PROGRESS IN ENGINEERING APPLICATION AND TECHNOLOGY
e-ISSN: 2773-5303

PEAT

Vol. 7 No. 1 (2026) 290-297
https://publisher.uthm.edu.my/periodicals/index.php/peat

AI-Powered Smart Warehouse Monitoring System Using NodeMCU ESP 32

Yong Jin Wei[1], Azlina Bahari[1]*

[1] *Department of Electronic Engineering Technology, Faculty of Engineering Technology, Universiti Tun Hussein Onn, 84600 Pagoh, Johor, MALAYSIA*

*Corresponding Author: lina@uthm.edu.my
DOI: https://doi.org/10.30880/peat.2026.07.01.033

Article Info

Received: 12 January 2026
Accepted: 20 February 2026
Available online: 25 May 2026

Keywords

AI, IoT, Smart Warehouse, Inventory Optimization, Environmental Control, RFID, ESP32.

Abstract

The rapid growth of e-commerce and global supply chains has increased the demand for efficient warehouse management. Traditional warehouse systems frequently struggle with inventory errors, inefficient space utilization, and inadequate environmental monitoring, resulting in increased operational costs and product losses [6]. This project presents the development of an AI-powered Smart Warehouse Monitoring System that integrates IoT sensors (DHT11, MQ-2), RFID technology, and machine learning to optimize inventory management and environmental conditions. The system utilizes an ESP32 microcontroller to collect real-time data on temperature, humidity, and gas levels, transmitting this information to a Blynk cloud platform for remote monitoring and automated alerting. The results for this project demonstrate significant improvements in inventory accuracy and storage efficiency, validating the effectiveness of AI-powered innovative warehousing in reducing human error and preventing spoilage.

1. Introduction

Warehouses are strategic structures within the global supply chain, as they store and process products before they reach consumers or enter production processes. In the current era of rapid e-commerce growth, the importance of efficient warehouse management cannot be overstated [1]. Historically, these facilities have relied on manual housekeeping and traditional recording methods to track stock and regulate environmental parameters [2]. To date, research on the integration of Artificial Intelligence (AI) and the Internet of Things (IoT) has gained great momentum as industries transition toward Industry 4.0. It is widely assumed that adopting these smart technologies will allow warehouses to operate with higher efficiency and respond more dynamically to changes in inventory status [3].

To address these issues, a project named "AI-Powered Smart Warehouse Monitoring System for Inventory Optimization and Environmental Control Using NodeMCU ESP32" has been designed. By integrating IoT-based technologies, RFID, and a cloud system, the project enhances the warehouse system by updating environmental information and inventory stock for each component within the warehouse [4].

2. Methodology

The project's methodology is based on the previous project studied, which corresponds to the required components. This project aims to design and develop an AI-powered inventory tracking system using RFID sensors for real-time stock monitoring and implement AI-based predictive maintenance for temperature and humidity control.

2.1 Materials

The hardware implementation of the proposed system comprises a NodeMCU ESP32 microcontroller, a DHT11 temperature and humidity sensor, an MQ-2 gas sensor, and an MFRC522 RFID module. IoT-based warehouse monitoring systems using environmental sensors have been widely adopted to reduce spoilage and improve storage safety [5]. These components were selected due to their low cost, ease of integration, and suitability for real-time warehouse monitoring applications.

2.2 System Block Diagram

The system architecture is structured as an input-process-output model centered on the ESP32 microcontroller (refer to Figure 1). On the input side, it consists of a power supply and a suite of sensors, including the MQ-2, which detects gas and smoke, the DHT11, which detects temperature and humidity, and an RFID module. For the processing part, the ESP32, as the main controller, collects real-time data from all inputs and prepares it for transmission. While on the output side, the Blynk app serves as the interface, displaying live environmental values and inventory status updates.

