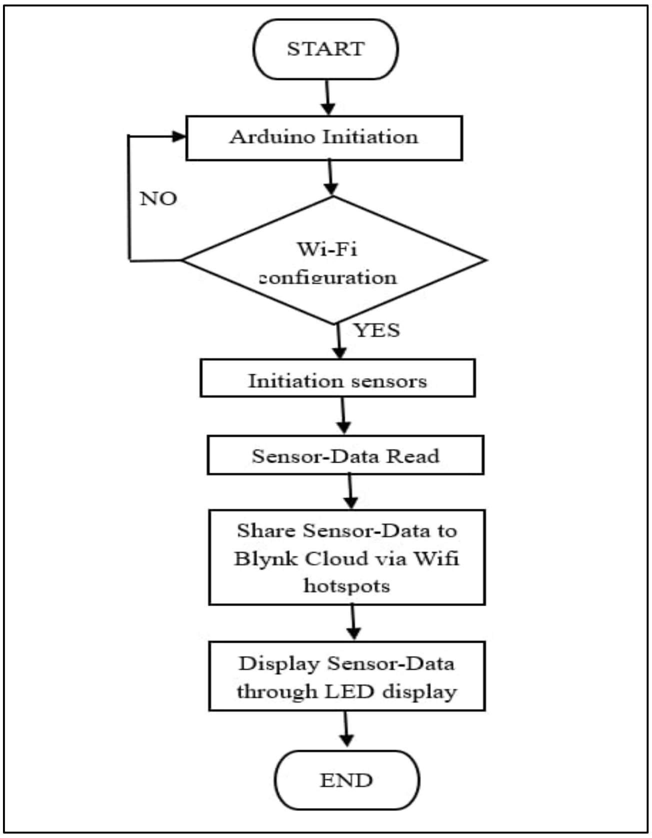


3.3. Flowchart of the Proposed Process Flow

This system employs an ESP8266 microcontroller to monitor environmental conditions using temperature, humidity, pressure, rain, and soil moisture sensors. The process begins with initializing the ESP8266 and establishing a Wi-Fi connection. Successful Wi-Fi connectivity is crucial for data transmission. Once connected, the system initializes each of the sensors: temperature, humidity, pressure, rain, and soil moisture.

After initialization, the ESP8266 starts reading data from these sensors through Arduino. The temperature sensor measures the ambient air temperature, providing crucial climate information. The humidity sensor gauges the moisture content in the air. The pressure sensor measures the atmospheric pressure. The rain sensor detects the presence and intensity of rainfall. Finally, the soil moisture sensor determines the water content in the soil. The step-by-step operations of the whole process is illustrated in the flowchart given in Fig. 1.2.

Figure 2.2: Flowchart of the Proposed Weather Monitoring System.



4. Hardware Details and Connectivity

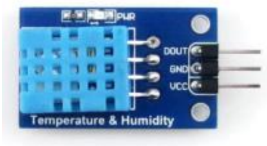
4.1. DHT11 Sensor Module

The DHT11 (Figure 2) 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 simple to use, but requires careful timing to grab data. You can get new data from it once every 2 seconds. It comes with a 4.7K or 10K resistor, which you will want to use as a pullup from the data pin to VCC.

Specifications

- Operating Voltage: 3.5V to 5.5V
- Temperature Range: 0°C to 100°C
- Humidity Range: 20% to 90%
- Resolution: Temperature and Humidity both are 16-bit
- Connection port: D3

Figure 2: DHT11 Sensor Module



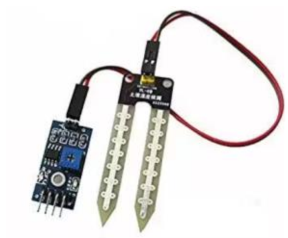
4.2. Soil Moisture Sensor

The Soil Moisture Sensor (Figure 3) is used to measure the water content (volumetric water content) of soil. This makes it ideal for performing experiments in courses such as soil science, agriculture science, environmental science, horticulture, botany, and biology.

Specifications

- Operating Voltage: 3.3V to 5V DC
- Working voltage: 5V
- Interface type: Analog
- Connectivity: AO (Analog Output), DO (Digital Output)

Figure 3: Soil Moisture Sensor Module



4.3. BMP180 Sensor Module

BMP180 (Figure 4) is one of sensors of BMP 180 series. They are all designed to measure Barometric Pressure **or** Atmospheric pressure. BMP180 is a high precision sensor designed for consumer applications. Barometric Pressure is nothing but weight of air applied on everything. The air has weight and wherever there is air its pressure is felt. BMP180 sensor senses that pressure and provides that information in digital output. Also, the temperature affects the pressure and so we need temperature compensated pressure reading. To compensate, the BM180 also has good temperature sensor.

Specifications

- Pressure range: 300 ~ 1100 hPa (altitude 9000 m ~ -500 m)
- Power supply voltage: 1.8 V ~ 5 V
- Connection port: SCL(D1), SDA(D2)

Figure 4: BMP 180 Sensor Module

4.4. Rain Sensor Module

The rain sensor module/board (Figure 5) is shown below. Basically, this board includes nickel coated lines and it works on the resistance principle. This sensor module permits to gauge moisture through analog output pins & it gives a digital output while moisture threshold surpasses. This module is similar to the LM393 IC because it includes the electronic module as well as a PCB. Here PCB is used to collect the raindrops. When the rain falls on the board, then it creates a parallel resistance path to calculate through the operational amplifier. This sensor is a resistive dipole, and based on the moisture only it shows the resistance. For example, it shows more resistance when it is dry and shows less resistance when it is wet.

