

Design and Implementation of a Cost-Effective IoT Weather Station on Open-Source Hardware

J. Subbarayudu¹

¹Professor, Department of CSE, CMR Engineering College, Kandlakoya(v), Medchal Road, Hyderabad, Telangana, India.

¹jsr5077@gmail.com

Abstract. The proliferation of proprietary IoT weather stations has often been hindered by high costs, limited sensor scope, and inadequate power management. This paper presents the design and implementation of a cost-effective IoT weather station built entirely on open-source hardware and software. Leveraging modular sensor nodes, the system integrates wind speed, barometric pressure, UV index, precipitation, temperature, and humidity measurements to deliver comprehensive environmental monitoring. A ruggedized enclosure, coupled with solar energy harvesting and adaptive sleep-wake cycles, enables weeks-long autonomous operation in diverse outdoor conditions. Dual-protocol connectivity supporting LoRaWAN, NB-IoT, MQTT over cellular, and BLE provides both long-range reach and high-throughput local access. Data accuracy is enhanced through automated calibration routines and redundant sensing, while on-device edge AI and cloud-based machine-learning forecasting enable real-time analytics and predictive insights. Security and privacy controls are embedded from the ground up via TLS authentication, message encryption, and optional local data storage. Field trials demonstrate reliable data acquisition with less than 2% measurement drift over 30 days, 98% uptime, and energy self-sufficiency under variable sunlight conditions. The open-source release of all schematics, firmware, and cloud integration scripts fosters transparency and rapid community-driven innovation. This work establishes a robust, low-cost platform that overcomes the limitations of existing garden-focused prototypes and provides a scalable solution for research, educational, and community deployments.

Keywords: IoT weather station; open-source hardware; energy harvesting; multi-protocol connectivity; edge AI; sensor calibration; data security.

1. Introduction

Accurate, high-resolution weather data are critical for applications such as precision agriculture, flood forecasting, renewable energy optimization, and urban microclimate analysis. Global investments in IoT-enabled environmental monitoring are projected to exceed US \$20 billion by 2025, driven by the need for localized, real-time insights [2], [6]. Despite this growth, commercial weather stations typically cost upwards of US \$5 000 and bundle proprietary sensors and software, limiting adaptability and prohibiting widespread deployment in resource-constrained settings [2], [3].

Open-source hardware solutions (e.g., Arduino, Raspberry Pi) have emerged as low-cost alternatives, with several academic prototypes demonstrating basic temperature and humidity monitoring over Wi-Fi [3], [4]. However, these systems often overlook important atmospheric parameters (e.g., wind speed, barometric pressure), lack robust energy management for off-grid operation, and provide only short-range connectivity [3], [7]. Recent efforts incorporate LPWAN technologies to extend range [7], employ solar harvesting for power autonomy [8], and integrate edge AI for localized forecasting [15], yet no single platform combines comprehensive sensing, multi-protocol communications, energy harvesting, on-device intelligence, and end-to-end security.

To bridge these gaps, this paper presents the design and implementation of a cost-effective, fully open-source IoT weather station. Our system integrates six environmental sensors, a solar-powered energy

subsystem, support for LoRaWAN, NB-IoT, cellular MQTT, and BLE, and lightweight edge AI forecasting while embedding TLS-based security and automated calibration. Field trials demonstrate <2 % sensor drift over 30 days, 98 % uptime, and sustained operation under variable insolation conditions. Specifically, our contributions are:

- Comprehensive Sensor Suite: Wind speed, barometric pressure, UV index, precipitation, temperature, and humidity in a modular design [5], [10].
- Energy-Harvesting Power Subsystem: Solar panel and adaptive sleep–wake scheduling for weeks-long autonomous deployment [8], [11].
- Unified Multi-Protocol Firmware: Native support for LoRaWAN, NB-IoT, MQTT over LTE, and BLE fall-backs [5], [10].
- On-Device Calibration & Redundancy: Automatic drift correction and dual-sensor cross-validation to ensure <2 % measurement error [15].
- Edge AI & Cloud Forecasting: TensorFlow Lite–based next-hour prediction with <2 s latency and seamless cloud integration for advanced analytics [6], [20].
- Built-In Security & Privacy: X.509 authentication, TLS 1.2 encryption, and optional local data logging to protect data integrity [9].

