International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

IoT Based Weather Station

Parakram Trivedi*1, Nitesh Kumar*2 , Mr. Hariom Tiwari*3

[*1]UG Student of Department of, Shri Ramswaroop Memorial College of Engineering and Management Lucknow, Uttar Pradesh, India.
[*2] UG Student of Department of, Shri Ramswaroop Memorial College of Engineering and Management Lucknow, Uttar Pradesh, India.
[*3]Assistant professor, Bachelor of Computer Application, Shri Ramswaroop Memorial College of Management Lucknow, Uttar Pradesh, India.

ABSTRACT :

This study explores the development of a cost-effective weather monitoring system using an *ESP32 microcontroller*, along with a variety of environmental sensors, including the *DHT11* (temperature and humidity), *BMP180* (pressure), and a *rain sensor module*. The system provides real-time weather data visualization on a *16x2 LCD display*. Powered by a *USB-Powered*, the system ensures continuous operation in remote locations. A *voltage regulator* (LM7805) ensures a stable power supply. The system's design incorporates low-cost components, making it an accessible solution for educational purposes. Its applications span across various domains, including *Agriculture*, *Disaster Management*, and *Smart Homes*, offering real-time monitoring and data collection capabilities.

KEYWORD- Real-Time Weather Monitoring, ESP32 Microcontroller, Smart Homes, DHT11 Sensor, BMP180 Sensor

INTRODUCTION

We present a smart weather reporting system that leverages the Internet to provide real-time weather updates. This system enables users to directly monitor weather conditions without relying on traditional forecasting agencies. By utilizing various sensors, such as temperature, humidity, and rain sensors, the system continuously tracks these parameters and offers live data on weather conditions. The key components of this system include an Internet of Things (IoT) framework and cloud technology, which work together to collect, process, and share data efficiently.

The IoT-based system captures weather data from sensors and sends it to the cloud through a Wi-Fi module. From there, the information is accessible to users anywhere in the world through an online web server. This setup allows for continuous monitoring of environmental conditions, offering real-time weather data, notifications, and alerts. Additionally, the system uses graphical representations to identify trends in weather parameters, aiding in long-term analysis. The cloud-based architecture ensures seamless data storage, processing, and access, providing a reliable platform for weather reporting.

The system is composed of several key devices, including an Arduino UNO microcontroller, DHT11 temperature and humidity sensors, and a Wi-Fi module for data transmission. These devices gather weather data and transmit it to a web server for real-time updates. Users can set alerts for specific weather events, enhancing the system's utility. In comparison to existing weather monitoring systems, this IoT-based model offers a more flexible and accessible approach to weather data collection and reporting, making it a valuable tool for environmental monitoring.

METHODOLOGY

Participants/volunteers: There were no live volunteers in this study; solely electrical components were used.

Study Design: To create a working prototype that could detect weather, an experimental approach was used. **Resources:**

- DHT11 (Measures temperature (°C) and humidity (%))
- BMP180 Tracks atmospheric pressure (hPa)
- Rain Sensor
- USB (For Power Supply)
- IR Radiation Sensor
- ESP32 WiFi Module

- A board and jumper wires
- 16x2 LCD Display (for real-time monitoring)
- Buzzer (for audio alert)

Utilized Software Libraries:

- DHT.h/SimpleDHT.h: For DHT11 sensor interfacing.
- Adafruit_BMP085.h: For BMP180 pressure sensor.
- ESP8266WiFi.h: Manages WiFi connectivity

Hardware Assembling:

- Attached all sensors (DHT11, BMP180, and rain sensor) and Buzzer to ESP32 microcontroller and ESP32 to 16x2 LCD .
- ESP32 got supply from USB source.
- Sensors gather real-time weather parameters, which are transmitted via an ESP32 WiFi module to cloud platforms like ThingSpeak and Blynk for remote access and analysis.
- Readings of real time weather parameters were shown on a 16x2 LCD display.
- To sound an alert if IR Radiation is high, a buzzer was included.
- A breadboard was used to firmly assemble all of the parts.

Data handling and programming:

- Microcontroller ESP32 microcontroller interfaces with sensors to read data and Sensors Utilizes DHT11 (temperature and humidity), BMP180 (pressure), and a rain sensor module to collect environmental data.
- Programs the ESP32 using the Arduino Integrated Development Environment. Processes sensor readings to compute temperature, humidity, pressure, and rainfall levels.
- Wi-Fi Module ESP32's built-in Wi-Fi capability enables internet connectivity and Cloud Platform Sends data to cloud services like ThingSpeak for real-time monitoring and analysis.
- CD Display displays real-time data locally on a 16x2 LCD screen.
- Serial Communication uses serial communication to send data between sensors and the microcontroller and Data Formatting formats data into structured packets before transmission.

