



International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 4, April 2025

ESP32 Based Weather Monitor Device

Vidya. M¹, Kuralmozhi. R²

Assistant Professor, Department of Electronics and Communication Engineering¹, Er. Perumal Manimekalai College of Engineering, Hosur, Tamil Nadu, India¹

P.G Student, Department of Applied Electronics, Er. Perumal Manimekalai College of Engineering, Hosur, Tamil Nadu, India²

ABSTRACT: This paper presents the development of an IoT-based weather monitoring system using the ESP32 microcontroller, designed for real-time environmental data acquisition, display, and wireless transmission. The system utilizes the BME280 sensor to accurately measure temperature, humidity, and atmospheric pressure. Real-time data is displayed on a 16x2 I2C LCD, and a buzzer alert is triggered when any parameter exceeds predefined thresholds. A key feature of the system is a custom-built Python-based graphical user interface (GUI), which wirelessly receives data every five minutes via Wi-Fi. The GUI is designed to be application-flexible, allowing users to adapt it for specific monitoring needs, while also logging the data into an Excel file for long-term analysis. The system is powered by a 2 x 18650 Lithium Battery Shield V8 Module, enabling continuous operation in both battery and USB power modes. Experimental validation confirmed the system's reliability, accuracy, and suitability for real-time monitoring in industrial, agricultural, and remote environments.

KEYWORDS: ESP32, IoT, Weather Monitoring, BME280 Sensor, Python GUI, Real-Time Data Logging, Lithium Battery Shield, Environmental Sensing, Wi-Fi Communication, Customizable Interface.

I. INTRODUCTION

In today's rapidly evolving industrial landscape, environmental monitoring plays a pivotal role in ensuring operational efficiency, workplace safety, and optimal resource management. Weather conditions can significantly impact various industrial processes, affecting productivity, equipment performance, and even the health and safety of workers. Industries such as manufacturing, agriculture, logistics, and warehousing rely heavily on accurate, real-time weather data to make informed decisions and implement necessary precautions to mitigate potential risks.

To address this critical need, we propose an innovative weather monitoring system utilizing the ESP32 microcontroller. This system offers a cost-effective, wireless, and reliable solution for real-time environmental monitoring, capable of measuring temperature, humidity, and atmospheric pressure using the BME280 sensor. The system is designed for efficient operation, offering valuable data to industrial and agricultural applications.

A key feature of the system is its ability to store recorded data in an Excel sheet, allowing for long-term analysis, trend observation, and data-driven decision-making. This feature is particularly valuable in industries that require continuous monitoring and regulatory compliance. Additionally, the system includes a buzzer alert mechanism, triggered when any environmental parameter exceeds predefined safety thresholds, ensuring immediate notification to users so that corrective actions can be taken promptly.

Power efficiency is a core consideration of this system's design. Instead of using the TP4056 charging module, the system employs a 2 x 18650 Lithium Battery Shield for power management, ensuring longer operational life and better portability. The integrated system provides a seamless transition between USB power and battery operation, ensuring consistent functionality even during power supply fluctuations.

To further enhance usability and functionality, the system sends mobile notifications when temperature values exceed predefined limits. This feature ensures that users are instantly notified about critical temperature changes, enabling real-time intervention and enhancing safety. The mobile notification system integrates seamlessly with the wireless communication capabilities of the ESP32, allowing the user to monitor environmental conditions remotely.

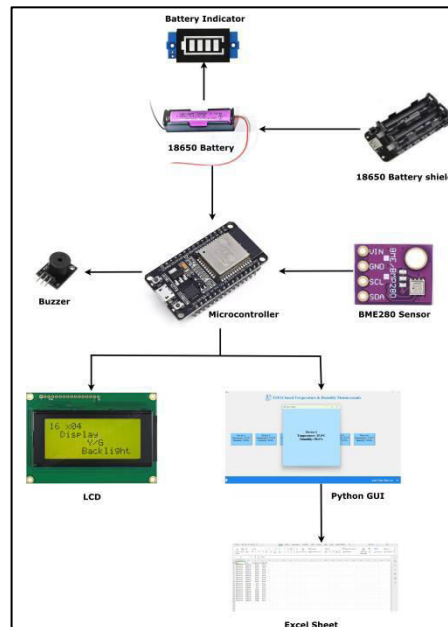


Figure 1. ESP32 Weather monitor device Process flow

The ESP32-based weather monitoring system provides several advantages over traditional weather stations. Unlike conventional systems that rely on standalone sensors with limited connectivity, this system supports real-time wireless data transmission, enabling remote monitoring without the need for constant human intervention. The system's modular design allows for scalability, enabling the integration of additional sensors such as those for wind speed, air quality, and rainfall to expand its functionality further.

