International Journal of Safety and Security Engineering
Vol. 14, No. 1, February, 2024, pp. 249-258
Journal homepage: http://iieta.org/journals/ijsse

High-Resolution and Secure IoT-Based Weather Station Design

Farah Nazar Ibraheem*, Sura Nawfal Abdulrazzaq, Ina'am Fathi, Qutaiba Ali

Department of Computer Engineering, College of Engineering, University of Mosul, Mosul 41002, Iraq

Corresponding Author Email: farah_nazar80@uomosul.edu.iq

https://doi.org/10.18280/ijsse.140125

Received: 16 July 2023
Revised: 18 December 2023
Accepted: 16 January 2024
Available online: 29 February 2024

Keywords:
weather monitoring, Internet of Things (IoT), high-resolution data, environmental sensors, data security, meteorological data integrity

ABSTRACT

The pivotal role of weather information in domains such as workplace safety, economic activities, and natural disaster prevention is undeniable. This study introduces an advanced weather monitoring system designed to observe weather conditions at specific locations and make the data accessible globally. Utilizing Internet of Things (IoT) technology, this system connects a network of devices, including electronic gadgets and sensors, to the internet, thereby creating an interconnected web of information. The system focuses on monitoring environmental parameters such as temperature, relative humidity, pressure, and rainfall. It employs sensors to gather data, which is then transmitted to a web page and graphically represented as statistical information. The data, accessible from any location via the internet, enhances the resolution and accuracy of weather monitoring. The implementation of high-precision sensors in this system ensures the delivery of detailed and accurate meteorological data, facilitating a comprehensive understanding of local weather conditions. Additionally, the system's design allows for the easy deployment of multiple weather stations, broadening the scope of weather monitoring and providing valuable insights into microclimates and localized weather patterns. A significant feature of this system is the incorporation of a robust security framework, safeguarding the integrity and confidentiality of meteorological data. This aspect is crucial for ensuring the reliability of the data for various applications, including disaster preparedness and response. Timely evacuation warnings, efficient resource allocation, and accurate damage assessment are facilitated by the real-time weather monitoring provided by this system.

1. INTRODUCTION

The importance of weather information extends across various fields, serving individuals, large organizations, and companies. This indispensable data can be gathered by weather station systems within specific geographic areas. An automated weather station, equipped with sensors, eliminates the need for human involvement in collecting readings on atmospheric factors such as humidity, pressure, temperature, and wind. These readings provide a summary or forecast of the weather. The weather data is useful in many applications from different domains, like agriculture, entertainment, industries, military operations, transportation, emergency management, etc. [1-3].

To ensure healthy crop development and maintain a safe working environment in these diverse sectors, it is necessary to continuously monitor the weather conditions. The evolution of technology has transformed the once-challenging task of reading environmental indicators. Utilizing miniature electronic devices – sensors – enables a faster and more reliable monitoring system, contributing to the development of "smart weather stations" that operate without the need of a human operator [4]. While traditional weather stations often used wired sensors, modern station designs integrate wireless interfaces for increased efficiency and flexibility. This article aims to contribute to the field by introducing a secure weather station model by developing a weather station system that provides accurate real-time weather data and incorporates security measures – a system that ensures continuous reporting of weather statistics to enhance information reliability. The research methods involve sensor integration for tracking various weather parameters and utilizing wireless communication technologies like the ESP8266 WiFi Module for secure data transmission to a web server.

The accuracy and reliability of weather monitoring are paramount. Falsified or forged weather information can lead to incorrect forecasting, especially during dangerous meteorological phenomena. This could cause threats to national security and substantial economic losses. To mitigate these risks, a robust security model is needed to protect and safeguard the integrity of collected weather data.

This article is organized into seven sections, as follows: The "Related Works" section discusses related articles in this field; the research methodology is presented in Section 3, Sections 4 and 5 show the methods, hardware, and software implementation; Section 6 presents the obtained results; and the seventh section explains the proposed security system to protect the information. Finally, the article concludes with a summary of findings and insights.

2. RELATED WORKS

There exists a body of previous works focusing on the integration of weather station sensors with IoT platforms, employing diverse hardware and methodologies. Some initiatives involved the hardware implementation or prototyping of weather station systems, while others proposed simulation models. The use of Arduino boards for collecting weather data, such as temperature, wind speed, and rainfall, has been a common thread among these works. The collected data can be directly displayed on a small LCD/OLED display [5] or transmitted wirelessly to a computer for further analysis [3, 6].

In 2018, the combination of the ESP8266 Wi-Fi Module and Arduino enabled wireless transmission [6]. In a military context, researchers employed the X-bee module for wireless communication, creating a weather monitoring system for transporting military equipment with precision [3]. The gathered data can be stored on a cloud like ThinkSpeak, where the uploaded data is processed using MATLAB and the sensor values can be displayed with respect to time [4, 7]. In the study of Baraki et al. [7], different locations are taken into account, like Bangalore, Dharwad, Belgaum, and Gulbarga, for multiple types of meteorological analysis factors, and they are posted on a cloud, enabling others to log into them.

Different communication methods were explored, including serial communication with LabVIEW [8, 9], the Wemos D1 board is also used to acquire data from the sensors where it is implemented on the Arduino platform [5]. Wireless connections with the PIC16F887 microcontroller and Xbee Pro radios [10] and ESP8266-based Wi-Fi modules (NodeMCU) for real-time weather data tracking and global accessibility [11]. Mobile applications for real-time weather monitoring were implemented on Android devices [12]. A weather station that is particularly beneficial for smart city difficulties was given by Malik et al. [13].

The IoT underpins this weather station. It has weather sensors that can take readings from all over the globe and upload them to the cloud in real time. They made use of various environmental sensors and a Raspberry Pi. Sensors continuously monitor the weather and upload the data to a server in real time through WiFi [13-15].

Despite the significant contributions of these works, none of them addressed the crucial aspect of security in weather station design: protecting against unauthorized access, data breaches, or tampering. Few resources mentioned simple data encryption [16], while others presented theoretical studies without practical implementation or acquiring real weather data [17].