Procedure:

1. Calibrate sensors (e.g., DHT11, BMP180) to ensure accurate measurements of temperature, humidity, and pressure.
2. Initialize the microcontroller (e.g., ESP32) and sensors, and establish Wi-Fi connectivity for data transmission.
3. Continuously collect environmental data from sensors at predefined intervals.
4. Process the collected data and transmit it to a cloud platform (e.g., ThingSpeak) via Wi-Fi for real-time monitoring.
5. Display real-time data on a local LCD screen and provide remote access through a web-based dashboard.
6. Evaluate system performance under various environmental conditions to ensure reliability and accuracy.
7. Install the weather station in a suitable outdoor location for continuous monitoring.

Testing Conditions:

- The system was tested both indoors for calibration and outdoors in an open field to simulate real-world conditions.
- Sensors measured temperature, humidity, atmospheric pressure, and rainfall, with accuracy and precision evaluated through repeatability tests under identical conditions.
- The functionality of real-time data display on the 16x2 LCD and web-based dashboards was evaluated for user accessibility and responsiveness.
- The system's performance under continuous USB power supply and voltage regulation was tested to ensure stable operation.

RESULTS

The weather reporting system was tested in an outdoor environment under varying weather conditions. The system successfully measured and displayed real-time data, including temperature, humidity, atmospheric pressure, and rainfall. The *DHT11 sensor* provided accurate temperature and humidity readings within the expected range, while the *BMP180 sensor* reliably tracked atmospheric pressure changes. The *rain sensor module* effectively detected rainfall, triggering a corresponding signal to the ESP32, which was displayed on the *16x2 LCD screen*.

Power supply through USB is used. The *LCD display* presented weather data with minimal delay, typically refreshing every 1-2 seconds. Real-time data visualization was accurate, with a slight delay in readings due to the sensor processing time, which did not exceed 2-3 seconds.

In terms of durability, the system operated reliably in both direct sunlight and shaded areas, demonstrating the versatility of the solar-powered design. However, it was observed that extreme temperature conditions (above 40°C) caused minor inaccuracies in humidity readings from the *DHT11* sensor. Despite this, the system performed consistently and can be effectively used for real-time weather monitoring in outdoor settings like agriculture and smart home applications.

Parameter	**Observation**
DHT11 Range	• Temperature Range: 0°C to 50°C (Accuracy: ±2°C) • Humidity Range: 20% to 90% RH (Accuracy: ±5%)
BMP180 (Pressure Sensor)	300 hPa to 1100 hPa (Accuracy: ±1 hPa)
Rain Sensor	detects the presence of water (rainfall).
IR Radiation Sensor	The sensor detects infrared radiation emitted from .
ESP32 WiFi Module	The ESP32 Wi-Fi module offers high-speed connectivity (up to 150 Mbps) and supports both Wi-Fi and Bluetooth.
16x2 LCD Display	clear visibility under normal lighting conditions, but its performance may degrade under direct sunlight, which can make the text hard to read.

DISCUSSION

The development of a smart weather reporting system utilizing the Internet of Things (IoT) has significantly transformed environmental monitoring by enabling real-time data collection, processing, and dissemination. This system integrates various sensors such as DHT11 for temperature and humidity, BMP180 for atmospheric pressure, and a rain sensor module, all interfaced with an ESP32 microcontroller. The collected data is transmitted via Wi-Fi to a cloud platform, where it is processed and visualized on a web server accessible globally.

The primary objective of the research is to develop a cost-effective, real-time weather monitoring system utilizing IoT technology. By integrating sensors like DHT11, BMP180, and rain sensors with an ESP32 microcontroller, the system enables continuous data collection and cloud-based visualization. This approach aims to provide accessible weather information for applications in agriculture, disaster management, and smart homes.

Factors like humidity, air temperature, and surface texture can affect sensor performance, leading to reduced accuracy in readings. Reliable internet access is essential for data transmission.

Future enhancements can consist of:

- Incorporating additional sensors such as particulate matter detectors and integrating Artificial Intelligence (AI) for predictive analytics.
- Implement robust security measures to prevent unauthorized access and ensure data integrity.
- Utilize advanced imaging techniques to monitor atmospheric composition and detect pollutants, enhancing environmental monitoring capabilities.

Implementing these enhancements can significantly improve the functionality, reliability, and global applicability of the IoT-based smart weather reporting system.

REFERENCES

1. Kodali, R. K., & Mandal, S. (2016). IoT Based Weather Station. 2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT).
2. DOI:10.1109/ICCICCT.2016.7987944
3. Blynk IoT Platform Documentation. (n.d.). Retrieved from Blynk Official Docs
4. ThingSpeak API Guide. (n.d.). MathWorks. Retrieved from ThingSpeak Documentation.
5. Arduino IDE Setup Guide. (n.d.). Arduino. Retrieved from Arduino Software.
6. Adafruit BMP180 Library Documentation. (n.d.). Retrieved from Adafruit GitHub.