IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 8, April 2025
Impact Factor: 7.67

Weather Monitoring and Forecasting System

Shreeyash Kapase, Kalyani Pawar, Soumya Patil
Department of Electronics and Telecommunications
Sinhgad College of Engineering, Vadgaon, Pune, India

Abstract: *This project proposes an IoT-based Weather Monitoring and Forecasting System that leverages sensor technology and Machine Learning (ML) to provide real-time weather data and accurate predictions. A NodeMCU (ESP8266) microcontroller collects data from DHT11 (temperature & humidity), BMP180 (barometric pressure), and MQ135 (air quality) sensors, transmitting it to a cloud server for analysis. The Decision Tree algorithm classifies weather conditions based on sensor readings and generates forecasts, which are displayed on an interactive dashboard. This cost-effective and scalable system enhances prediction accuracy, disaster preparedness, and environmental monitoring, making it well- suited for agriculture, smart cities, and climate studies.*

Keywords: IoT-based Weather Monitoring

I. INTRODUCTION

Weather forecasting is an essential discipline that plays a crucial role in various sectors, including agriculture, disaster management, aviation, transportation, and environmental protection. Accurate and timely weather predictions allow authorities and industries to make informed decisions that minimize risks, optimize operations, and protect lives and assets. Traditionally, weather forecasting has relied on numerical weather prediction (NWP) models, satellite imagery, and radar-based observations to predict atmospheric changes. While these methods are effective at large-scale predictions, they require high computational resources and often struggle to provide precise, localized weather forecasts in real time.

The rapid advancement of Internet of Things (IoT) technology and Machine Learning (ML) has introduced new possibilities for improving the accuracy and efficiency of weather monitoring and prediction. IoT-based weather monitoring systems can collect real-time environmental data using various sensors and transmit it wirelessly for processing and analysis. This approach enables continuous monitoring of local weather conditions without the need for expensive infrastructure. Additionally, ML algorithms can process large volumes of sensor data to identify patterns and make accurate weather predictions based on historical trends.

This project proposes an IoT-based Weather Monitoring and Forecasting System that integrates real-time sensor technology with ML-based prediction models to provide accurate, localized weather forecasts. The system is designed to collect, process, and analyze environmental data using a combination of temperature, humidity, pressure, and air quality sensors connected to a NodeMCU (ESP8266) microcontroller. The collected data is transmitted to a cloud server for storage and further analysis using the Decision Tree algorithm, which classifies weather conditions and generates predictions based on observed trends. The final output is presented in an interactive web dashboard, allowing users to monitor real-time weather conditions and forecasts conveniently.

One of the major advantages of this system is its low cost and scalability. Traditional meteorological stations require sophisticated infrastructure and expensive equipment, limiting their deployment in rural or remote areas. In contrast, this IoT-based system is built with low-cost, energy-efficient sensors and can be easily deployed in different locations, making weather monitoring accessible to a wider audience.

Moreover, the integration of ML models enhances the accuracy of predictions by learning from past weather patterns and adapting to new environmental conditions over time.

The applications of this system extend beyond weather forecasting. It can be used for climate monitoring, pollution tracking, precision agriculture, and disaster preparedness

Farmers can utilize real-time weather data to make informed decisions about irrigation and crop protection, while environmental agencies can track air quality levels and issue alerts for hazardous conditions. Additionally, smart city implementations can benefit from automated weather monitoring to improve urban planning and infrastructure management.

This paper provides a detailed analysis of the system's design, implementation, and performance. The following sections discuss the literature review, system architecture, ML algorithm implementation, results, and conclusions. By leveraging IoT and ML technologies, this project aims to develop an efficient, accurate, and scalable weather monitoring solution that can contribute to better weather forecasting and environmental monitoring.

II. PROPOSED SYSTEM

The proposed Weather Monitoring and Forecasting System is designed to provide real-time, localized weather predictions by integrating low-cost IoT sensor technologies with Machine Learning (ML) algorithms. Similar to the approach used in compact and cost-effective CNC systems, this project emphasizes simplicity, scalability, and ease-of-use while maintaining robust functionality for continuous environmental monitoring.

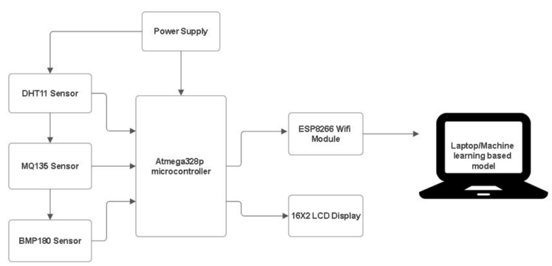
The system functions as an end-to-end solution that:

Collects environmental data: Real-time readings of temperature, humidity, atmospheric pressure, and air quality are captured by dedicated sensors.