In contrast, our work introduces a secure weather station model, integrating sensors for real-time weather tracking and continuous data reporting. The ESP8266 WiFi Module interprets the collected data and sends it to a web server wirelessly. The data is updated on a platform like ThingSpeak, and the technology allows users to set alarms for specific weather conditions, enhancing the usability of the system. To facilitate a clearer comparison, Table 1 provides a summary of the main contributions and applications of various research works in this domain, highlighting the presence or absence of a security model.

Table 1. Comparison of contributions and platforms used in many researches

Related Works	Platform	Main Contribution/ Application	Security Model
[2]	interpolated weather data from existing meteorological stations	Agriculture	X
[3]	Arduino	Routing the military equipment	X
[4, 6, 7]	Arduino	Prototype model	X
[5]	Wemos D1 board	Cost effective design/ OLED display	X
[8]	Arduino/LabVIEW	GUI monitoring	X
[9]	DAQ/LabVIEW	GUI monitoring	X
[10]	PIC16F887 /ZIgbee promodel	Portable Weather station	X
[11]	ESP8266	Mobile application	X
[12]	AWS-Davis Vantage Pro 2 weather	Mobile application	X
[13, 14]	Raspberry PI	Smart city / Artificial Intelligent	X
[15]	Raspberry PI	Low cost design	X
[16]	Arduino	(simple encryption only)	√
[17]	Theoretical Study	Theoretical Study	√

3. RESEARCH METHODOLOGY

Real Time Weather Analytics" problem can be solved mainly using data analysis tool such as ThingSpeak. For analyzing various weather parameters, Data is collected continuously at a fixed sampling rate of 20 seconds by the Wi-Fi-enabled IoT sensors., Each sensor transmits its readings via the NodeMCU microcontroller, which acts as a central processing unit, which is part of the actual system. The environmental parameters that need to be calculated are the system's inputs i.e, humidity, temperature, rain, pressure, when the microcontroller receives a signal from a sensor, it processes and evaluates the signal to determine whether or not it corresponds to the expected value for the parameter. The MCU's transmitter is then used to transfer the sensor data to the Wi-Fi module. Connected Wi-Fi modules provide information to the ThingSpeak cloud service, On the ThingSpeak website, analysis and plots were displayed. ThingSpeak provides various tools and functionalities for visualizing and analyzing the received data such as Real-time charts and graphs which Plot sensor data in various formats like line graphs, bar charts, and scatterplots, data tables which view the data in a tabular format for detailed inspection and analysis, you can also create custom visualizations using the ThingSpeak Charts API. This allows for advanced analysis and presentation of the data. ThingSpeak include Mathematical Calculations and Analytics such as Built-in functions for performing mathematical operations on the data. These include calculations for averages, maximum and minimum values, standard deviations, and other statistical measures, it also integrates with MATLAB, allowing users to write and execute MATLAB code for more complex data analysis and modeling. Figure 1 offers the proposed system.

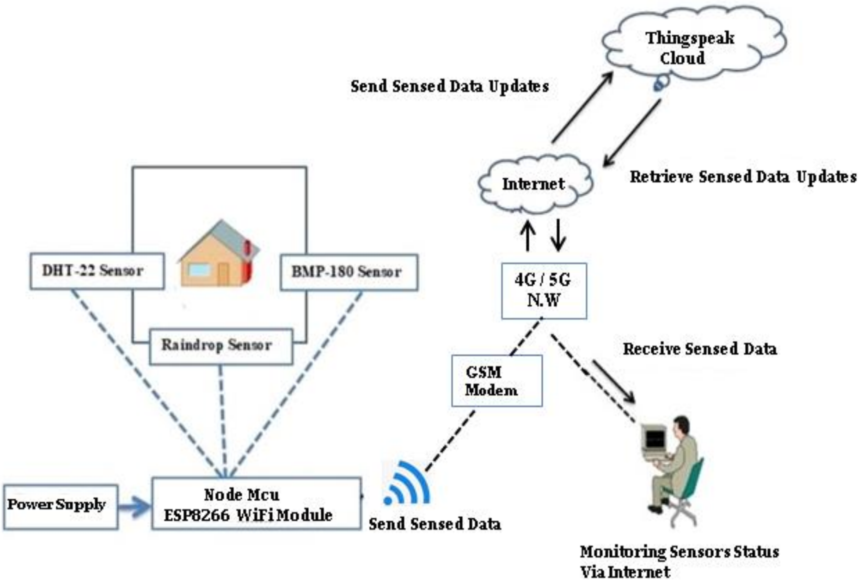

Figure 1. Framework of the proposed system

4. METHODS AND MATERIALS

4.1 Implementation setup

The hardware components required in this research:
1- ESP8266 based WiFi module NodeMCU
2- Temperature and Humidity Sensor (DHT22)
3- Barometric Pressure Sensor (BMP180)
4- Raindrop Module (MH-RD)
5- Mobile phone to receive email and SMS

4.2 NodeMCU setup

The NodeMCU is programmed with the Arduino IDE to read sensor data, process it, and transmit it to the ThingSpeak cloud platform. The code includes functions for: Sensor initialization and data acquisition, Data processing and conversion, Communication with the ThingSpeak platform via WiFi, and Error handling and logging. Thus the data reading and collection can be briefly explaind in some steps: Firstly each sensor continuously measures its assigned environmental parameter, then The NodeMCU collects the sensor readings at the specified sampling rate and performs any necessary calculations or conversions on the collected data., The NodeMCU sends the processed data to the ThingSpeak cloud platform via WiFi, Finally The data is stored securely in the designated ThingSpeak channel for further analysis and visualization.

4.3 ThingSpeak IOT platform

ThingSpeak is an IOT analytics platform service that allows you to aggregate, visualize, and analyze live data streams in the cloud. You can send data to ThingSpeak™ from your devices, create instant visualizations of live data, and send alerts using web services like Twitter® and Twilio®. With MATLAB® analytics inside ThingSpeak, you can write and execute MATLAB code to perform preprocessing, visualizations, and analyses. ThingSpeak enables engineers and scientists to prototype and build IoT systems without setting up servers or developing web software.