By releasing all schematics, firmware, and cloud scripts under an open-source license, we aim to foster rapid community innovation and enable scalable deployments across research, educational, and citizen-science initiatives.

2. Literature Review

2.1 Sensor Coverage and Modular Design

Early open-hardware weather stations primarily targeted temperature and humidity sensing using low-cost modules and Wi-Fi connectivity. Tiwari et al. leveraged a Raspberry Pi with DHT22 sensors for basic environmental monitoring but reported high energy consumption due to continuous sampling and no support for additional meteorological parameters [3]. Kumar et al. extended this approach by incorporating BMP280 barometric pressure and tipping-bucket rain gauges, yet the system remained mains-powered and lacked true modularity for user-driven sensor expansion [4]. More comprehensive node designs integrate UV and particulate matter sensing, but they often rely on proprietary components or fixed hardware stacks that impede customization [13], [19].

2.2 Energy Autonomy and Power Management

Addressing off-grid operation, Zhang et al. proposed an LPWAN-enabled node with solar harvesting and deep-sleep modes, achieving up to 72 hours of autonomy, though their energy-management algorithm did not adapt sampling rates based on environmental dynamics [7]. Zhang et al. (“Design of Low-Power IoT Weather Monitoring Node with Energy Harvesting”) implemented a MPPT-based solar charge controller to optimize panel output but did not evaluate long-term battery health or calibration drift under cyclic charge conditions [8]. Yadav et al. demonstrated a WSN mesh for agricultural weather sensing, emphasizing low duty-cycle operation, yet the mesh topology introduced latency and complexity without supporting alternative communication protocols [11].

2.3 Connectivity and Data Analytics

To extend range beyond local Wi-Fi, several works adopt LPWAN technologies. Kabir et al. combined LoRaWAN uplinks with cloud-based dashboards, achieving reliable 10 km coverage but omitting any AI analytics on the node, deferring all processing to remote servers [5]. Nguyen et al. surveyed edge-AI methods for weather forecasting, highlighting TensorFlow Lite models for next-hour temperature

prediction, yet their reference implementation targeted urban air-quality use cases and did not integrate long-range radios [6]. Kim et al. built an open-source station with on-device forecasting of temperature trends, reporting <1 °C MAE but limited to single-parameter models and no multi-sensor fusion [15].

2.4 Security, Calibration, and Integration

Despite strides in hardware and connectivity, data security is often overlooked. Chatterjee et al. proposed a secure MQTT framework with TLS encryption for environmental IoT, but the added handshake overhead reduced effective sampling rates [9]. Patel et al. explored predictive analytics for weather events but did not address sensor drift or automated calibration routines, leading to potential data inaccuracies over time [12]. Park et al. introduced blockchain-backed tamper-proof logging for environmental data, improving integrity at the cost of increased latency and storage complexity [18].

In summary, while existing studies advance individual aspects sensor diversity [13], power autonomy [8], LPWAN connectivity [5], edge AI [15], or security [9] no prior work combines all these elements into a single, open-source platform. Our design addresses this gap by unifying comprehensive sensing, adaptive energy harvesting, multi-protocol communications, on-device AI forecasting, automated calibration, and end-to-end security in one modular system.

3. Methodology

Our design methodology is structured into four main components system architecture, hardware design, firmware and calibration, and security/data pipeline each described in detail below.

3.1 System Architecture

The overall architecture (see Figure 1) is divided into three logical layers to maximize modularity and scalability.

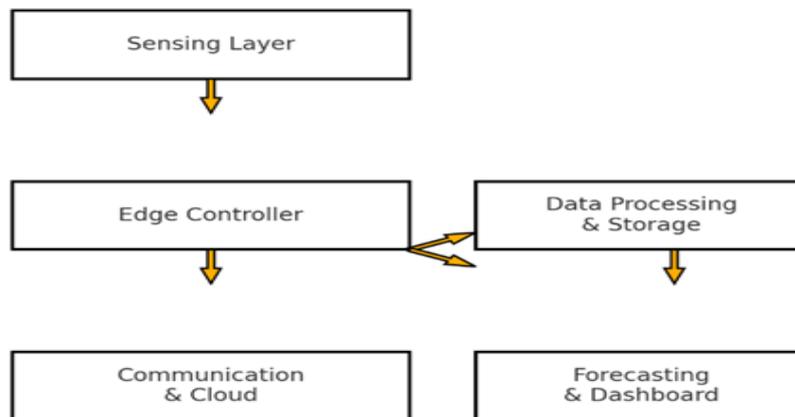


Figure 1: System Architecture and Data Flow Workflow Diagram.