By leveraging the power of IoT and automation, this weather monitoring solution addresses the growing demand for reliable, real-time environmental data in industrial applications. The ability to collect, display, transmit, and store weather information wirelessly makes it an invaluable tool for industries that require constant environmental monitoring. This system not only improves operational efficiency but also enhances safety by providing timely alerts for extreme weather conditions. With its combination of affordability, ease of use, and advanced functionality, the ESP32-based weather monitoring system stands out as a practical and innovative solution for modern industrial needs.

1.1 OBJECTIVE

- To collect real-time temperature, humidity, and pressure data using the BME280 sensor.
- To display environmental data on a 16x2 LCD and alert users via buzzer when thresholds are exceeded.
- To ensure seamless power management using a 3.7V lithium-ion battery and 18650 battery shield.
- To transmit weather data wirelessly through ESP32's Wi-Fi to a Python-based GUI every 5 minutes.
- To store sensor data in Excel for long-term monitoring, trend analysis, and decision-making.
- To provide customizable alert and monitoring features suitable for industrial and remote applications.

II. LITERATURE SURVEY

This project proposes an efficient and economical weather monitoring system that combines IoT and Artificial Intelligence. Using image classification via the Teachable Machine platform, the system employs an ESP32 camera module to detect weather conditions visually. The detected output is transmitted to the Blynk application for real-time virtual display, offering a smart, automated solution for weather observation using AI-driven image recognition [1]. This system employs an ESP32 microcontroller with built-in Wi-Fi to monitor key environmental parameters such as temperature, humidity, and rainfall. It integrates DHT11 and raindrop sensors to provide accurate, real-time data,

allowing users to track weather conditions in specific areas. The system is ideal for local weather forecasting and timely alerts based on sensor input [2].

This research discusses the expanding role of the Internet of Things (IoT) in building advanced, intelligent climate monitoring systems. With IoT's integration into various fields such as health, agriculture, and smart cities, the paper highlights how IoT can revolutionize weather tracking by enabling real-time, location-based environmental monitoring. It emphasizes the importance of understanding IoT architecture, sensor technologies, microcontrollers, and communication protocols in designing scalable and accurate systems. The study also explores how technological advances like low-cost sensors and renewable energy sources can enhance the precision and sustainability of weather stations. The work aims to guide future development by combining knowledge from digital learning and environmental data collection [3].

This study focuses on the practical implementation of an environmental monitoring system for two Pakistani cities—Lahore and Sargodha—using IoT technology. It aims to collect real-time data on temperature, humidity, and air quality, addressing the lack of publicly available datasets for these regions. The system not only monitors local environmental conditions but also stores the data on the cloud, making it accessible for future research. This initiative supports efforts toward building cleaner, greener urban environments by enabling data-driven decision-making and facilitating the application of machine learning model [4].

This article outlines the development of a cost-effective asynchronous web server using an ESP32 microcontroller paired with a DHT22 sensor for real-time temperature and humidity monitoring. The system is designed to support environmental management by providing continuous, low-power data acquisition, enabling effective monitoring in various applications where maintaining specific atmospheric conditions is critical [5].

This work presents a smart, sustainable mini weather station built using the ESP32 platform, designed to monitor localized weather conditions. It measures six weather parameters and operates independently using an internal power source, lasting up to 11 hours without external power. The data is uploaded to an online database and made accessible via a website and mobile app, making it particularly useful for disaster-affected areas with limited infrastructure [6].

This study presents a cost-effective real-time environmental monitoring system using IoT technology. It employs an ESP32 microcontroller with multiple environmental sensors and integrates with the Blynk cloud platform for data visualization and management. The system tracks air temperature, humidity, pressure, and dust concentration, offering low power consumption and affordability. It was tested in comparison with a commercial ecological monitoring system to validate its performance [7].