Transmits data wirelessly: A NodeMCU (ESP8266) microcontroller serves as the central hub, gathering sensor data and forwarding it to a cloud server via wireless communication.

Processes and predicts weather conditions: The cloud platform applies a Decision Tree ML algorithm to classify current weather patterns and forecast future conditions.

Displays results for user interaction: An interactive web dashboard enables remote monitoring and provides timely alerts based on the predicted weather data



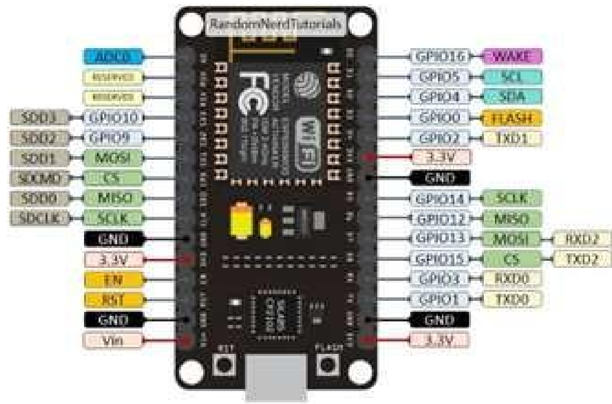
Hardware and Interface

In our Weather Monitoring and Forecasting System, the hardware interface is designed to ensure accurate data acquisition, efficient processing, and reliable wireless communication. Each component plays a specific role in achieving these objectives.

NodeMCU (ESP8266) Microcontroller

The NodeMCU is the central processing unit of the system, integrating a Wi-Fi module with an embedded microcontroller. This combination is pivotal for the following reasons:

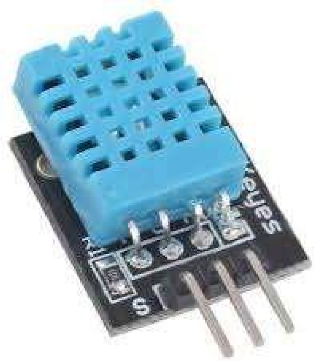
- Wireless Connectivity: It offers built-in Wi-Fi capabilities, enabling seamless data transmission from the sensor module to the cloud or remote server using protocols like MQTT or HTTP.
- Processing Capability: The microcontroller processes sensor inputs, organizes data, and controls the timing for data acquisition.
- Ease of Programming: It supports programming environments like Arduino IDE and Lua, simplifying firmware development for IoT applications.
- Interface Support: The NodeMCU has multiple GPIO pins that allow easy interfacing with various sensors, making it a flexible platform for integrating additional components if needed



DHT11 Sensor (Temperature and Humidity Sensor):

The DHT11 sensor is used for measuring ambient temperature and relative humidity. Key attributes include:

- Sensing Range: It covers a temperature range from 0°C to 50°C with an accuracy of ±2°C, and a humidity range from 20% to 80% with an accuracy of ±5% RH.
- Digital Output: The sensor outputs data in a digital format, which simplifies the interfacing process with the NodeMCU by reducing the need for analog-to- digital conversion.
- Low Power Consumption: Ideal for continuous monitoring in IoT applications, its low power requirements contribute to the overall energy efficiency of the system.



BMP180 Sensor (Barometric Pressure Sensor):

The BMP180 sensor is responsible for capturing atmospheric pressure, which is an essential parameter for weather prediction. Its features include:

- Measurement Range: The sensor can measure pressure between 300 hPa and 1100 hPa, with an accuracy of ±0.12 hPa, ensuring precise data for forecasting.
- Temperature Compensation: It also provides temperature measurements, which can be used for calibration and compensation in pressure readings.
- Digital Communication: Similar to the DHT11, the BMP180 uses an I2C interface for easy connection with the NodeMCU, reducing wiring complexity and enhancing reliability.

IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 5, Issue 8, April 2025

Impact Factor: 7.67

MQ135 Sensor (Air Quality Sensor)

The MQ135 sensor monitors air quality by detecting various gases and pollutants. Its attributes include:

- Detection Range: Capable of identifying pollutants such as CO_2, NH_3, and volatile organic compounds within a detection range of approximately 10 ppm to 1000 ppm.
- Analog Output: Although the sensor outputs an analog signal, the NodeMCU can be interfaced with an external ADC (Analog-to-Digital Converter) if necessary, or use integrated circuit boards that convert the output to digital for simpler processing.
- Application Relevance: By continuously monitoring air quality, the sensor provides critical data that complements meteorological parameters, contributing to a more comprehensive analysis of the environment.