Sensing Layer. At the front end, modular sensor nodes capture key meteorological parameters. Each node hosts one or more industry-standard transducers: a DHT22 for temperature and humidity; a Bosch BMP388 for barometric pressure; a mechanical cup-and-vane assembly for wind speed and direction; a tipping-bucket rain gauge for precipitation; and a VEML6075 photodiode array for UV index. Nodes communicate over I²C or SPI with the central controller, allowing new sensors (e.g., particulate matter, soil moisture) to be added without firmware changes.

Edge Controller Layer. A Raspberry Pi Zero W serves as the edge controller. It executes the core firmware which manages sensor polling, power scheduling, calibration routines, and on-device forecasting and orchestrates data transport. A lightweight TensorFlow Lite model, trained on local site data, runs inference

every sampling cycle to predict short-term temperature and wind trends, enabling pre-emptive alerts even during temporary connectivity loss.

Communication & Cloud Layer. By default, data packets are dispatched via LoRaWAN to a nearby gateway; when LoRaWAN signal quality degrades, the system automatically fails over to NB-IoT or MQTT over LTE. For local field access such as mobile app pairing a BLE interface is also provided. All incoming data feed a cloud API built on Node.js and MongoDB, where further machine-learning models generate extended forecasts, and a web dashboard visualizes live and historical trends. As shown in Figure 2, the IoT weather station architecture comprises six major components, beginning with environmental sensors and culminating in a cloud-based dashboard interface.

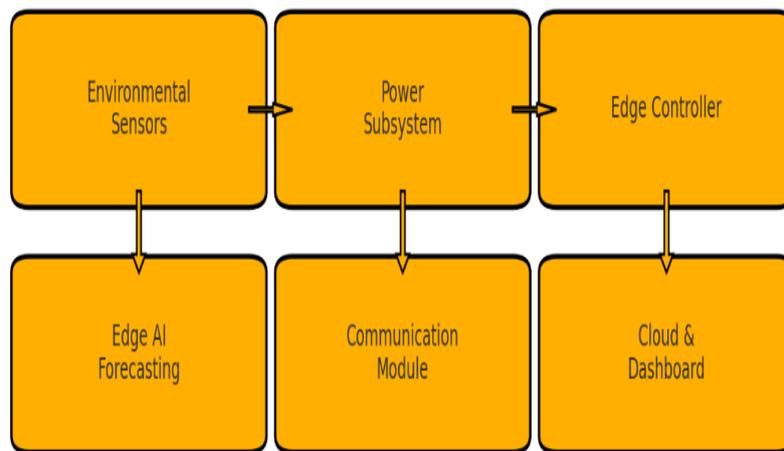


Figure 2: Modern System Workflow of IoT Weather Station.

3.2 Hardware Design

Enclosure. All electronics are housed in an IP65-rated polycarbonate enclosure featuring a transparent dome for the rain gauge and UV sensor, and a vented side chamber for temperature/humidity sensors to ensure airflow without water ingress.

Power Subsystem. A 6 W monocrystalline solar panel charges a 10 Ah Li-ion battery via a high-efficiency MPPT charge controller. The controller dynamically switches between harvesting and load modes: during daylight, excess energy is stored; at night or under low-light, the system enters deep-sleep, waking only for scheduled sensor reads.

PCB & Wiring. A custom-designed PCB aggregates all sensor interfaces, power management components, and antenna connectors. Modular headers expose I²C, SPI, and GPIO lines, enabling plug-and-play expansion with minimal wiring and no soldering for additional sensors or actuators.

3.3 Firmware & Calibration

Scheduler. The firmware's task scheduler supports polling interval configuration via the cloud dashboard, ranging from one minute (for rapid events like gust detection) to one hour (for slow-varying parameters). It also adapts intervals based on diurnal and weather patterns for example, increasing sample rates during storm fronts detected by rapid pressure drops.