This project integrates real-time weather monitoring with predictive analytics using an ESP32-based hardware module and cloud-based data storage. It collects live sensor data and combines it with 21 years of historical weather records stored in a MySQL database. Machine learning algorithms analyze this data to forecast weather conditions and suggest suitable crops for cultivation. A user-friendly HTML web interface allows users to input queries and view predictions, making the system valuable for both planning and agricultural decision-making [8].

This system leverages ESP32 and the UBI DOTS platform to enable real-time health monitoring by collecting data like heart rate, temperature, and oxygen levels. The IoT-based setup ensures seamless transmission of vital health information to the cloud for analysis and alerts, supporting timely medical responses and enhancing patient care [9].

This health monitoring system uses ESP32 and sensors to track vital signs like oxygen saturation and body temperature, transmitting real-time data via Wi-Fi. The information is displayed on a smartphone app, promoting convenient and continuous health tracking, especially for busy individuals and the elderly [10].

III. METHODOLOGY

The proposed system employs an ESP32-WROOM-32 microcontroller to develop a real-time, low-cost, and power-efficient weather monitoring system. The architecture is designed to be modular and adaptable, integrating sensor data collection, local display, wireless data transmission, and notification mechanisms in a compact framework. The methodology includes both hardware and software design to ensure accurate sensing, efficient data handling, and reliable user notifications.

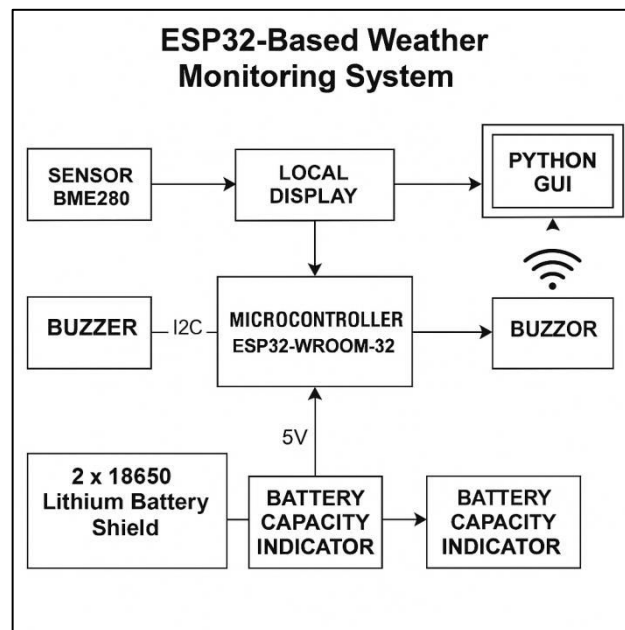


Figure 2. Proposed Block diagram

3.1. System Workflow

The workflow begins with initializing the ESP32 and connected peripherals. The BME280 sensor continuously monitors temperature, humidity, and atmospheric pressure. A Battery Capacity Indicator circuit provides the voltage level of the 18650 battery used in the power module. These readings are periodically updated and displayed on a 16x2 LCD using I2C communication.

If any parameter exceeds its respective threshold, a buzzer provides immediate audible alerts. In parallel, the ESP32 transmits sensor data every 5 minutes via its built-in Wi-Fi module to a custom-developed Python GUI hosted on a local server. The GUI is capable of logging sensor readings into an Excel sheet, which can be used for future reference and analysis.

Additionally, the system includes a mobile notification alert feature. When critical environmental thresholds are breached—especially temperature—a notification is pushed to the user’s phone using Wi-Fi-based messaging or integration with IFTTT/webhooks.