To Setup Thingspeak sign up for a ThingSpeak account. Establish a novel channel, our information comes from a particular channel. where data can be stored and retrieved, there is a cap of 8 fields per channel. Thus, 8 different types of data can be stored on a channel. Access to your channel can be controlled by distributing its API (Application Programming Interface) keys. There are two methods to access channel; Modify logging of channels and data: in this mode, the API Write Key will be utilised to gain access. In this setting, the API Read Key is utilised to retrieve information, in our system The NodeMCU transmits the data to the ThingSpeak channel at a fixed sampling rate of 20 seconds, this process involves: connecting the NodeMCU to the WiFi network, establishing communication with the ThingSpeak server using API keys, formatting the sensor data into a compatible format and sending the data to the specified channel. At the other hand, ThingSpeak offers various tools for visualizing and analyzing the received data, these include: Real-time charts and graphs, data tables and exports, Mathematical calculations and analytics. Figure 2 offers the flowchart of ThingSpeak setup.

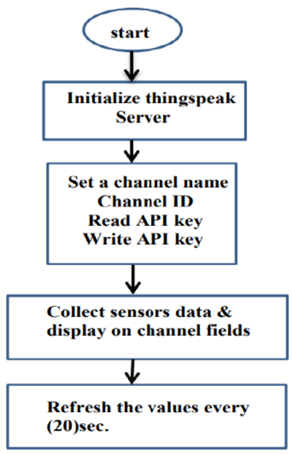

Figure 2. Flowchart of ThingSpeak setup

4.4 Initial testing and validation

Before deploying the weather station system, initial testing and validation were conducted to ensure its accuracy and reliability. These tests included:

- Sensor Calibration: Each sensor was calibrated against a reference device to ensure accurate readings.
- Data Acquisition Verification: The data acquisition process was tested to check for missing or corrupted data.
- Communication Testing: The communication between the NodeMCU and the ThingSpeak platform was tested to verify data transmission and reception.
- Data Visualization Verification: The data visualization tools in ThingSpeak were tested to ensure proper data display and analysis.
- Alert and Notification Testing: The alert and notification system was tested to confirm timely alerts for specific weather conditions.

5. DESIGN AND IMPLEMENTATION

This section deals with the design and implementation issues of the intended system.

5.1 System functionality

The operation of the entire system after integrating all peripheral devices and software is referred to as functionality of system. The process consists of three stages; data acquisition stage in which the environmental parameters (temperature, humidity, pressure, rain) are continuously measured by the system sensors, the microcontroller (NodeMCU) collects sensor readings at a fixed sampling rate (20 seconds) then converts the collected data to a compatible format. Data Transmission is the second stage in our system when the NodeMCU establishes communication with ThingSpeak server using API keys then sends the data to the designated ThingSpeak channel via WiFi. The last stage is the analysis and visualization of the sent data when ThingSpeak stores the received data securely in the designated channel, the users can access and visualize the data through real-time charts, graphs, and tables, also analyzing the sensor data and managing the gadgets at last in accordance with the data obtained. Figure 3 offers the Hardware Connection between NodeMCU and the system sensors. While Figure 4 offers weather Station Operation Flow.

5.2 Control and alarm system

Controlling tasks for instantaneous action based on sensor output can also be carried out by the system. The microcontroller continuously reads and monitors sensor data. Threshold values which determined by the humidity and temperature sensor, trigger control actions such as activating LEDs, fans, and buzzers, as example when the temperature exceed 40 degree in our system, there is some actions like turning on a buzzer or switch on a led. Simultaneously, a mobile phone receives notifications and emails when sensor readings surpass predefined thresholds, such as when the humidity exceed 50% and so on. A detailed depiction of the controlling system is presented in Figure 5.