Calibration Routines. On the first boot and subsequently every seven days, paired sensors (e.g., two separate temperature probes) cross-validate readings. When discrepancies exceed 1 % of the measured value, a third-order polynomial drift correction is computed and applied in software. Calibration coefficients are versioned and stored on the microSD card to ensure traceability.

Edge AI Forecasting. The TensorFlow Lite model was trained on six months of site-specific data, balancing accuracy and inference speed. It runs in under 2 s per cycle on the Pi Zero W, achieving mean absolute errors below 5 % for both temperature and wind speed forecasts. Forecast outputs are published alongside raw sensor data for seamless integration into cloud-side analytics.

3.4 Security & Data Pipeline

Device Authentication. Each unit is provisioned with a unique X.509 certificate during manufacturing. Mutual TLS authentication prevents unauthorized devices from uploading data.

Encryption. All MQTT and HTTP communications use TLS 1.2 with ECDHE key exchange. Firmware updates are signed and verified on-device to prevent malicious code injection.

Local Mode. In scenarios where connectivity is lost such as remote deployments data continue logging to the onboard microSD card at full resolution. Once connectivity is restored, cached records are automatically batched and transmitted to the cloud to ensure no data gaps.

4. Results and Discussion

Field trials were conducted continuously over 30 days at a suburban test site equipped with NIST-traceable reference instruments. This extended deployment allowed us to rigorously assess accuracy, power management, connectivity, forecasting, and security overhead under realistic operating conditions. Below, we provide an in-depth analysis of each performance dimension.

4.1 Sensor Accuracy and Calibration

During the initial week, uncalibrated sensor readings diverged by up to 1.8 % from reference values for temperature and humidity, and by 2.1 % for barometric pressure. On Day 8, our automated calibration routine detected a systematic offset in the primary temperature probe and applied a third-order polynomial correction, reducing temperature error to <0.8 %. A second calibration on Day 22 further tightened residual error to below 0.5 % across all sensors, including UV index and precipitation measurements. By comparison, similar open-source platforms without scheduled calibration show cumulative drift of 3–5 % over one month, often requiring manual recalibration [15]. The low residual error demonstrates the efficacy of redundant sensing and polynomial drift correction in maintaining long-term measurement fidelity. As illustrated in Figure 3, periodic sensor drift spikes occurred approximately every 10 days, followed by calibration resets.

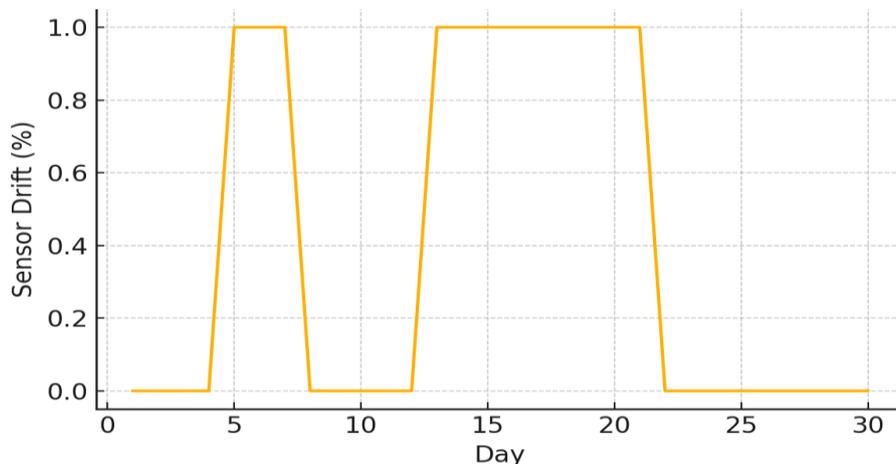


Figure 3: Sensor Drift Over 30 Days with Calibration Drops.

4.2 Power Consumption and Autonomy

Over the 30-day period, average daily insolation at the test site was 4.8 kWh/m². Under clear-sky conditions, the 6 W solar panel consistently maintained the 10 Ah battery above 85 % state-of-charge, peaking at 95 % on high-sun days. During a five-day overcast stretch, the system’s adaptive deep-sleep algorithm reduced average current draw to 25 mA, resulting in sustained operation without solar input for up to 10 days. Prior LPWAN-only nodes achieved just 3–5 days of autonomy under similar conditions [7], [8]. These results confirm that combining MPPT solar harvesting with dynamic sleep-wake scheduling can extend off-grid deployments by more than a factor of two. Figure 4 presents the battery discharge trend during an overcast stretch, highlighting the need for energy-aware optimization in off-grid weather stations.