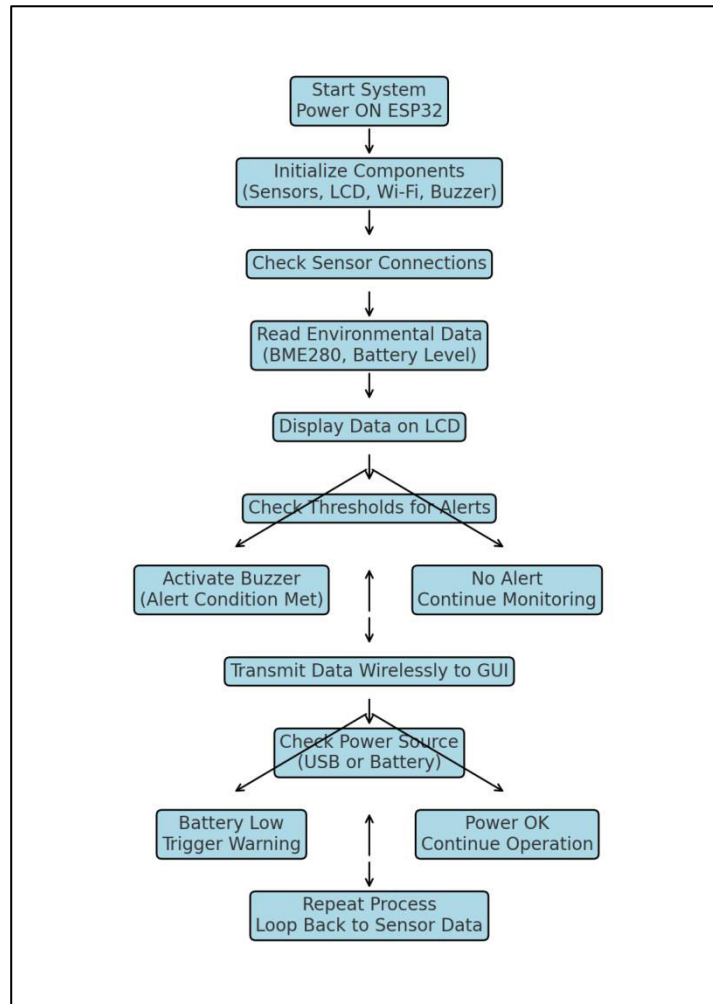


Figure 3. System Workflow

3.2 Hardware Integration

A) ESP32-WROOM-32: Core processing and wireless transmission

The ESP32 module is the central processing unit of the weather monitoring system, responsible for data acquisition, processing, display management, wireless communication, and alert mechanisms. Developed by Espressif Systems, the ESP32 is a powerful, low-power microcontroller that features a dual-core Xtensa LX6 processor, capable of operating at speeds of up to 240 MHz. This high processing speed allows the ESP32 to handle multiple tasks simultaneously, such as reading sensor data, updating the display, and transmitting data over Wi-Fi. One of its most significant advantages is its built-in Wi-Fi and Bluetooth connectivity, making it ideal for IoT applications where wireless data transmission is required. In this project, the ESP32 is responsible for transmitting sensor readings to a Python-based GUI every five minutes, allowing remote monitoring and real-time analysis.

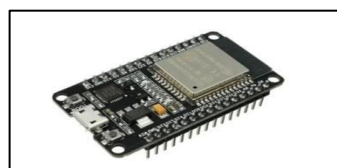


Figure 4. ESP32 Module

B) BME280 Sensor: Measures temperature, humidity, and atmospheric pressure

The BME280 sensor is a compact and highly accurate environmental sensing module that measures temperature, humidity, and atmospheric pressure. Manufactured by Bosch Sensortec, it is widely utilized in weather monitoring systems and IoT applications due to its low power consumption and reliable performance. In this project, the BME280 provides essential real-time weather data, helping industries monitor environmental conditions and make data-driven decisions. It is integrated with the ESP32 microcontroller via the I2C protocol, ensuring efficient and seamless data transmission with minimal wiring.

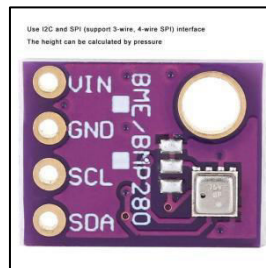


Figure 5. BME280 Sensor

C) 16x2 I2C LCD: Displays real-time readings.

The 16x2 Liquid Crystal Display (LCD) is a widely used and cost-effective display module, ideal for embedded systems and IoT applications. It consists of two rows, each displaying up to 16 characters, making it suitable for showing important environmental parameters such as temperature, humidity, and battery voltage. This display offers a simple and efficient solution compared to other display types, offering good power efficiency and versatility.



Figure 6. 16x2 LCD

D) Buzzer: Alerts users to threshold breaches.