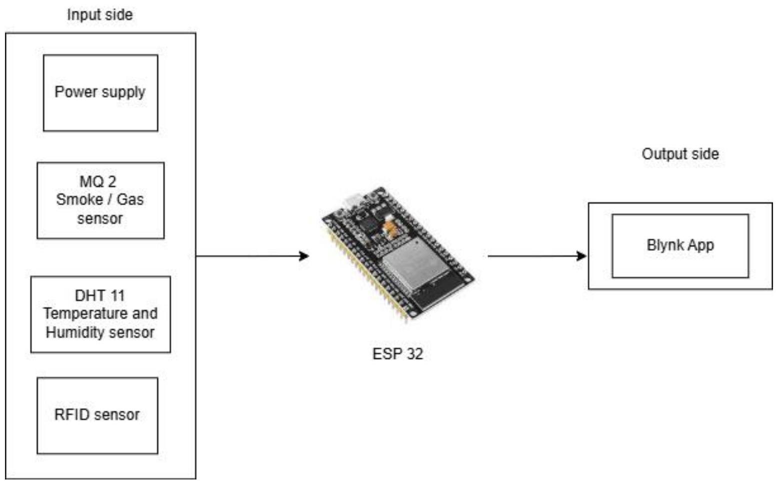


Fig. 1: *System Block Diagram for Smart Warehouse Monitoring System*

2.3 System Flowchart

Figure 2 shows the whole flow of the Smart Warehouse Monitoring System. The process begins by initializing sensors, including temperature and humidity sensors, as well as gas and smoke detectors, and establishing network connectivity. These sensors are continuously monitored to ensure that safe storage conditions are maintained. This ensures all hardware components are operational and ready to collect real-time data. At the same time, an RFID reader scans tags to track item entry or exit conditions, logging this data along with the related IDs in the inventory database.

Next, the system analyzed the combined sensor and RFID data to detect abnormal conditions, such as temperature fluctuations, unexpected value changes, or high hazard gas levels. If no alerts are detected, the data proceeds as normal data and is sent directly to Blynk for storage and monitoring. On the other hand, if one of the abnormal conditions occurs, an alert will be triggered, and the system will send notifications through SMS or email to warehouse managers. During alerts, the system displays “sending data," formats the information, and transmits it to the cloud. Even without alerts, sensors, and inventory data are formatted and sent to Blynk on a regular basis. The system operates in a cyclic manner, with a 5-second delay between iterations to balance real-time responsiveness and resource efficiency.

The AI further enhances decision-making through anomaly detection and optimization recommendations. The system ensures real-time tracking and prevents frequent breaks during operations by continuously looping with brief delays, thereby balancing responsiveness and resource utilization. By integrating AI and IoT, this system not only ensures accurate inventory, but also enables operation across multiple locations without requiring enforcement.