Pin Configuration

The pin configuration of this sensor is shown below. This sensor includes four pins which include the following.

- Pin1 (VCC): It is a 5V DC pin
- Pin2 (GND): it is a GND (ground) pin
- Pin3 (DO): It is a low/ high output pin
- Pin4 (AO): It is an analog output pin

Figure 5: Rain Sensor Module

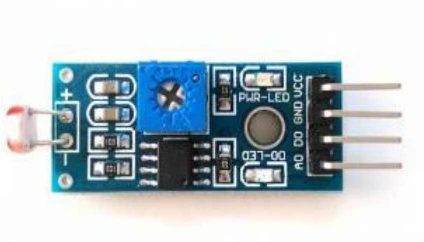
4.5. LDR Sensor Module

An LDR (Light Dependent Resistor) sensor module (Figure 6) with an LM393 comparator is used to detect light intensity. The LDR changes its resistance based on the amount of light falling on it — low resistance in bright light and high resistance in darkness. The LM393 comparator is used to compare the voltage across the LDR with a reference voltage set by a potentiometer. Based on this comparison, it gives a digital output (HIGH or LOW) to indicate whether the surrounding is bright or dark.

Specifications

- LM393-based design.
- It can detect ambient brightness and light intensity.
- Output Digital – 0V to 5V
- LEDs indicating output and power.
- Connection port: Analog and Digital Out
- Input Voltage (V): 3.3V to 5V

Figure 6: LDR Sensor Module



4.6. LCD Display Module

LCDs (Figure 7) are essential display units in electronics projects, offering a more versatile alternative to 7-segment displays. They utilize liquid crystal technology to present output values and messages. 16x2 LCD modules are particularly common, capable of displaying 32 ASCII characters across two lines. These displays are crucial for effective communication between humans and machines. Display units, regardless of size, operate on a fundamental principle. Understanding basic displays like 16x2 is important, even with the existence of more complex display technologies

Specifications of 16x2 LCD Display Module

- Operating Voltage: 4.7V to 5.3V
- Connection port: SCL(D1), SDA(D2)

Figure 7: LCD Display Module

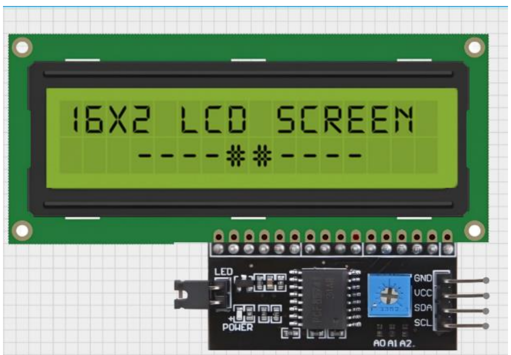
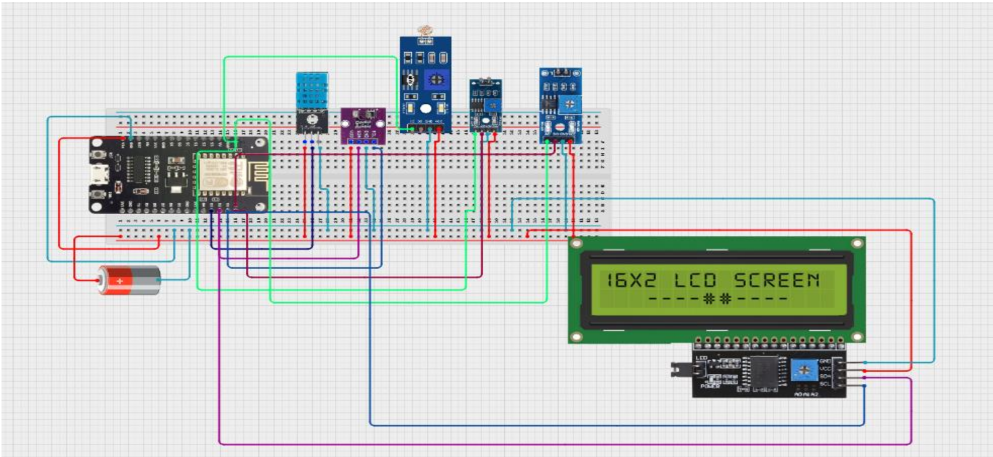