The buzzer module is an essential feature of the ESP32-based weather monitoring system. It provides immediate auditory alerts whenever critical environmental conditions such as high temperature, extreme humidity, or low battery voltage are detected. This ensures that users are promptly notified, even in noisy environments or when they are not looking at the display. The buzzer helps maintain safety and efficiency by enabling timely intervention to address potential issues.



Figure 7. Buzzer Module

E) Battery Capacity Indicator: Monitors remaining battery charge.

The Battery Capacity Indicator is a key component in the ESP32-based weather monitoring system, enabling users to monitor the status of the Li-Ion battery in real-time. This ensures the system remains operational by providing timely alerts when the battery voltage drops below a critical level, helping avoid unexpected shutdowns. It enhances the system's reliability, particularly in remote locations where recharging may not be immediately possible, ensuring continuous performance and reducing the risk of data loss.

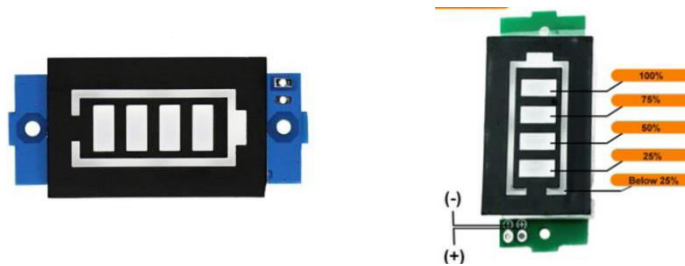


Figure 8. Battery Capacity Indicator

F) 2 x 18650 Battery Shield V8: Provides stable power with USB and battery inputs.

To ensure uninterrupted operation, the system incorporates a 2 x 18650 Lithium Battery Shield V8 Mobile Power Expansion Board. This board supplies stable 5V/3A and 3V/1A outputs to power the ESP32 and peripheral components. The battery shield allows seamless switching between USB charging and battery operation. An onboard battery capacity indicator monitors voltage levels and prevents over-discharge.



Figure 9. 18650 battery shield

3.3 Software Implementation

The Arduino IDE is used for programming the ESP32. Sensor data acquisition, threshold comparison, LCD output, buzzer triggering, and Wi-Fi connectivity are all controlled through this firmware. The Python GUI receives data via socket communication and uses openpyxl or pandas for Excel logging. Notification services are handled by integrating IFTTT or Telegram bots for sending real-time alerts.

This methodology ensures the development of a robust system that supports industrial deployment, ease of customization, and scalability for additional sensors or network-based analytics.

Python Software Description:

Python is a powerful, high-level, and easy-to-learn programming language widely used for general-purpose programming. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. In the context of IoT-based applications, Python is frequently used to create graphical user interfaces (GUIs), handle data logging, and enable communication with hardware via Wi-Fi or serial interfaces.

Python's extensive libraries, ease of syntax, and vast community support make it an ideal choice for developing data visualization tools, automation scripts, and integration with cloud services or IoT dashboards.

Key Features:

- 1) Easy-to-Understand Syntax: Clean and readable code, ideal for rapid development and prototyping.

- 2) Extensive Libraries: Libraries such as socket, openpyxl, pandas, matplotlib, and tkinter help in GUI design, data logging, and real-time plotting.
- 3) Cross-Platform Compatibility: Runs on Windows, macOS, Linux, and embedded systems like Raspberry Pi.
- 4) Integration Support: Seamlessly connects with databases, cloud APIs, and microcontroller communication protocols.
- 5) GUI Development: Modules like Tkinter, PyQt, and Kivy allow for quick design of desktop applications.
- 6) Data Handling & Analysis: Efficient tools for logging sensor data to Excel, CSV, or databases, with capabilities for real-time plotting or trend analysis.

IV. EXPERIMENTAL RESULTS

The ESP32-based weather monitoring system was thoroughly tested to evaluate its performance, accuracy, and reliability. The BME280 sensor consistently measured temperature, humidity, and atmospheric pressure within the specified accuracy range, with readings validated against a calibrated reference device. The 16x2 LCD provided clear and continuous real-time data display, and the buzzer alert system effectively triggered warnings when thresholds were exceeded.