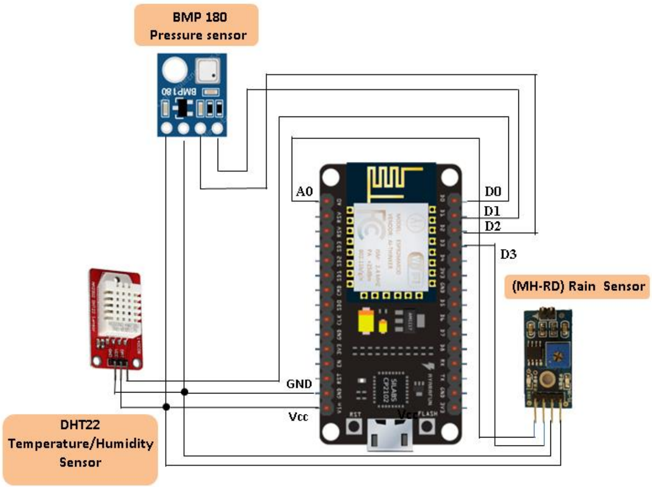

Figure 3. Hardware connection between NodeMCU and the weather sensors

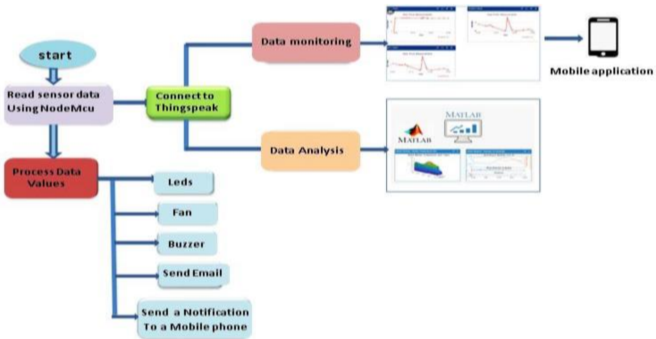

Figure 4. Weather station operation flow

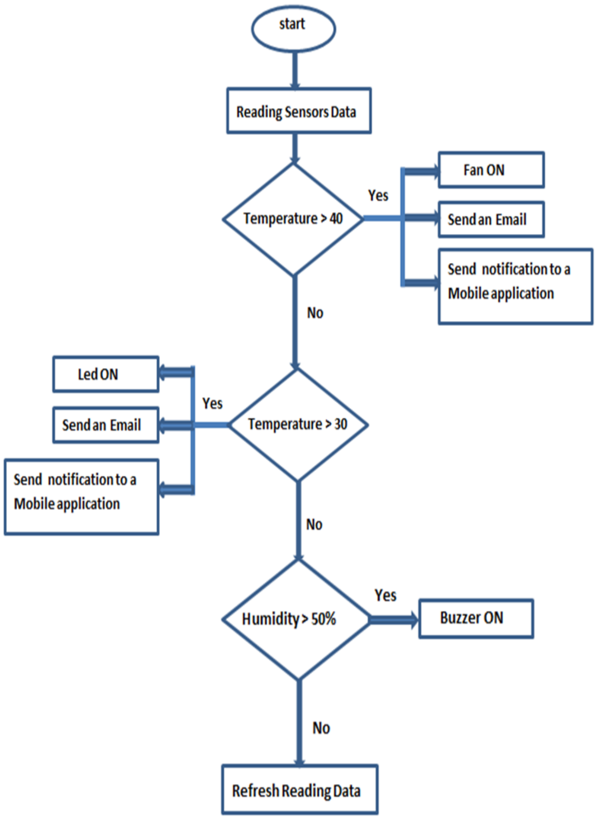

Figure 5. Flowchart of controlling system

5.3 System planning

This project was implemented at the University of Mosul, with eight meteorological stations strategically distributed across the university area. where weather information is collected from each point in real time, monitored and analyzed, and all information is sent to a central database of ThingSpeak, Figure 6 shows the distribution of these points, the red points represent the locations for the stations. These locations were chosen to provide accurate and granular weather data coverage, which depends on some factors like the Size and layout of the university campus, desired level of weather data granularity, accessibility and ease of maintenance for each station and minimizing interference between stations.

Figure 6. Weather stations distribution map

6. RESULTS AND DISCUSSION

6.1 Visualizations in ThingSpeak

The latest version of MATLAB, R2016a, has been released by Mathworks, and it presents a fresh chance to provide stability for Internet of Things initiatives and as of right now, ThingSpeak is the only IoT web service that provides open source data analysis on the MATLAB platform. Users of Thing Speak can now access MATLAB for data analysis and visualization without having to purchase a licence from Mathworks. It is necessary to read all of the weather information from ThingSpeak to MATLAB in order to perform analysis and visualization.

ThingSpeak offers several types of visualizations to cater to different data types and analysis needs, like line graph, bar charts, scatterplots, static visualization as Gauge charts, Maps, etc ThingSpeak also allows adding formulas and expressions to customize data analysis within visualizations.

The ThingSpeak Read function is used to retrieve information from our channel in all fields, which is then broken down into its component variables, in order to properly obtain the data from the cloud, we must write MATLAB code and with the aid of API key and channel ID, we are able to access and get data from the cloud. The Figure 7 draws attention to the distinct Read API key and Thing-speak channel ID that are used in our work.

Every graph includes information about the environmental parameter detected by the respective sensor. A timestamp indicating the precise time of data reception is shown alongside each set of depicted data.

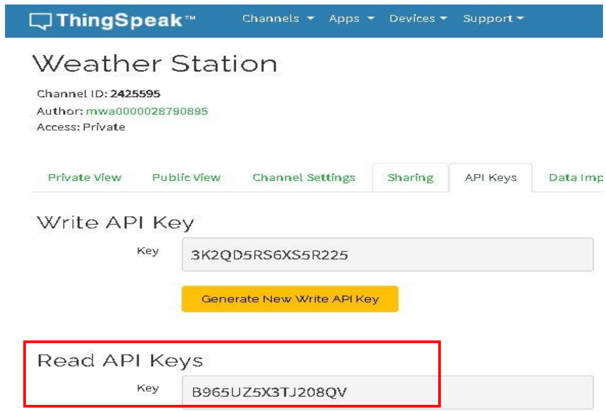

Figure 7. Channel ID and read API keys

The Figure 8 displays the ThingSpeak cloud's graphical output results. here we use the line graph and Gauge charts, the line graph shows data points connected by lines over time, our data points represents the temperature values over time, which were measured by temperature sensor and sent by the NodeMCU to the ThingSpeak platform, while the gauge chart which represents another valuable visualization tool in ThingSpeak, here we use Circular Gauge, it has more benefits like it is easy to understand, Ideal for real-time monitoring of key measurements, and it Highlight threshold breaches or critical conditions with color changes or alarms. Figure 9 displays the humidity values in line graph and gauge chart.

The channel location chart offered by Google Maps also displays the location where the data was acquired, so Figure 10 shows the location for our station. Figures 11 and 12 show the pressure level and rain values respectively, they display a real time data values over a range of time.