Figure 8: Interconnection of sensors output modules with microcontroller ESP8266



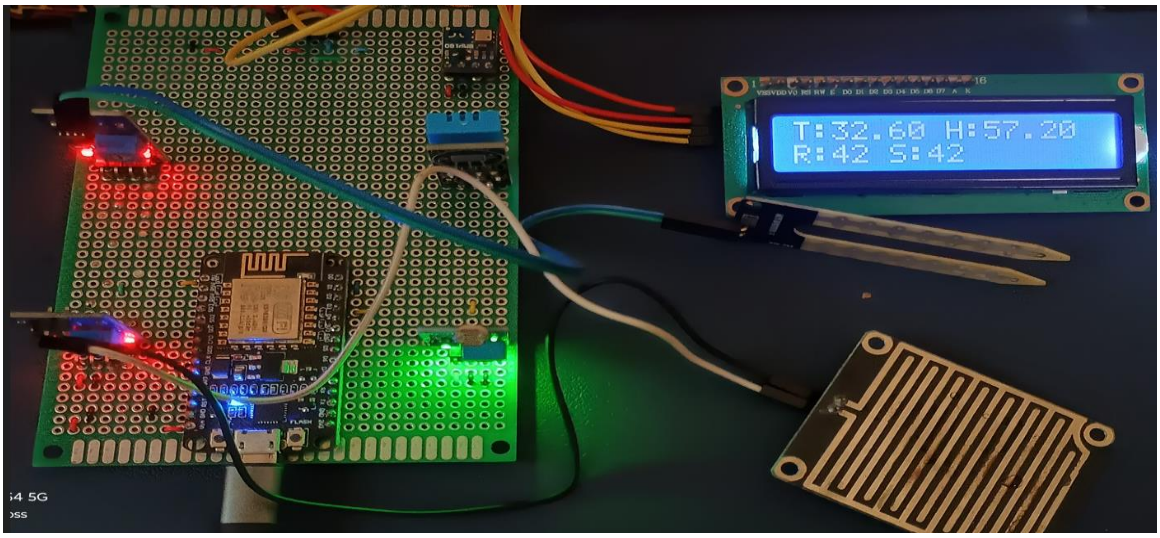
5. Hardware Implementation

The selection of hardware components plays a crucial role in the development of this project. The key hardware used in this project includes ESP8266 for wireless connectivity, DHT11 for temperature and humidity sensing, a soil sensor for soil moisture detection, BMP180 for atmospheric pressure measurement, a rain sensor for detecting precipitation, an LDR sensor for light intensity monitoring, and an LCD for data display. These components collectively contribute to the efficient operation of the system.

To set up the ESP8266 microcontroller, all sensors, including DHT11, rain sensor, and soil moisture sensor, are first tested using a multimeter. The sensors are then connected to the 5V, V_{CC}, and GND pins of the ESP8266. Digital sensors like DHT11, BMP180, and LDR sensor are linked to appropriate data pins, while analog sensors such as the soil moisture sensor and rain sensor are similarly connected. After verifying all connections, a B-type data cable is used to establish a connection. Once the circuit implementation is complete, the system is integrated with the Blynk platform via Arduino to transmit real-time sensor data for monitoring and analysis.

Figure 9: LCD display in run time condition with sample readings

Figure 9: Weather monitoring module in run time condition



6. Software Implementation

6.1. Creation of Blynk Mobile Application

Blynk is a comprehensive Internet of Things (IoT) platform that enables users to build mobile and web applications for controlling and monitoring hardware devices remotely. Designed with both beginners and professionals in mind, Blynk provides a user-friendly environment where developers can quickly prototype and deploy IoT solutions without extensive expertise in mobile app development or cloud infrastructure. It supports a wide range of hardware platforms, such as Arduino, ESP8266, ESP32, Raspberry Pi, and many others, making it a versatile tool for connecting everyday objects to the internet.

First, log in to the Blynk website and create the project. After developing the webpage and adding necessary widgets, download the Blynk app from the Play Store on an Android device. Log in to the app using the same email ID and password that is used on the website. The same project name will be displayed in the app that has been created on the website. In the app, select the gauge widget and complete the pin setup. Once everything is set correctly, the app will display real-time sensor data.

6.2. Webpage development with ThingSpeak and Database Management

ThingSpeak is an open-source Internet of Things (IoT) platform that enables users to collect, store, analyze, and visualize sensor data in real time. Developed by Math Works, the makers of MATLAB, ThingSpeak is designed to simplify the process of building IoT systems by providing a cloud-based environment where devices can send their data via standard web protocols, such as HTTP or MQTT. At the core of ThingSpeak is the concept of "channels," which are containers that store data streams from connected devices. Each channel can hold multiple fields of data, along with location information and status updates, making it easy to organize and retrieve sensor readings.

First, a web page is developed on the ThingSpeak website. After log in using institutetional email ID, a new channel is set with six fields. Each field is named after the type of data it would collect, such as, "Field1" for Temperature, "Field2" for Humidity, and so on. ThingSpeak allows free-of-cost log in and supports up to 8000 data entries per day using the API.

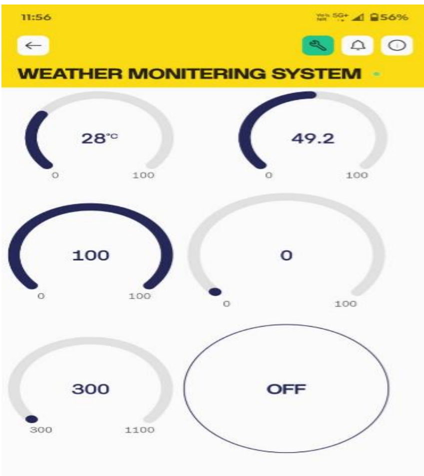
Once the channel was set up, a Read API Key is generated through ThingSpeak. This key was then copied into the Arduino code. Once the code is uploaded to Arduino, all sensors are connected and real time data are sent to ThingSpeak channel online. To retrieve and store the data from ThingSpeak, a database can be generated in a form of google sheets, which is implemented through a Python script using Google Apps Script or Python script with the Sheets API. Therefore, through Google sheets, connected with the ThingSpeak channel, sensor data is stored and maintained periodically, as defined by the user. As a result, the database goes on continuously updated on the ThingSpeak website and stored in the connected Google Sheet for further analysis and maintaining the records.