Power Supply and Voltage Regulation

A stable power supply is critical for continuous operation of the system. Key aspects include:

- Voltage Compatibility: The NodeMCU and sensors typically operate at 3.3V to 5V. A dedicated voltage regulator or DC-DC buck converter is used to provide a stable voltage level, protecting sensitive components from voltage fluctuations.
- Battery or Adapter Option: Depending on deployment, the system can be powered by a rechargeable battery or a fixed DC power adapter, ensuring flexibility and portability.
- Power Management: Efficient power management circuits help minimize consumption and extend the operational lifespan of the system, especially in remote deployments.

Connectivity and Data Interface

The hardware interface is further enhanced by the integration of communication protocols that facilitate data exchange:

- GPIO and I2C Interfaces: The NodeMCU's GPIO pins and I2C bus are used to connect and communicate with all the sensors, ensuring synchronized data collection.
- Wireless Data Transmission: Embedded Wi-Fi connectivity in the NodeMCU ensures that real-time data is transmitted to the cloud for processing and visualization. This eliminates the need for manual data retrieval and allows for remote monitoring through a web dashboard.

ATmega328P Microcontroller

It acts as the primary processing unit for interfacing with environmental sensors.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 8, April 2025
Impact Factor: 7.67

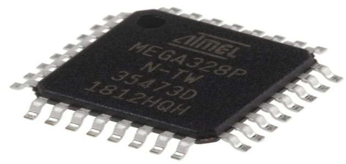
Key Features:

Memory: 32 KB flash, 2 KB SRAM, 1 KB EEPROM.

Speed: Operates up to 20 MHz, suitable for processing sensor inputs in real time.

- Interfaces: Supports UART, SPI, and I²C for seamless communication with sensors (e.g., DHT11, BMP180, MQ135) and external modules.
- Integration: Collects sensor data, performs initial processing, and passes information to the ESP8266 module for wireless transmission.
- Advantage: Its compact size, low power consumption, and robust performance make it ideal for embedded applications such as our weather monitoring system.

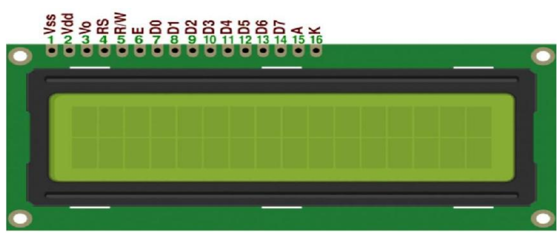
This concise integration ensures that the ATmega328P efficiently manages sensor data and supports reliable system performance in real-time weather monitoring and forecasting.



LCD (16x2) Display

The LCD used is a 16x2 character display, meaning it can show 16 characters per line over 2 lines. This setup is sufficient for showing key information such as temperature, humidity, pressure, and system status in a clear and concise manner.

It provides immediate visual feedback of sensor data (e.g., temperature, humidity, pressure) and system status and is typically connected via a parallel (or serial) interface using data and control lines, ensuring quick information updates. Enables local monitoring without needing remote access to the cloud dashboard.



III. PROCESSING STEPS

The Weather Monitoring and Forecasting System operates via a series of sequential stages to ensure efficient real-time monitoring and prediction.

Sensor Module Assembly

The sensor module is designed to provide a compact, modular, and robust setup for continuous atmospheric monitoring. The sensors are arranged to ensure accurate, reliable, and representative data collection.

Sensor Installation and Placement

- DHT11 Sensor: Mounted securely to measure ambient temperature and relative humidity. Its placement avoids direct sunlight or airflow disturbances, ensuring readings are representative of the local environment.
- BMP180 Sensor: Installed in a stable position for precise barometric pressure measurement, kept away from sources of heat to prevent measurement errors.
- MQ135 Sensor: Positioned to monitor air quality by detecting pollutants, with careful calibration to accurately reflect ambient air conditions.

IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429 Volume 5, Issue 8, April 2025 Impact Factor: 7.67

Build Platform and Housing

- The sensors are integrated onto a custom-designed circuit board within a weatherproof enclosure that protects the components while allowing adequate ventilation.
- The enclosure is designed with proper cable routing and secure connectors to enable easy maintenance and reliable operation.

Electronic Connections and Wiring

Efficient electronic integration is critical for real-time data acquisition and reliable communication. The system's wiring ensures robust connections and stable power delivery.

Control Units and Communication Interfaces

- ATmega328P Microcontroller: Acts as the primary unit for collecting raw sensor data. It performs initial data processing and communicates data to the secondary wireless module.
- NodeMCU (ESP8266): Functions as the wireless communication hub. It receives data from the ATmega328P and transmits it via Wi-Fi to a remote laptop or local server.