Figure 10. BME Sensor Outputs

Wireless data transmission to the Python GUI over Wi-Fi was stable and efficient, with no data loss or delay. The GUI displayed real-time values and logged data in an Excel sheet, enhancing long-term monitoring capabilities. The system's power management, using a 2x18650 battery shield and battery indicator, ensured uninterrupted operation and timely low-voltage alerts.

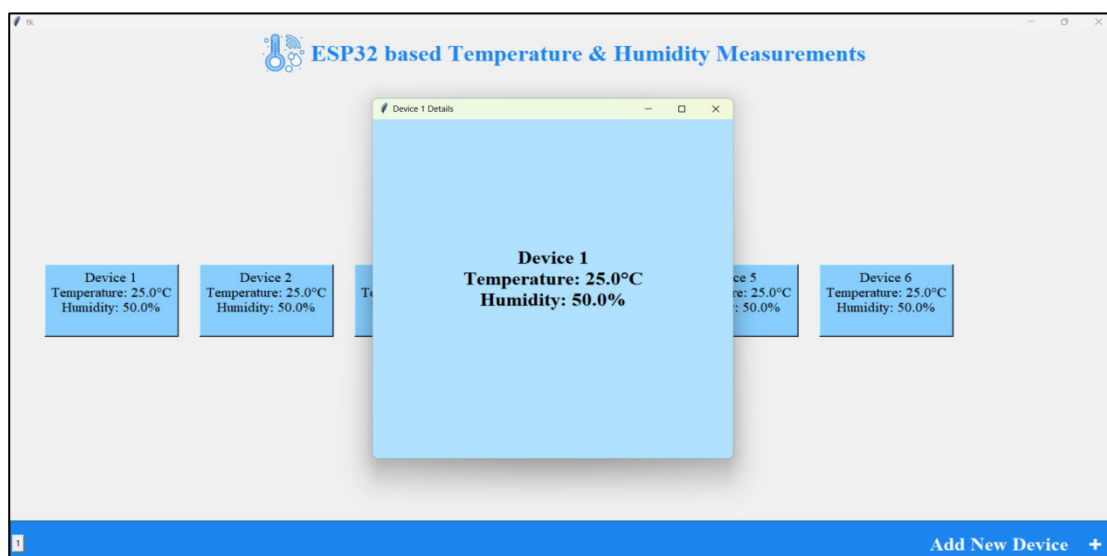
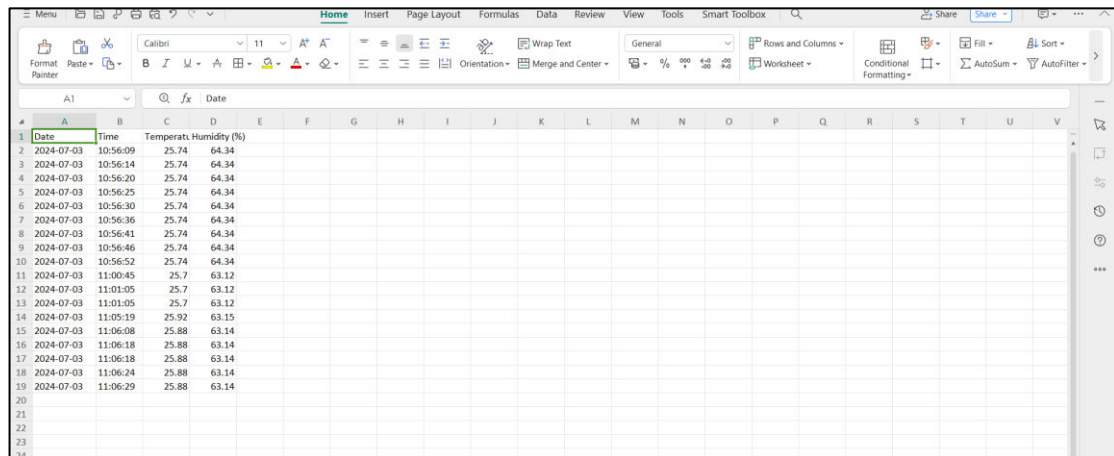


Figure 11. Python GUI Results

A 24-hour stability test confirmed system robustness, with no overheating or communication failures. However, minor limitations such as limited LCD space and the absence of offline data storage were identified. Overall, the system met all functional objectives and proved to be a scalable, reliable, and efficient solution for environmental monitoring.