7. Results and Discussions

7.1. Mobile App with Blynk

In Blynk app development, users can display real-time sensor data. They monitor values like temperature, humidity, soil moisture, rainfall, and other environmental parameters. The connected sensors continuously update the data, allowing users to track conditions remotely and efficiently.

Figure 10: Weather monitoring Application user interface with Blynk.



7.2. Webpage User Interface using ThingSpeak

7.2.1. Temperature and Humidity Sensor Data

Through the ThingsSpeak website, the real-time temperature data is dispalyed as shown in Fig. 11.1. Fig. 11.1(a) displays the current temperature, whereas Fig. 11.1(b) displays the temperature variation over time. Similarly, the humidity has been shown in Fig. 11.2.

Figure 11.1: Weather monitoring Webpage-Temperature Value

(a) (b)

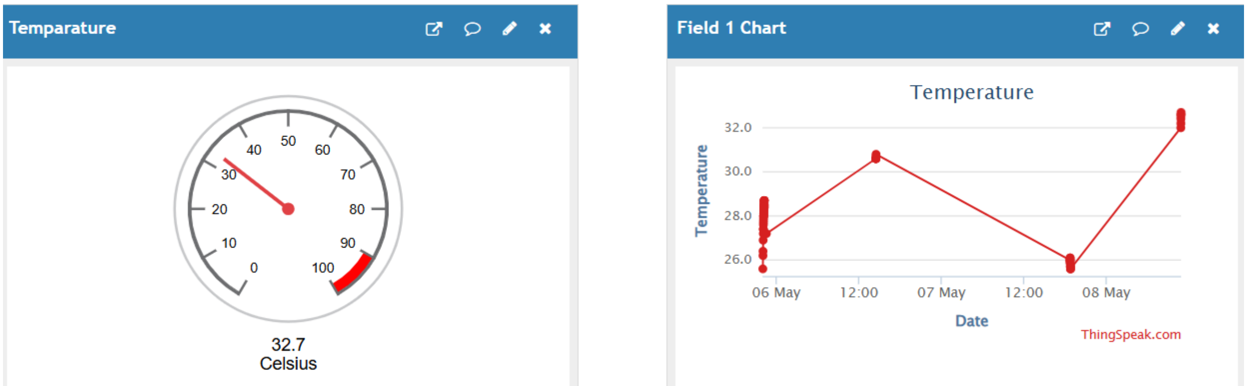
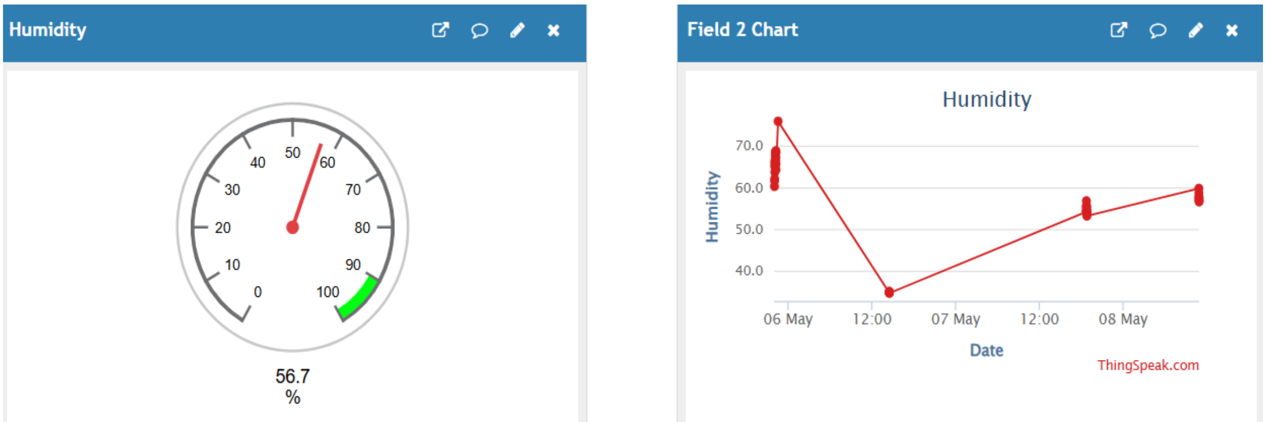


Figure 11.2: Weather monitoring Webpage-Humidity Value

(a) (b)



7.2.2. Soil Moisture and Rain Sensor Data

On the ThingsSpeak website, the real-time Soil Moisture data is displayed as shown in Fig 11.3. The measuring gauge shows the current moisture level, while the graph tracks the moisture level over the time. When the soil sensor rod is inserted into the soil, it detects the internal moisture content. The sensor readings are represented in percentages, for example, a value of 90% indicates the soil is wet, and a value of 30% indicates the soil is light wet and a value of 0% indicates the soil is fully dry.

Similarly, the real-time rain sensor data is displayed as shown in Fig. 11.4. When the rain sensor detects water, it measures the intensity of rainfall in millimeters (mm), providing an estimate of rainfall levels over time. The sensor data reflects ranges from 0 to 50 mm of rainfall, where 0 mm indicates no rainfall, 10–20 mm represents short rainfall, 20–35 mm indicates moderate rainfall, and values approaching 50 mm signifies heavy rainfall. This measurement reflects the volume of rainfall over a specific area and helps to categorize the intensity of rainfall effectively.