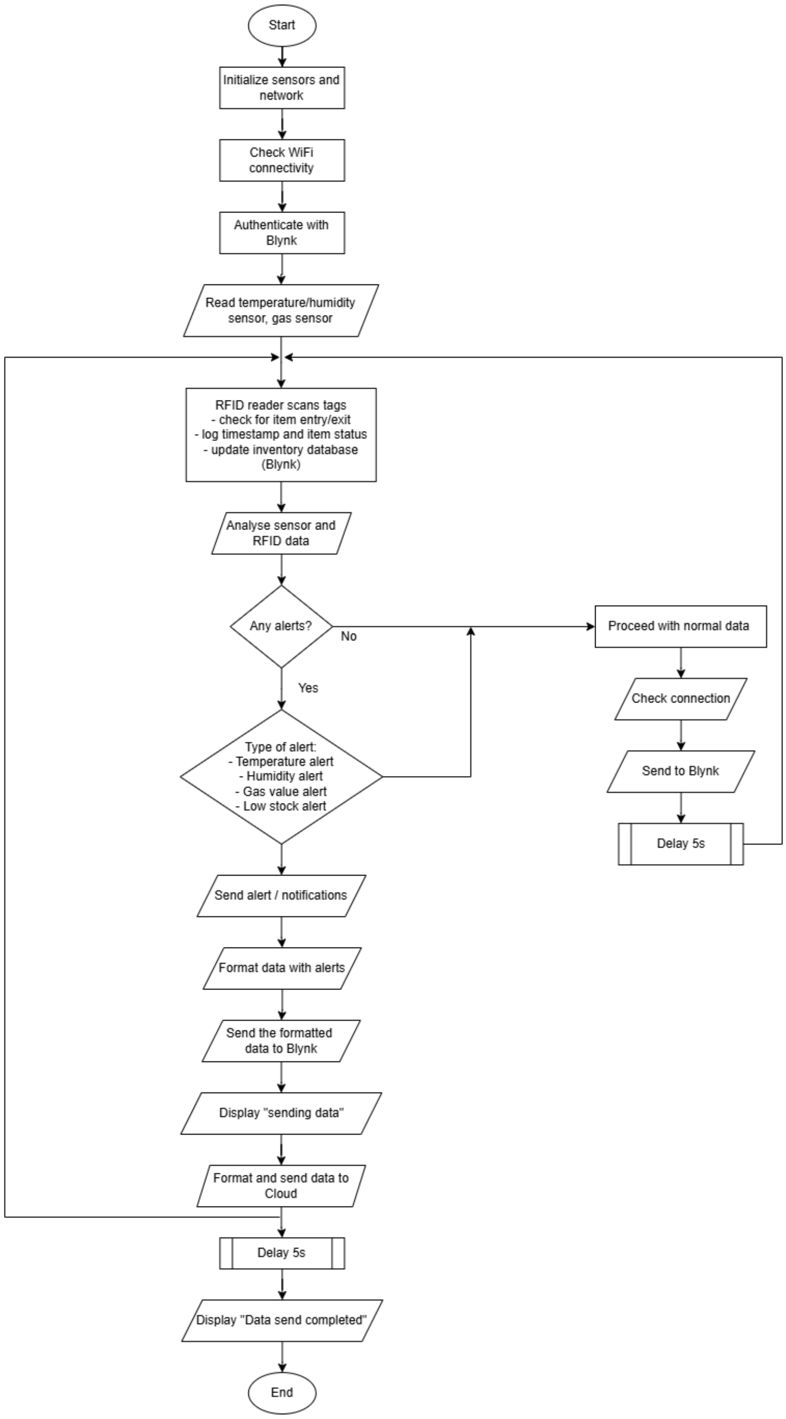


Fig. 2: *System Flowchart for Smart Warehouse Monitoring System*

3. Results, Analysis & Discussion

This section discusses the results obtained from the tests that have been carried out. The prototype and hardware have been assembled and developed. Also, the analysis has been done to discuss the functionality of the components. The experiments and troubleshooting for this system are discussed in an orderly. The result of the whole hardware prototype (refer to Figure 3). The details of the prototype will be discussed in the following subtopics.

Fig. 3: *Hardware Prototype*

3.1 Results

The implementation of the AI-powered Smart Warehouse Monitoring System yielded successful data collection across both primary subsystems. The NodeMCU ESP32 served as the main controller, capturing both environmental variables and inventory movements simultaneously. All results were displayed in real-time on the Blynk dashboard, confirming that the hardware and software integration was fully operational.

3.1.1 Part of Environmental Control

The environmental control results showed that the system could accurately track warehouse safety parameters. The DHT11 sensor provided consistent readings for temperature and humidity (refer to Figure 4 and Figure 5), while the MQ-2 sensor successfully measured gas concentrations (refer to Figure 6). Data was updated every 3000 ms, ensuring that any sudden changes in the warehouse atmosphere were captured immediately. For instance, testing showed stable temperature readings of around 32.4 °C and a humidity level of 57.00%. If the temperature rises above a safe limit, such as 35°C, or if gas levels reach hazardous thresholds, the system immediately triggers a notification to the user through the Blynk platform (refer to Figure 7).