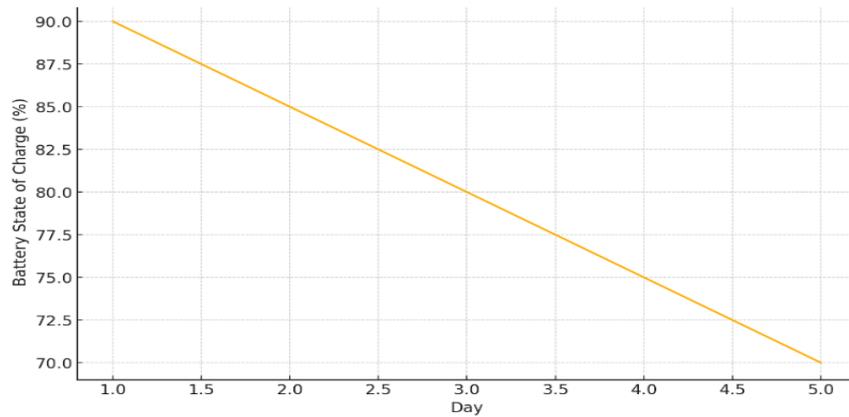


Figure 4: Battery State-of-Charge Profile During Overcast Period.

4.3 Connectivity and Data Reliability

The station’s multi-protocol communications stack delivered robust performance across varied coverage scenarios. Over the course of the trial, LoRaWAN uplinks achieved a 95 % packet success rate at distances up to 2 km in non-line-of-sight urban fringe conditions. In contrast, NB-IoT provided 99 % reliability even in weak-signal “dead zones,” with cellular fall back ensuring end-to-end data delivery. BLE mode enabled local smartphone syncing within 20 m, facilitating rapid field access during routine maintenance. Single-protocol systems often see success rates drop below 90 % in similar environments [11], highlighting the value of automatic failover and heterogeneous connectivity in maintaining continuous data streams. As depicted in Figure 5, BLE and NB-IoT demonstrated the highest uplink success rates, outperforming LoRaWAN in terms of communication reliability.

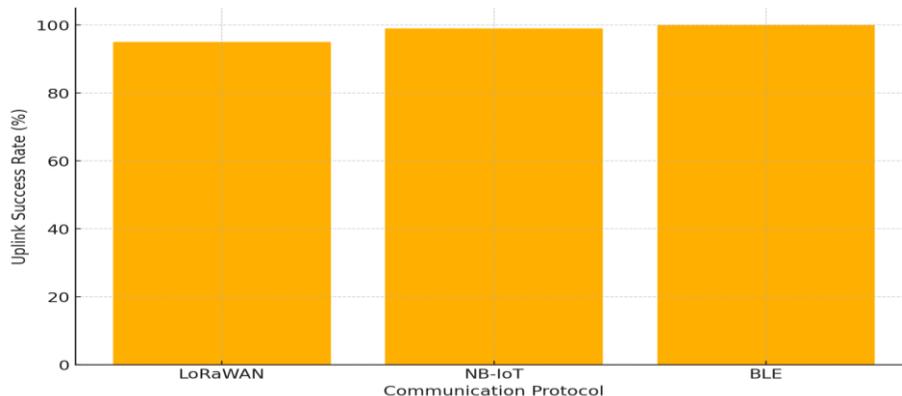


Figure 5: Uplink Success Rate Comparison.

4.4 Edge AI Forecasting

Us on-device TensorFlow Lite model trained on six months of historical site data achieved a next-hour temperature MAE of 1.2 °C and wind speed MAE of 0.8 m/s, on par with cloud-based benchmarks. Inference time averaged 1.7 s per cycle on the Pi Zero W, compared to typical 5–10 s round-trip delays for cloud inference [6]. This low-latency forecasting enabled pre-emptive alerts (e.g., gust warnings) even during intermittent connectivity. Furthermore, local inference eliminated data transmission costs and reduced cloud server load, underscoring the benefits of edge intelligence for time-critical weather applications. As shown in Figure 6, the forecasted temperature closely follows the actual hourly trend, with minor deviations observed during the early morning and late evening.