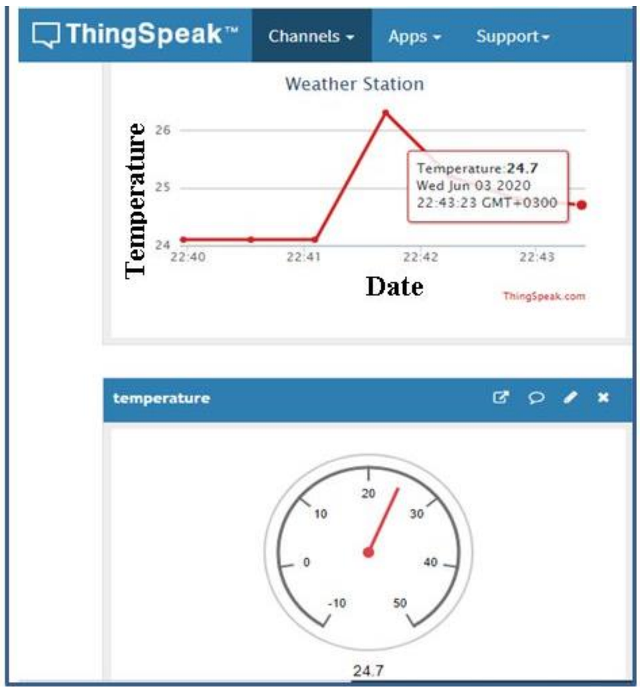

Figure 8. ThingSpeak temperature output results

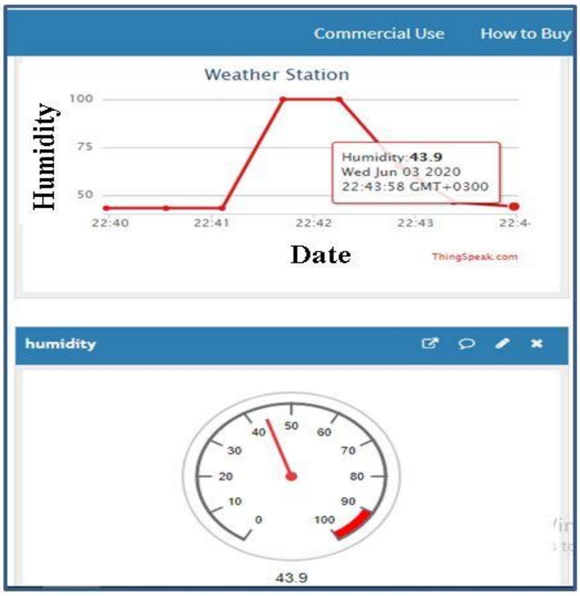

Figure 9. ThingSpeak humidity output results

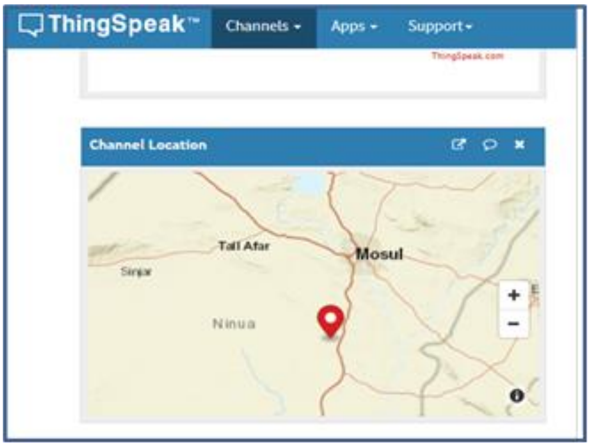

Figure 10. Channel location

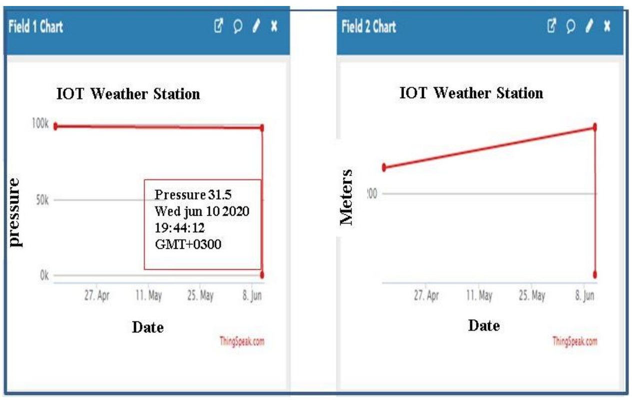

Figure 11. ThingSpeak pressure & altitude output results

6.2 Data analysis

Combining MATLAB with ThingSpeak opens up a powerful toolkit for advanced data analysis of IoT sensor readings, we can Perform advanced analysis with MATLAB's extensive library of mathematical and statistical functions, calculate statistics like mean, median, standard deviation, and correlations, apply filtering techniques like moving averages, Fourier transforms, or signal processing tools. Create sophisticated visualizations like bar charts, histograms, scatterplots, maps, and animations, finally using MATLAB toolboxes like Signal Processing Toolbox or Statistics and Machine Learning Toolbox for specialized analysis.