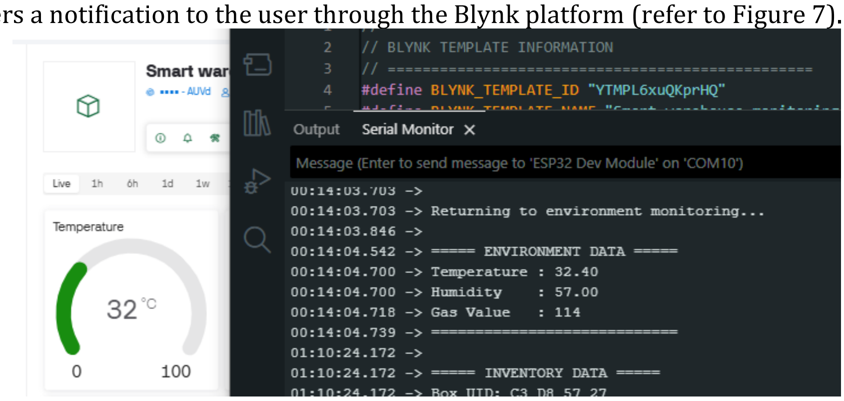


Fig. 4: *Temperature reading in Arduino IDE and synchronous to Blynk*

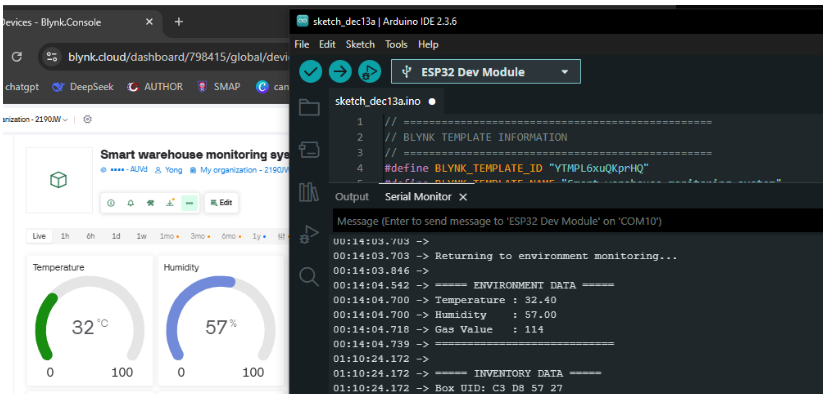


Fig. 5: *Humidity reading in Arduino IDE and synchronous to Blynk*

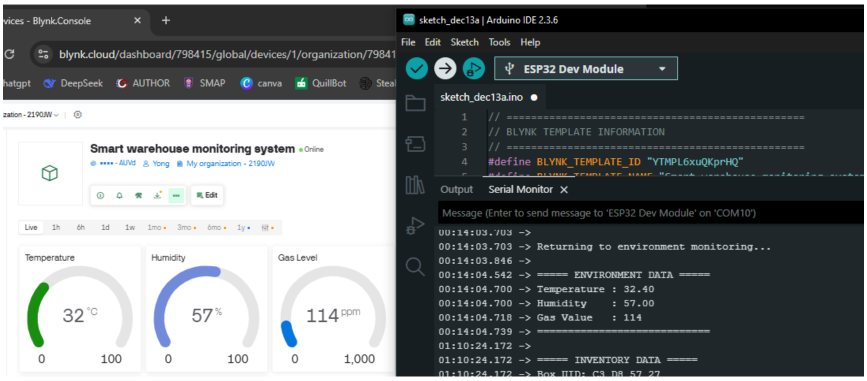


Fig. 6: *Gas Value reading in Arduino IDE and synchronous to Blynk*

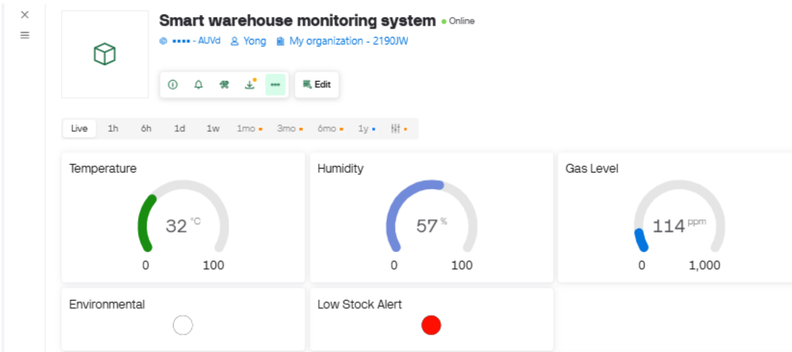
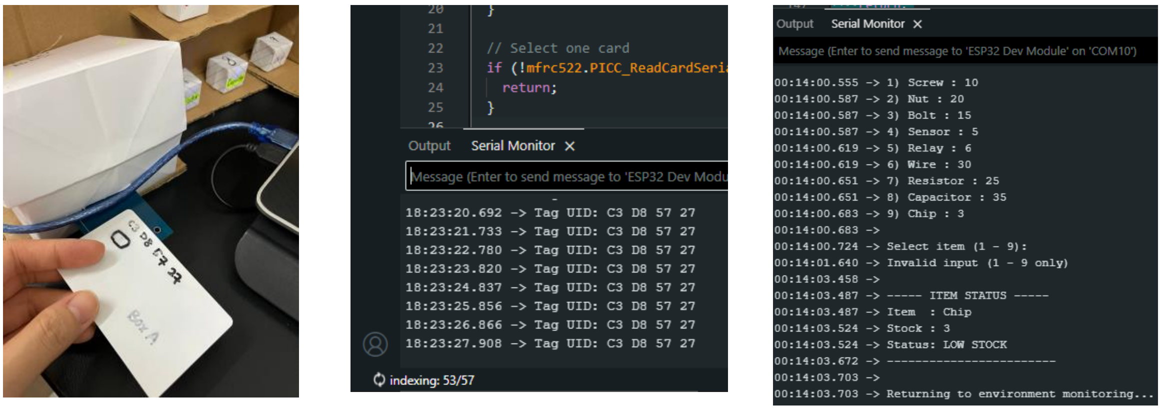