Figure 11.3: Weather monitoring Webpage-Soil Moisture Value

(a) (b)

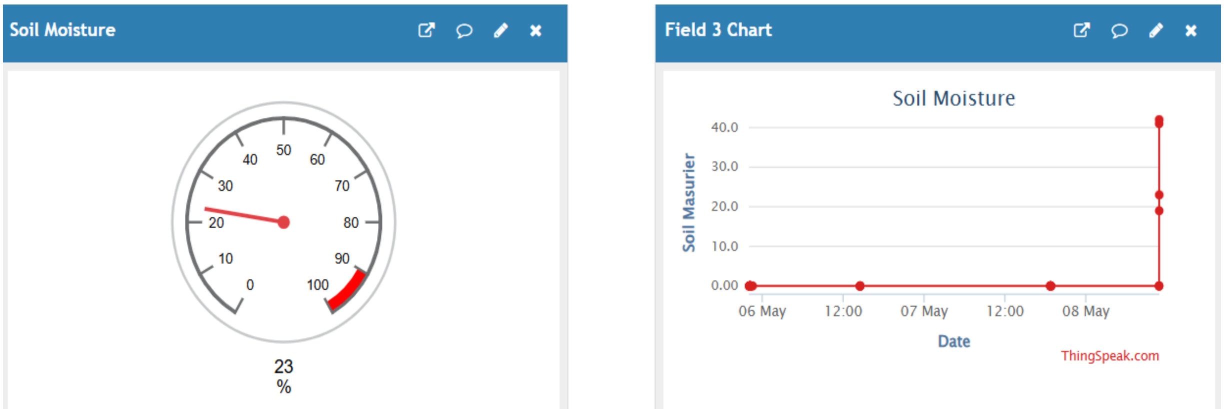
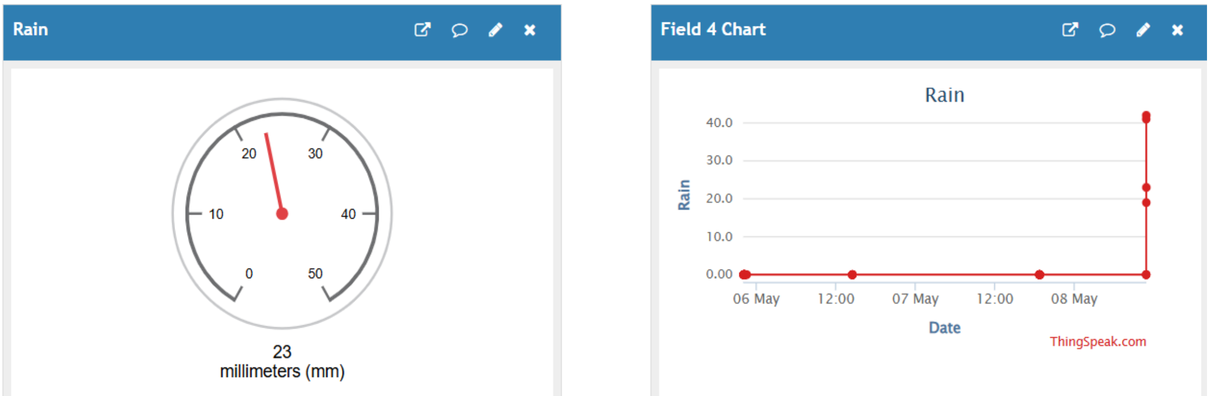


Figure 11.4: Weather monitoring Webpage-Rain fall Value

(a) (b)

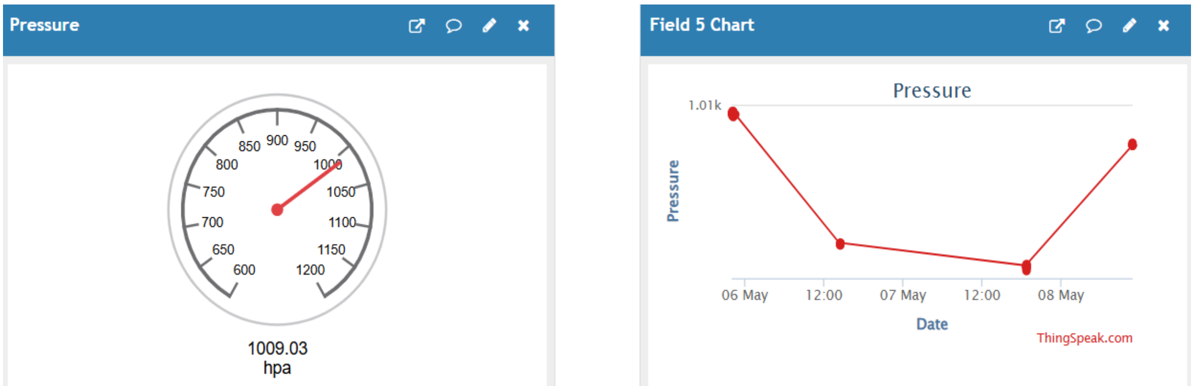


7.2.3. Pressure Sensor Data

On the ThingSpeak website, real-time pressure sensor data is displayed, as shown in Fig. 11.5. The current atmospheric pressure is presented with a graphical display that shows pressure variations over time. When the pressure sensor records data, it detects atmospheric pressure levels categorized as low, moderate, or high. The sensor data is represented in specific units, for example, a value of 1200 hPa indicates high pressure while a value of 600 hPa indicates low pressure Note that one hectopascal (hPa) is equal to one millibar (mbar).