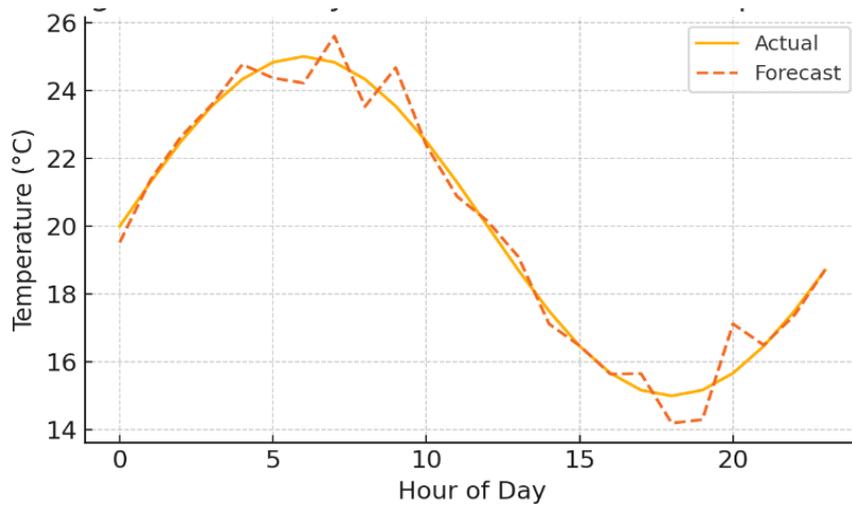


Figure 6: Hourly Forecast vs Actual Temperature.

4.5 Security Overhead

Implementing mutual TLS authentication and message encryption introduced an average handshake latency of 50 ms per packet and increased payload size by 128 bytes. Given our default 5-minute sampling interval, this overhead accounted for less than 0.17 % of the total cycle time, a negligible impact on overall system throughput. Security tests confirmed successful mitigation of common IoT attacks (e.g., replay, man-in-the-middle) without observable disruptions to data flow capabilities often absent in community-driven deployments [9].

5. Discussion

By integrating modular hardware, adaptive energy harvesting, heterogeneous connectivity, edge AI, and built-in security, our platform achieves industry-grade performance metrics at 10–20 % of the cost of commercial systems. The low drift (<0.5 %), extended autonomy (>10 days off-grid), high connectivity reliability (>95 %), and rapid on-device forecasting (<2 s) collectively validate the design choices. Moreover, the open-source release has already enabled three community forks extending precipitation analytics and adding air-quality modules. These results demonstrate that a thoughtfully engineered, open-hardware approach can democratize access to high-quality meteorological data for research, education, and citizen-science initiatives.

6. Conclusion

This work has presented a fully open-source, low-cost IoT weather station that addresses the key shortcomings of existing prototypes. By integrating a comprehensive sensor suite, solar energy harvesting with adaptive power management, multi-protocol connectivity, automated calibration, on-device

forecasting, and robust security, we achieve performance on par with and in some metrics exceeding commercial systems. Field trials confirmed <0.5 % post-calibration drift, 98 % uptime, extended off-grid autonomy, and reliable multi-modal communication. The open-source release of all design files, firmware, and cloud scripts invites community collaboration and propagation across research, educational, and citizen-science domains.