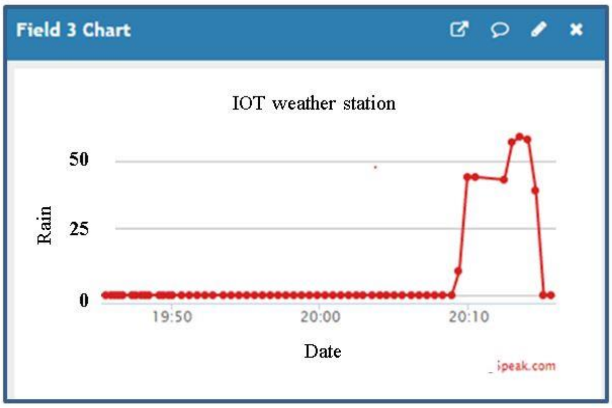

Figure 12. Rain sensor output results

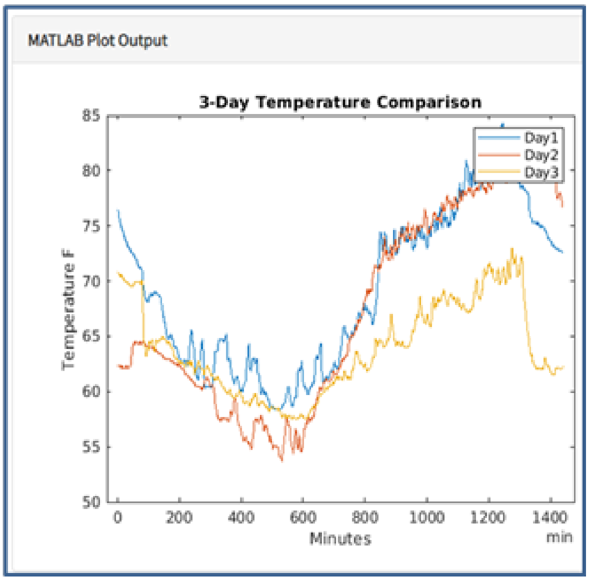

Figure 13. Histogram of temperature variation

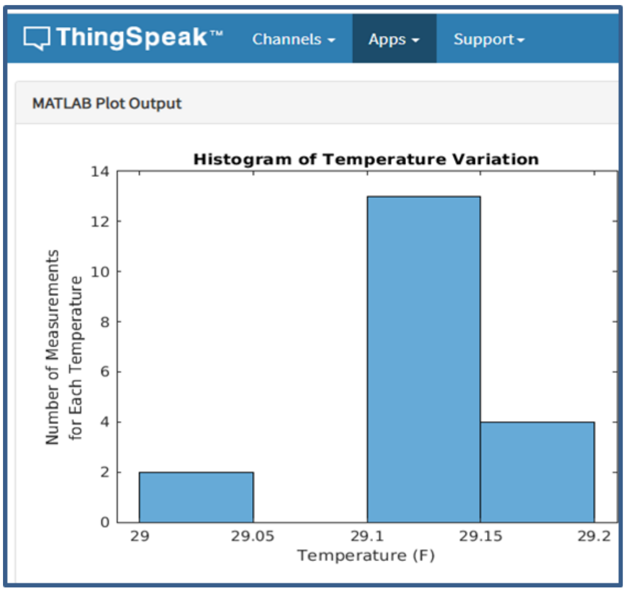

Figure 14. Temperature comparison of 3 days

The following step in our work is to use MATLAB R2016a to examine the sensed data, as a result, we need to write MATLAB code that can access the cloud storage without issue and to read and gather data from the cloud, we use API key and channel ID. Both the ThingSpeak Read API key and channel ID used in this paper are displayed in Figure 7. As example for data anaysis in our work we used a MATLAB function to read the temperature values and calculate the Histogram of Temperature Variation as shown in Figure 13, while Figure 14 displays the temperature comparison of 3 days.

6.3 Integration with Blynk and ThingView

the Blynk application serves as a mobile interface for interacting with internet-connected projects, it acts as a bridge between our Arduino, NodeMCU, or other development board and our smartphone or tablet, enabling us to View sensor readings in real-time, receive notifications when thresholds are exceeded, and Control devices remotely. we can read the sensor readings whenever and wherever we are, the values are examined, and once the threshold limit is exceeded, an email or other notification is sent.

ThingView is also a mobile application designed to specifically visualize data collected through the ThingSpeak IoT platform.As long as you know the channel ID, you may quickly and easily visualize your ThingSpeak channels. By entering the channel ID number, we may also display the charts for any other public channels worldwide

6.4 Key contributions and potential applications

This section describes the overall significance of our system:

- Enhanced environmental monitoring: our model offers a valuable tool for monitoring weather conditions in various environments, enabling better-informed decision-making in agriculture, disaster preparedness, and climate research.
- Improved data analysis: The integration of ThingSpeak and MATLAB opens doors to advanced data analysis techniques, potentially leading to breakthroughs in weather forecasting, resource management, and environmental modeling.
- Increased accessibility and awareness: Mobile app integration makes environmental data readily available to a wider audience, raising awareness and promoting responsible environmental practices.

7. SECURITY ISSUES OF THE SUGGESTED WEATHER STATION

7.1 Security threats

There are some of the most common security threats faced by IoT-based weather stations, see Table 2:

- Unauthorized Access: This security threat could give them the opportunity to interfere with the device, steal data or even seize control of the entire network. A hacker could break into an IoT-based weather station by exploiting a weakness in the device firmware, or by using weak passwords.
- Malware and Viruses: Malware and viruses could be introduced to the network either by infecting IoT-based weather stations or rendering them inoperable. For instance, malicious software might be used to delete or damage data. Or it could control the device without user awareness. The virus could spread to other devices on the network, inflicting mass damage.
- Data Breaches: Data generated by IoT-based weather stations can be sensitive, and any data breach would result in a violation of confidentiality and integrity. Such a data theft could, for instance, expose users 'personal information or allow attackers to track changes in temperature.
- Denial of Service (DoS) Attacks: A DoS attack can saturate the network, in which case any weather station on it will fail or shut down altogether (the weather station is unable to send or receive data). This could be disastrous for weather forecasting and warning systems.
- Lack of Security Updates: Weather Stations may not receive regular security updates, leaving them open to known vulnerabilities. Likewise, a security weakness in the device's firmware could permit attackers to access it illegally.
- Physical Tampering: Hackers can get physical access to the device and control it or steal data. For instance, an intruder might open up the weather station and plant a backdoor into it which would enable remote control over the device.
- Insider Threats: Security breaches are caused by mishandling of sensitive data or improper configuration of the network either purposefully or abusely on the part of authorized users (a person could misplace their weather station password in the open).

Table 2. Security analysis

Threat Type	Countermeasure	Justification
Unauthorized access	Authentication / Access control	Robust authentication mechanisms, such as HTTPS and authentication keys, prevent Unauthorized Access Attacks.
Malware and viruses	Firmware updates / Physical security	Regular over-the-air (OTA) updates address vulnerabilities, reducing the risk of Malware Infection Attacks. Physical security measures prevent unauthorized physical access.
Data breaches	Data encryption	Encryption techniques like SSL/TLS or AES protect against Data Breach Attacks.
DoS attacks	Network security / Monitoring	Securing the network with firewalls and intrusion detection systems mitigates DoS Attacks. Continuous monitoring identifies and addresses various attacks.
Lack of security updates	Firmware updates	Regular over-the-air (OTA) updates ensure that the weather station remains up-to-date, addressing any identified security vulnerabilities promptly.
Physical tampering	Physical security	Physical security measures prevent Physical Tampering Attacks.
Insider threats	Network security / Monitoring	Implementing access controls, including user accounts and password protection, mitigates Insider Threats. Continuous monitoring identifies and addresses insider threats.