Figure 11.5: Weather monitoring Webpage-Atmospheric Pressure Value

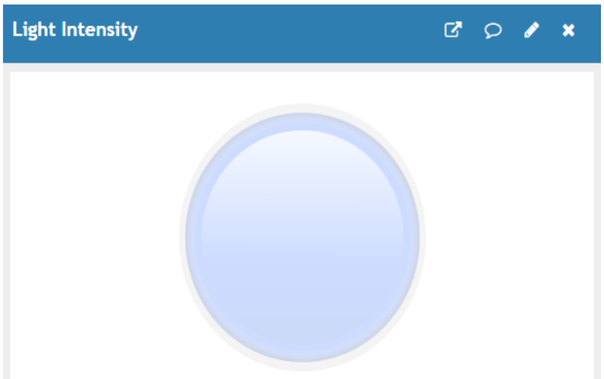
(a) (b)



7.2.4. Light Intensity Sensor Output

In the proposed system, an LDR (Light Dependent Resistor) sensor is used to detect ambient light intensity levels. This is particularly useful at no-light conditions at night or in situations where there is a sudden power cut. If the light turns off, it becomes difficult to see the sensors, such as the soil sensor rods or the rain sensor. This can lead to issues like not being able to place the sensors properly or provide the necessary voltage supply. Through the LDR sensor a lamp can be turn on depending upon the light intensity, ensuring that the sensors remain visible and functional even in low-light conditions.

Figure 11.6: Weather monitoring Webpage- Light ON/OFF Signal



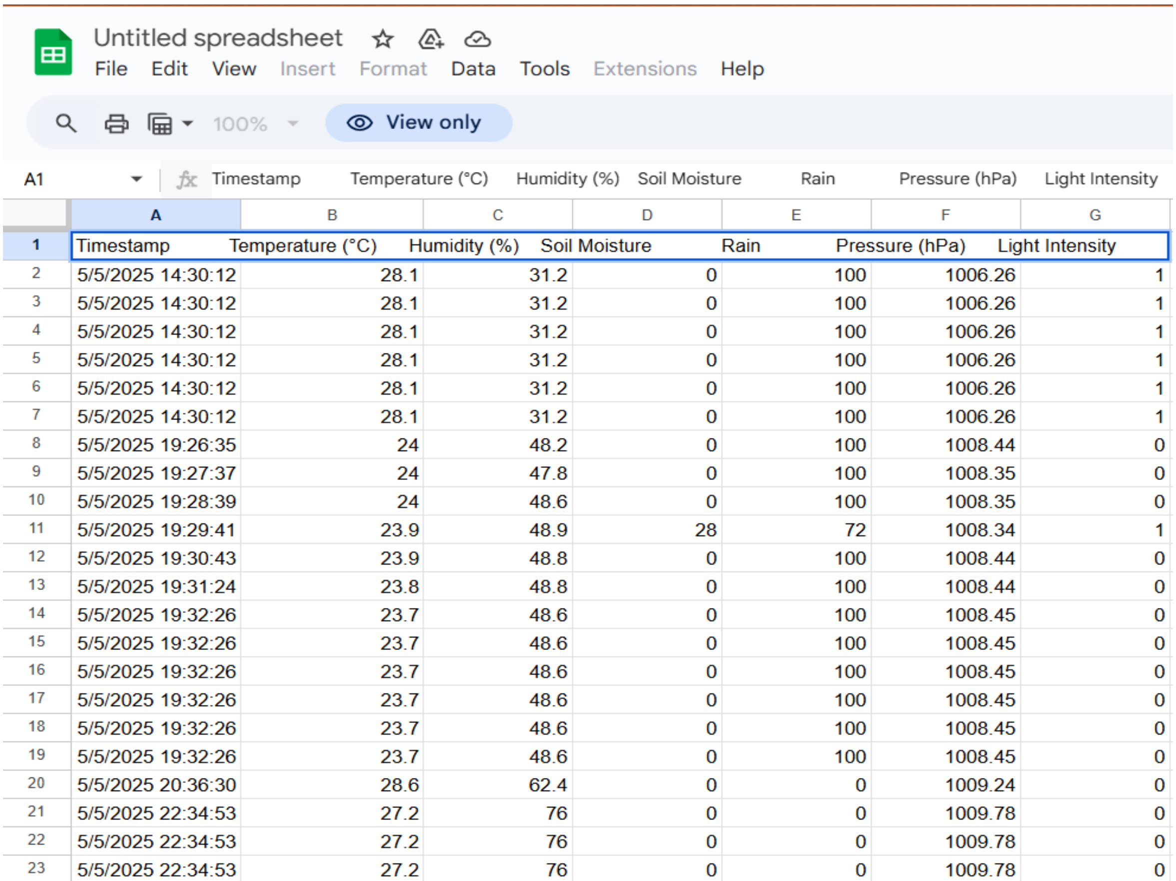
7.3. Database Management

The sensor data recorded in a Google Sheet using the ThingsSpeak Write API key is presented in Fig.13. The data, displayed in rows 2 to 7, was recorded on 14/05/2025 at 14:30 PM (Indian Standard Time). Later, additional data was collected between 19:26 PM and 19:32 PM on the same day, which is shown in rows 8 to 19. Following a 2-hour interval, more data was recorded and displayed in rows 20 and 21 at 22:34 PM and 22:36 PM respectively in Indian Standard Time. The link of Google sheet is given by, https://docs.google.com/spreadsheets/d/1OV9kbjsyOfko-HNrw_kMuTc0PCql1tlEyO8rwb7mhdE/edit?gid =0#gid=0.