#	Date	Time	Temperat.	Humidity (%)
2	2024-07-03	10:56:09	25.74	64.34
3	2024-07-03	10:56:14	25.74	64.34
4	2024-07-03	10:56:20	25.74	64.34
5	2024-07-03	10:56:25	25.74	64.34
6	2024-07-03	10:56:30	25.74	64.34
7	2024-07-03	10:56:36	25.74	64.34
8	2024-07-03	10:56:41	25.74	64.34
9	2024-07-03	10:56:46	25.74	64.34
10	2024-07-03	10:56:52	25.74	64.34
11	2024-07-03	11:00:45	25.7	63.12
12	2024-07-03	11:01:05	25.7	63.12
13	2024-07-03	11:01:05	25.7	63.12
14	2024-07-03	11:05:19	25.92	63.15
15	2024-07-03	11:06:08	25.88	63.14
16	2024-07-03	11:06:18	25.88	63.14
17	2024-07-03	11:06:18	25.88	63.14
18	2024-07-03	11:06:24	25.88	63.14
19	2024-07-03	11:06:29	25.88	63.14

Figure 12. Excel sheet Outputs

V. CONCLUSION

This paper presented the design and implementation of an IoT-based weather monitoring system using the ESP32-WROOM-32 microcontroller. The system successfully integrates real-time environmental sensing, data display, wireless transmission, buzzer alerts, and battery monitoring, all within a cost-effective and compact platform. Testing confirmed the accuracy of temperature, humidity, and pressure measurements, the reliability of data transmission via Wi-Fi, and the robustness of the power management system.

The custom Python GUI ensured effective visualization and data logging, while the buzzer and mobile alert system enhanced user awareness during critical environmental changes. Despite minor limitations, such as limited LCD display space and dependence on continuous Wi-Fi connectivity, the system demonstrates strong potential for industrial and remote environmental monitoring applications. Future enhancements could include integration of additional sensors, offline data storage, and a larger or graphical display for extended functionality.

REFERENCES

- [1] Kalpana Murugan, Ravi Kiran Tiruveedhi, Dinesh Reddy Ramireddygar, Deepika Thota, Chandralekha Neeli, "AI based Weather Monitoring System"
- [2] Richard Lincoln Paulraj, M. Tejasree, Swathi M D, S. Ragasahithi, "Effective Weather Monitoring System using MQTT Protocol"
- [3] Muhammad Haris, Bilal Ubaid, Uzma Bibi, Ameer Said, Syed Najm us Saqib, Shahbaz Khan, "Design and Development of IoT Based Weather and Air Quality Monitoring Station"
- [4] Muhammad Afzal, Hafiz Ali Hamza Gondal, Muhammad Najeeb Ullah, Saad Akbar, "IoT Based Smart Environmental Monitring System Using ESP32/We-Mos D1 Mini & ThingSpeak"
- [5] Paul Macheso, Sylvester Chisale, Chisomo Daka, Nelson Dzupire, Justice Mlatho, Didacienne Mukanyirigira, "Design of Standalone Asynchronous ESP32 Web-Server for Temperature and Humidity Monitoring"
- [6] Anshika Gupta, Ankita Tripathi, Reuben Coutinho, Rhea Rodrigues, Prachi Raut, "Smart Sustainable Mini Weather Station"
- [7] Dejan Babić, Ivan Jovović, Tomo Popović, Nataša Kovač, Stevan Čakić, "An Internet of Things System for Environmental Monitoring Based on ESP32 and Blynk"
- [8] Pranathi. S, K.N. Sinchana, A.N. Mukunda Rao, Akash B.U, Vinutha S.J, Rakesh M.M, "Development of IoT Based Weather Reporting System"

- [9] Balamanikandan A; Siddavatam Lakshmi Neeraja; Thathireddy Venkata Kishore; Udatha Deepika; Settipalli Prathyusha; Venkatachalam K, “IoT-Enabled Advanced Health Monitoring System using ESP32 and UBI DOTS“
- [10] Norsuzila Ya'acob; Syila Izawana Ismail; Muhammad Zakwan Mohd Amin; Nur Farhana Abdul Aziz, “Design and Implementation of a Wireless Health Monitoring System for Real-Time Patient Data Using ESP32”



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details