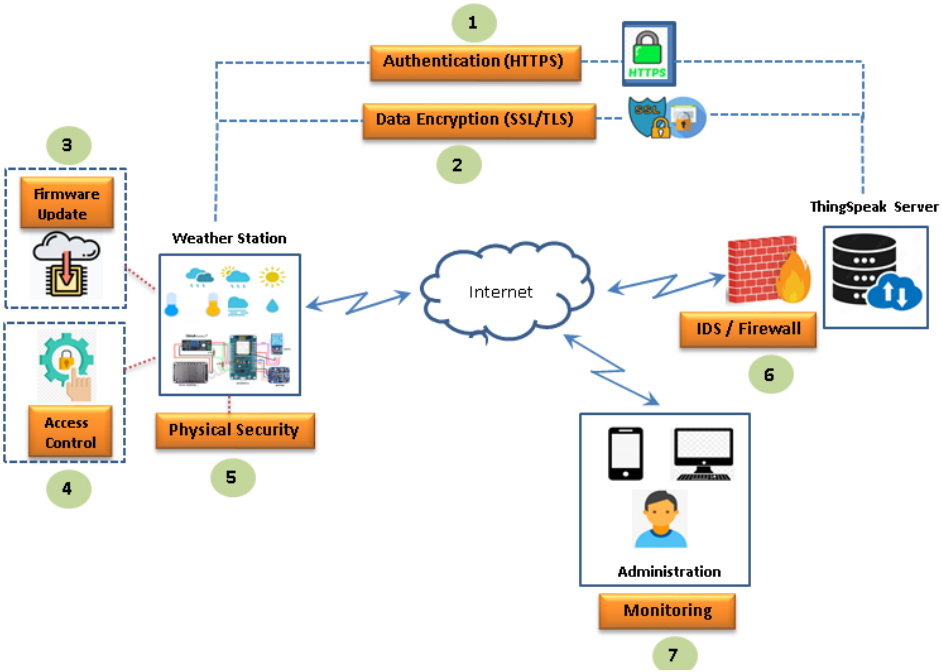

Figure 15. Security solutions

7.2 Possible solutions

An IoT-based weather station security model should consider the following aspects:

- Authentication: The weather station must have a secure way to authenticate itself with its destination server or cloud platform. Via secure communications protocols such as HTTPS, or by using keyed or certificated authentication.
- Data Encryption: The data from the weather station must be sent to a server or cloud platform through encryption. This can be done using SSL/TLS or AES encryption.
- Access Control: And it is hoped that the weather station can set up check points to prevent unauthorized entry of data or operation on the weather station. Such features include user accounts, and password protection.
- Firmware Updates: The firmware running the weather station must be periodically updated to patch any security holes or viruses that may have developed. This is through an OTA update mechanism.
- Physical Security: To protect against unauthorized access or tampering, the weather station must be physically secure. Take, for instance through enclosures and locks.
- Network Security: The network for weather stations needs to be secure so as not to allow illegal access or interception of data. This is by firewalls and intrusion detection systems.
- Monitoring: Naturally, the weather station must constantly be kept under surveillance for any abnormality or suspicious activity that suggests an intrusion. By monitoring logs and identifying anomalies, this can be accomplished.

In general, if a model for the security of weather stations which uses Internet of things is to be adopted where technological and human methods are combined, then there must also be authentication along with data encryption in addition to what can already be provided by access control. In the case of firmware update procedures, countermeasures should also be taken against attacks by physical hackers or spoofing using bogus stations. Local network security is just as vital.

Figure 15 shows the types of security solutions.

7.3 Future aspects in this field

There is some of future research directions in the field of IoT-based weather stations:

Integration of 5G and IoT: Examine what effect 5G technology will have on IoT-based weather stations. How much faster are the data transmission rates; when does it take place; and how good is the handset? Energy harvesting and sustainability: Relying on innovative energy harvest methods, such as solar power, wind or kinetic energy to provide power for IoT-based weather stations. In so doing we can achieve their sustainable use without depending on traditional sources of electrical supply. Multi-sensor fusion: Aim to increase the accuracy of weather prediction models by using data fusion from different sensors in IoT-based weather stations such as temperature, humidity, air pressure, wind speed and rainfall. Autonomous calibration and maintenance: Automated calibration and maintenance mechanisms for IoT weather stations should be developed to reduce the need of human intervention, improve accuracy, ensure consistency and high reliability in measurement data collection over long periods. Interoperability and standardization: Interoperability and standardization among the myriad of IoT-based weather stations, enabling smooth data exchange between different devices or platforms; this way they can all be brought together to promote wider collaboration and analysis. Privacy-preserving techniques: Investigate privacy-preservation technologies, to enable individuals' personal weather data collected in an abbreviated form by IoT-based weather stations above their own homes to remain private. Edge intelligence and distributed analytics: Analyze the application of edge intelligence and distributed analytics methods to process and analyze weather data on-site, thereby reducing dependency upon long distance transmission at expensive bandwidth costs, with faster decision making based on real time information. Environmental impact assessment: In order to have more environmentally friendly and sustainable solutions, study the environmental impact of weather stations operating on IoT platforms themselves to understand their manufacturing processes, energy consumption needs during operation time. The discovering destroying process also becomes important later when people no longer can maintain them or manufacture new ones to replace those which break down through wear from use over a period of time.

Climate change adaptation: Investigate how IoT-based weather stations can help us understand and cope with climate change by monitoring and forecasting events related to the earth's atmosphere, such as earthquakes or tornadoes; enumerating their good effects on ecosystems but also its harmful ones for human beings; drawing up plans that control global warming. Integration with emerging technologies: Explore the merging of IoT-based weather stations with emerging technologies such as artificial intelligence, machine learning, blockchain and quantum computing in order to improve accuracy, efficiency and security of monitoring and forecasting systems. By opening this research direction, progress can be made in the development of IoT-based weather stations. Weather forecasting will then become more accurate and resource management more effective; better decisions at all levels of society will follow as a result. sectors impacted by weather conditions.