The Table 1 provides a comparison of weather monitoring data collected from the proposed work and NASA's API across several dates. On 03/03/2025 at 12 PM (Indoor), the proposed work recorded 35°C temperature, 54% humidity, and 1001 millibar pressure, which closely matched with NASA's data of 36°C temperature, 58% humidity, and 1004 millibar of atmospheric pressure. On 06/03/2025 at 3 PM (Outdoor), both sources recorded similar results, confirming our system's consistency. Data recorded on 11/03/2025 at 12 PM (Outdoor), on 29/03/2025 at 6 PM (Indoor) and on 02/04/2025 at 1 PM (Outdoor), and on 17/04/2025 at 6.04 PM (Indoor) through our system have been tabulated and matched with the data collected from NASA. Our data were in good agreement with the NASA's record with some acceptable tolerance. Therefore, all the test results are marked "OK", demonstrating the accuracy, reliability and effectiveness of the proposed weather monitoring system.

A comparative analysis has been drawn among a few IoT based weather monitoring systems, published so far, and is given in Table 2. The comparison is based on six essential features: IoT architecture & sensors, data collection & transmission, cloud integration, real-time monitoring, mobile/web interface, and power management & cost efficiency. Five contemporary works in this field have been considered for comparison. A checkmark (✓) indicates that the feature is present in the respective work, while an "X" denotes its absence.

Figure 12: Sample Data Stored through Weather Monitoring System (Google Sheet)



	Timestamp	Temperature (°C)	Humidity (%)	Soil Moisture	Rain	Pressure (hPa)	Light Intensity
2	5/5/2025 14:30:12	28.1	31.2	0	100	1006.26	1
3	5/5/2025 14:30:12	28.1	31.2	0	100	1006.26	1
4	5/5/2025 14:30:12	28.1	31.2	0	100	1006.26	1
5	5/5/2025 14:30:12	28.1	31.2	0	100	1006.26	1
6	5/5/2025 14:30:12	28.1	31.2	0	100	1006.26	1
7	5/5/2025 14:30:12	28.1	31.2	0	100	1006.26	1
8	5/5/2025 19:26:35	24	48.2	0	100	1008.44	0
9	5/5/2025 19:27:37	24	47.8	0	100	1008.35	0
10	5/5/2025 19:28:39	24	48.6	0	100	1008.35	0
11	5/5/2025 19:29:41	23.9	48.9	28	72	1008.34	1
12	5/5/2025 19:30:43	23.9	48.8	0	100	1008.44	0
13	5/5/2025 19:31:24	23.8	48.8	0	100	1008.44	0
14	5/5/2025 19:32:26	23.7	48.6	0	100	1008.45	0
15	5/5/2025 19:32:26	23.7	48.6	0	100	1008.45	0
16	5/5/2025 19:32:26	23.7	48.6	0	100	1008.45	0
17	5/5/2025 19:32:26	23.7	48.6	0	100	1008.45	0
18	5/5/2025 19:32:26	23.7	48.6	0	100	1008.45	0
19	5/5/2025 19:32:26	23.7	48.6	0	100	1008.45	0
20	5/5/2025 20:36:30	28.6	62.4	0	0	1009.24	0
21	5/5/2025 22:34:53	27.2	76	0	0	1009.78	0
22	5/5/2025 22:34:53	27.2	76	0	0	1009.78	0
23	5/5/2025 22:34:53	27.2	76	0	0	1009.78	0

Our system includes all six features, making it a comprehensive solution. Other works show varying levels of completeness across the criteria. The goal of this comparison is to highlight the advantages and gaps in existing works. It also demonstrates the completeness of our implementation. This structured evaluation helps understanding where improvements can be made furthest. Overall, it supports the importance of integrated and cost-effective IoT based weather monitoring solutions.

Table 1: Weather monitoring Data of Our proposal work vs. API NASA

	Test Results_Proposed Work			**Test Results_API NASA**			
DATE	**TEMPARATURE IN DEGREE CELCIUS**	**HIMUDITY %**	**PRESSURE mBar**	**TEMPARATURE IN DEGREE CELCIUS**	**HIMUDITY %**	**PRESSURE mBar**	**TEST RESULT**
03/03/2025 12 PM INDOOR	35	54	1001	36	58	1004	OK
06/03/2025 OUTDOOR 3PM	33	20	1000	33.5	19	1001	OK
11/03/2025 IN FIELD 12PM	36	43	1004	35.5	42.7	1004	OK
29/03/2025 INDOOR 6PM	35	48	1003	36.4	49.9	1004	OK
02/04/2025 OUTDOOR 01PM	40.5	49	1005	42.2	51.1	1005	OK
17/04/2025 INDOOR 6.04PM	31.6	63	1005	31	63	1005	OK

Table 2: Comparative Analysis of Contemporary IoT based Weather Monitoring Systems

Feature	Ref [9]	Ref [10]	Ref [11]	Ref [13]	Ref [16]	Our Proposed Work
IoT Architecture & Sensors	✓	✓	✓	✓	✓	✓
Data Collection &	✓	✓	✓	✓	✓	✓

Transmission						
Cloud Integration	X	✓	X	✓	✓	✓
Real-Time Monitoring	✓	✓	✓	✓	✓	✓
Mobile App/Web Interface	X	✓	✓	X	✓	✓
Power Management & Cost Efficiency	✓	X	✓	✓	X	✓

8. Conclusions

The proposed IoT-based weather monitoring system offers an efficient, low-cost, and smart solution in real-time environment. The proposed system enables continuous data collection and analysis, providing valuable insights into weather conditions. The integration of cloud storage ensures that the collected data is accessible, shareable, and useful for future analysis, contributing to better decision-making in various sectors. This system has the potential to be implemented for weather monitoring in smart cities and industrial zones. By offering a scalable and cost-effective approach, it serves as a leading IoT solution in addressing environmental challenges through continuous and automated monitoring.