Wiring and Connectivity

Sensor Connections:

- The DHT11 is connected to a digital I/O pin on the ATmega328P.
- The BMP180 utilizes the I2C interface, connecting to the dedicated SDA and SCL pins.
- The MQ135 sensor, if using analog output, connects to one of the microcontroller's analog input pins; an external ADC may be used if necessary.

Inter-Module Communication:

- Data collected by the ATmega328P is relayed to the NodeMCU through serial communication.

Power Supply System:

- A regulated power supply, often through a DC-DC buck converter, provides a stable voltage (commonly 5V or 3.3V) to all components.
- Power management is optimized to ensure low energy consumption and sustained operation, whether powered by a battery or a fixed adapter.

Cable Management:

Connections are secured using soldered joints or reliable connectors, with cables neatly arranged to avoid interference and ensure ease of troubleshooting.

Software Implementation

- The software framework is designed to transform raw sensor inputs into actionable weather forecasts with an intuitive user interface.

Data Acquisition and Preprocessing

- Custom firmware is developed (using the Arduino IDE) for the ATmega328P to continuously read sensor data at set intervals.
- Preprocessing routines include noise reduction, data normalization, and timestamping, ensuring that the data is reliable and ready for analysis.

Wireless Data Transmission

- The NodeMCU is programmed to establish a Wi-Fi connection and transmit preprocessed data to a remote laptop or server using standard protocols such as MQTT or HTTP.
- This local network setup ensures that data is stored remotely without relying on third-party cloud services.

Prediction and Visualization

- On the remote system, a Machine Learning module (implemented in Python with libraries such as scikit-learn) processes the incoming sensor data using a Decision Tree algorithm.
- The algorithm classifies weather conditions and generates forecasts based on the continuous stream of data.
- Forecast results are visualized via an interactive web dashboard, and immediate sensor readings are displayed locally on a 16x2 LCD for on-site monitoring.

Workflow Summary

- Initialization: Power up the ATmega328P and NodeMCU, initialize sensors, and configure the LCD.
- Data Collection: Continuously acquire sensor data with appropriate timestamping.
- Preprocessing: Clean, normalize, and prepare data for transmission.
- Data Transmission: Send data wirelessly from the NodeMCU to a remote laptop/server.
- Prediction: Process data using the Decision Tree ML model to forecast weather conditions.
- Visualization: Update the local LCD and remote dashboard with current and forecasted weather information.
- Continuous Operation: Repeat the cycle for ongoing real-time monitoring and prediction.

IV. CONCLUSION

The Weather Monitoring and Forecasting System developed in this project provides a real-time, cost-effective, and efficient solution for weather prediction. By integrating multiple sensors with the ATmega328P microcontroller and NodeMCU, the system successfully collects, processes, and transmits environmental data to a remote server for further analysis. The implementation of the Decision Tree algorithm enables reliable weather forecasting based on historical and real-time sensor data.

Unlike traditional weather monitoring systems that rely on large-scale infrastructure, this system offers a localized and scalable approach without the need for cloud-based storage. The use of a local server ensures data privacy and accessibility, making it ideal for personalized and regional weather forecasting. The real-time data visualization on an LCD and a remote dashboard enhances usability, allowing users to monitor weather conditions efficiently.

Overall, this project demonstrates the feasibility of IoT-based weather forecasting using machine learning, with potential applications in agriculture, disaster preparedness, and environmental monitoring. Future enhancements could include additional sensors for more parameters, optimization of the prediction model, and the integration of an alert system for extreme weather conditions.

REFERENCES

[1] International Journal of Advanced Research in Computer and Communication Engineering ISO3297:2007 Certified Vol. 5, Issue 9, September 2016

[2] International Journal of Engineering Trends and Technology (IJETT) – Volume 32 Number 2-February2016

[3] International Journal of Engineering Science and Computing, May2017

[4] Sagar J. S. T. , M. S. Balamurugan and J. A. Vivek,"A wireless framework for automotive monitoring systems," in Indian Journal of Science and Technology, Vol 8(19), IPL0146, August 2015

[5] https://www.openhacks.com/uploadsproductos/ra in_sensor_module.pdf

[6] content://com.sec.android.app.sbrowser/readingli st/0721190655.mhtml

[7] https://www.hackster.io/techmirtz/using-16x2- lcd-with-arduino-d89028

[8] A. D. Vidhate, S.G. Hate, "Advancement of Brain Tumor DetectionUsing SOM-Clustering and Proximal Support Vector Machine",IJAREEIE, ISSN no. 2320-3765, vol. 6, issue 7, pp. 5552 –5558, 2017.

[9]A. D. Vidhate et.al.," Garbage Classifier Using Arduino", Journal of Mechatronics & Automation, ISSN 2455-1988, vol 11, issue 1, pp.18-23, 2024.