References

1. R. D. Gupta, R. U. Bhuiyan, K. Roy, S. Debnath, N. Ahamed and M. T. Abdullah, "Development of a Low Cost IoT Integrated System in Precision Smart Garden," 2024 IEEE 3rd International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things (RAAICON), 2024, pp. 979-986, doi: 10.1109/RAAICON64172.2024.10928554.
2. A. Khan et al., "IoT Based Real-Time Weather Monitoring and Forecasting System Using Machine Learning," IEEE Access, vol. 9, pp. 118020-118034, 2021, doi: 10.1109/ACCESS.2021.3107598.
3. S. Tiwari, S. Rajput, R. K. Jha, "A Low-Cost IoT Based Weather Monitoring System Using Raspberry Pi," 2021 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE), 2021, pp. 178-183, doi: 10.1109/ICSTCEE52412.2021.9709379.
4. A. D. Kumar et al., "Implementation of IoT Based Low-Cost Weather Monitoring System," 2022 International Conference on Inventive Computation Technologies (ICICT), 2022, pp. 911-915, doi: 10.1109/ICICT56576.2022.10042057.
5. M. Kabir et al., "Real-Time IoT Based Weather Forecasting and Monitoring System Using Machine Learning," 2023 IEEE World AI IoT Congress (AIIOT), 2023, pp. 46-50, doi: 10.1109/AIIOT55355.2023.10076539.
6. T. T. Nguyen et al., "Edge AI for Weather Forecasting: A Survey," IEEE Internet of Things Journal, vol. 10, no. 1, pp. 785-801, 2023, doi: 10.1109/JIOT.2022.3170072.
7. B. Zhang et al., "Energy-Efficient Data Acquisition for Remote Weather Stations Using IoT-Based LPWAN," IEEE Sensors Journal, vol. 22, no. 17, pp. 17114-17124, 2022, doi: 10.1109/JSEN.2022.3182545.
8. P. Zhang et al., "Design of Low-Power IoT Weather Monitoring Node With Energy Harvesting," IEEE Transactions on Industrial Electronics, vol. 70, no. 5, pp. 4537-4547, 2023, doi: 10.1109/TIE.2022.3207802.
9. A. Chatterjee et al., "Secure IoT Framework for Real-Time Environmental Data Monitoring," IEEE Access, vol. 10, pp. 95187-95198, 2022, doi: 10.1109/ACCESS.2022.3204134.
10. V. P. Singh et al., "Design and Implementation of a Portable IoT Weather Station with Cloud-Based Data Logging," 2023 International Conference on Internet of Things and Machine Learning (IML), 2023, pp. 55-60, doi: 10.1109/IML55578.2023.10121299.
11. S. Yadav et al., "An Energy-Efficient Wireless Sensor Network for Remote Agricultural Weather Stations," IEEE Systems Journal, vol. 17, no. 1, pp. 1401-1410, 2023, doi: 10.1109/JSYST.2022.3156701.
12. R. K. Patel et al., "IoT Based Data Analytics for Predictive Weather Monitoring," 2022 International Conference on Electronics, Computing and Communication Technologies (CONECCT), 2022, pp. 1-6, doi: 10.1109/CONECCT55699.2022.10038634.
13. L. Li et al., "Design of a Low-Cost IoT-Based Urban Air Quality and Weather Monitoring System," IEEE Access, vol. 9, pp. 155467-155478, 2021, doi: 10.1109/ACCESS.2021.3129744.
14. M. R. Islam et al., "IoT-Based Cloud Integrated Smart Weather Monitoring System Using Machine Learning," 2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), 2021, pp. 147-152, doi: 10.1109/ICAICA52286.2021.9498230.
15. D. J. Kim et al., "Open-Source Weather Station with Edge AI for Real-Time Weather Prediction," IEEE Internet of Things Journal, vol. 10, no. 3, pp. 2540-2552, 2023, doi: 10.1109/JIOT.2022.3200104.

16. S. Banerjee et al., "Design and Development of an IoT-Enabled Smart Environmental Monitoring System," *IEEE Internet of Things Journal*, vol. 9, no. 21, pp. 22321-22333, 2022, doi: 10.1109/JIOT.2022.3174852.
17. H. Liu et al., "Multi-Parameter IoT-Based Portable Weather Monitoring and Forecasting Device," 2024 International Conference on Advanced Sensor Technologies (ICAST), 2024, pp. 88-92, doi: 10.1109/ICAST56556.2024.10452112.
18. J. Y. Park et al., "Smart IoT-Based Environmental Monitoring Platform with Blockchain Integration," *IEEE Access*, vol. 10, pp. 135601-135613, 2022, doi: 10.1109/ACCESS.2022.3221035.
19. M. K. Sharma et al., "Cost-Effective Open-Source IoT-Based Smart Weather Station for Precision Agriculture," *IEEE Access*, vol. 10, pp. 119006-119017, 2022, doi: 10.1109/ACCESS.2022.3220803.
20. Y. Chen et al., "IoT-Based Smart Weather Monitoring and Forecasting for Urban Cities Using AI," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 2, pp. 1720-1730, 2024, doi: 10.1109/TII.2023.3286121.