9. Future Scope

The proposed IoT-based weather monitoring system has strong potential for future development and global integration. By adding sensors for air pressure, and oxygen levels, CO_2 and other toxic gases the system can offer more detailed insights into environmental conditions and pollution leves. Integration with satellite data can further enhance its scope, enabling real-time weather monitoring on a global scale. The database, that is maintained regularly, can be utilized to create ML based predictive weather forecasting model in future. This technology holds critical value in sectors like aviation, navigation, and defense, where accurate weather data is vital for safety and planning. It can also support hospitals and research institutions in studying the relationship between weather and health, aiding in preventive care. Looking ahead, it is planned to develop a permanent, user-friendly web platform tailored specifically for farmers. This website will provide fast, accessible weather data to help farmers make informed decisions, improving agricultural productivity and contributing to sustainable rural development.

References:

1. Mircea Popa and Catalin Iapa "Embedded Weather Station with Remote Wireless Control", 19th Telecommunications forum TELFOR 2011 Serbia, Belgrade, November 22-24, 2011, 978-1-4577-1500- 6/11/$26.00, IEEE, 2011.
2. A Study on IoT Approach for Monitoring Water Quality Using MQTT Algorithm, Alfiya Abubaker1, Kavya C R2, Tilju Thomas3, Nikhil Joseph4, Shifana Begum5, 1,2,3,4 Final Year UG Students, Dept. of CSE, Srinivas School Of Engineering, Mangalore.
3. IoT Based Water Quality Monitoring System, Mourvika Shirode, Monika Adaling, Jyoti Biradar, Trupti Mate, Department of Electronics & Telecommunication Keystone School of Engineering, Pune, Maharashtra, India.
4. https://internetofthingsagenda.techtarget.com/definition/I nternet-of-Things-IoT.
5. Girija C Department of Electronics and Communication, NIEIT, Andreanna Grace Shires Department of Electronics and Communication, NIEIT, Mysuru Internet of Things (IoT) based Weather, International Journal of Engineering Research & Technology (IJERT).
6. Mobile APP & IoT Based Station Weather Station. Author: K. N. V.Satyanarayana, S. R. N. Reddy, K. N. V. Suresh Varma and P. Kanaka Raju.
7. Arduino Based Weather Monitoring System. Author: Karthik Krishnamurthi, Suraj Thapa, Lokesh Kothari, Arun Prakash.
8. Internet of Things (IoT) Based Weather Monitoring system, Bulipe Srinivas Rao1, Prof. Dr K. Srinivasa Rao2, Mr N. Ome3, international Journal of Advanced Research in Computer and Communication Engineering, ISO 3297:2007 Certified, Vol. 5, Issue 9, September 2016.
9. Mary Nsabagwaa, Maximus Byamukamab, Emmanuel Kondelaa, "Towards a robust and affordable Automatic Weather Station ", journal homepage: www.elsevier.com/locate/deveng.
10. Ravi Kishore Kodali and Snehashish Mandal "IoT Based Weather Station" 2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT) 978-1-5090- 5240-0/16/$31.00, IEEE, (2016).
11. Ravi Kishore Kodali and Archana Sahu "An IoT based Weather Information Prototype Using WeMos" 2016 2nd International Conference on Contemporary Computing and Informatics (ic3i), 978-1-5090-5256- 1/16/$31.00, IEEE, (2016).
12. Zi-Qi Huang, Ying-Chih Chen and Chih-Yu Wen, "Real-Time Weather Monitoring and Prediction Using City Buses and Machine Learning", Vols. 3 to 21 Published 10 September (2020).
13. Kavya Ladi, A V S N Manoj, G V N Deepak, "IOT Based Weather Reporting System to Find Dynamic Climatic Parameters", International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017).
14. P. Susmitha, G. Sowmyabala "Design and Implementation of Weather Monitoring and Controlling System", International Journal of Computer Applications (0975 – 8887) Volume 97– No.3, (July 2014).
15. Tanmay Parashar1, Shobhit Gahlot2, Akash Godbole3, Y.B. Thakare4 "Weather Monitoring System Using Wi-Fi", (IJSR) ISSN (Online): 2319-7064 Index Copernicus Value (2015): 78.96, 2015.
16. Nutant Sabharwal, Rajesh Kumar, Abhishek Thakur, Jitender Sharma "A low-cost zigbee based automatic wireless weather station with gui and web hosting facility" e-ISSN: 1694-2310 | p-ISSN: 1694-2426, Vol. 1, Spl. Issue 2 (May 2014).

17. M. Prasanna, M. Iyapparaja, M. Vinothkumar, B Ramamurthy, S.S. Manivannan," An Intelligent Weather Monitoring System using Internet of Things", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue4, November (2019).
18. Mircea Popa and Catalin Iapa "Embedded Weather Station with Remote Wireless Control", 19th Telecommunications forum TELFOR 2011 Serbia, Belgrade, November 22- 24, 2011, 978-1-4577-1500- 6/11/$26.00, IEEE, 2011.