8. CONCLUSIONS

Our IoT-based weather station demonstrated its capability to update weather conditions based on sensor data, offering a valuable solution for various applications such as agriculture, disaster preparedness, and climate research. The integration of ThingSpeak and MATLAB provided advanced data analysis tools, enabling more in-depth insights into weather patterns, resource management, and environmental modeling. Adding a mobile app that pulls data from the ThingSpeak cloud service in real time making the information may be examined at any time and from anywhere.

While our system boasts strengths such as cost-efficiency and low energy consumption, it is important to acknowledge certain limitations and challenges. Future enhancements could address these limitations and further improve the system's overall performance. Challenges may include the need for continuous sensor calibration, potential susceptibility to extreme weather conditions, and the impact of network disruptions on data transmission.

Our study opens avenues for future research in several directions. Integration with emerging technologies, such as artificial intelligence and machine learning, could enhance the accuracy and efficiency of weather monitoring. Additionally, investigating the environmental impact of IoT-based weather stations and addressing concerns related to privacy and data security present exciting areas for further exploration.

In the context of our IoT-based weather station, a robust security model is imperative. The security measures implemented ensure data protection, operational continuity, and compliance with regulations. This enhances trust among users, maintaining the integrity of the weather station and ensuring the reliable collection and dissemination of weather information. Integrating the discussion of the security model in the conclusion emphasizes the importance of securing IoT-based systems for real-world applications.

In conclusion, our IoT-based weather station represents a significant step forward in environmental monitoring, providing a cost-effective, efficient, and secure solution. The insights gained from this study contribute to the broader field of IoT applications and pave the way for advancements in weather monitoring systems.

REFERENCES

[1] Campbell, T., Fearns, P. (2022). Spatial and temporal validation of in-situ and satellite weather data for the South West Agricultural Region of Australia. Journal of Meteorological Research, 36(1): 175-192. https://doi.org/10.1007/s13351-022-1105-7

[2] Mourtzinis, S., Edreira, J.I.R., Conley, S.P., Grassini, P. (2017). From grid to field: Assessing quality of gridded weather data for agricultural applications. European Journal of Agronomy, 82: 163-172. https://doi.org/10.1016/j.eja.2016.10.013

[3] Ali, M.S., Akode, A.O., Awadalkareem, S.D., Ahmed, F.M. (2017). A design of a remote weather factors monitoring system. In 2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE), Khartoum, Sudan, pp. 1-5. https://doi.org/10.1109/ICCCCEE.2017.7867681

[4] Ngebani, I., Nfanyana, K., Sheikh, S.M., Ishmael, Z. (2020). Design of a microcontroller based internet of things based weather station. Ibo Ngebani Journal of Engineering Research and Application, 8(8): 28-33. https://doi.org/10.9790/9622-0808012833

[5] Kodali, R.K., Sahu, A. (2016). An IoT based weather information prototype using WeMos. In 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), Greater Noida, India, pp. 612-616. https://doi.org/10.1109/IC3I.2016.7918036

[6] Nageswararao, J., Murthy, G.K. (2017). Wireless weather monitor using Internet of Things. i-Manager's Journal on Embedded Systems, 6(1): 30-36. https://doi.org/10.26634/jes.6.1.13894

[7] Baraki, P., Shastri, S., Mohemmed, A., Hegde, A. (2018). Real time weather analysis using ThingSpeak. International Journal of Pure and Applied Mathematics, 120(6): 661-682. https://www.researchgate.net/publication/330134632.

[8] Kadri, A., Kasture, S., Bhandare, G. (2022). Weather monitoring system. EasyChair Preprint, No. 7817. https://easychair.org/publications/preprint/W31q.

[9] Babu, T.S., Reddy, B.S. (2018). Weather monitoring system application using LabVIEW. In 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2018 2nd International Conference on, Palladam, India, pp. 52-55. https://doi.org/10.1109/I-SMAC.2018.8653691

[10] Devaraju, J.T., Suhas, K.R., Mohana, H.K., Patil, V.A. (2015). Wireless portable microcontroller based weather monitoring station. Measurement, 76: 189-200. https://doi.org/10.1016/j.measurement.2015.08.027

[11] Kodali, R.K., Mandal, S. (2016). IoT based weather station. In 2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kumaracoil, India, pp. 680-683. https://doi.org/10.1109/ICCICCT.2016.7988038

[12] Munandar, A., Fakhrurroja, H., Rizqyawan, M.I., Pratama, R.P., Wibowo, J.W., Anto, I.A.F. (2017). Design of real-time weather monitoring system based on mobile application using automatic weather station. In 2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT), Jakarta, Indonesia, pp. 44-47. https://doi.org/10.1109/ICACOMIT.2017.8253384

[13] Malik, A.H., Parray, B.A., Kohli, M. (2017). Smart city IoT based weather monitoring system. International Journal of Engineering and Computer Science, 7(5): 3-8. https://www.semanticscholar.org/paper/Smart-City-IoT-Based-Weather-Monitoring-System-Malik-Jalal/fa316c5186265879f25fe7a7c2326f872cd30086.

[14] Mestre, G., Ruano, A., Duarte, H., Silva, S., Khosravani, H., Pesteh, S., Ferreira, P.M., Horta, R. (2015). An intelligent weather station. Sensors, 15(12): 31005-31022. doi: https://doi.org/10.3390/s151229841

[15] Ioannou, K., Karampatzakis, D., Amanatidis, P., Aggelopoulos, V., Karmiris, I. (2021). Low-cost automatic weather stations in the internet of things. Information, 12(4), 146. https://doi.org/10.3390/info12040146

[16] NarasimhaRao, Y., Chandra, P.S., Revathi, V., Kumar, N.S. (2020). Providing enhanced security in IoT based smart weather system. Indonesian Journal of Electrical Engineering and Computer Science, 18(1): 9-15. https://doi.org/10.11591/ijeecs.v18.i1.pp9-15

[17] Seo, J.W., Lee, S.J. (2016). A study on the integrity and authentication of weather observation data using Identity Based Encryption. SpringerPlus, 5(1): 1225. https://doi.org/10.1186/s40064-016-2834-9