Fig. 7: *Environmental Control Alert System in Blynk*

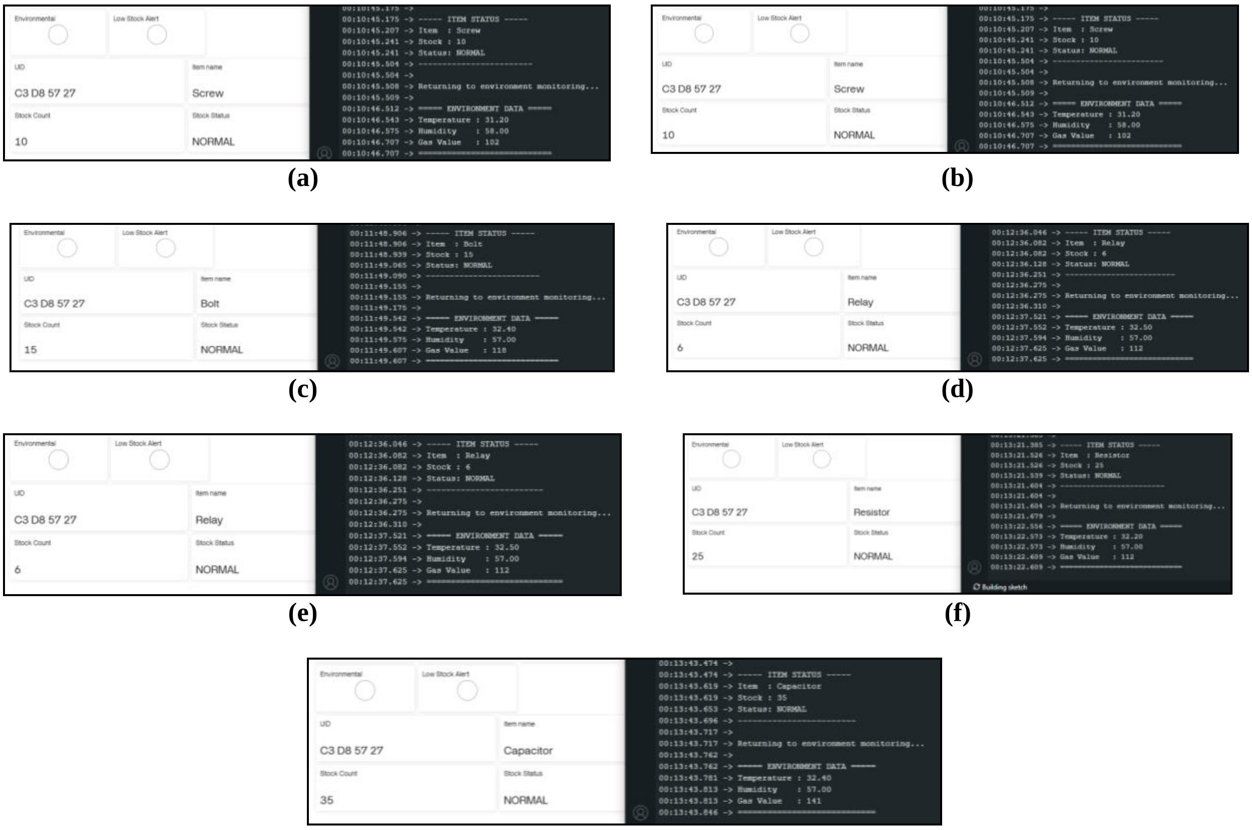
3.1.2 Part of Inventory Optimization

For inventory optimization, the RFID subsystem proved highly effective for real-time stock management. RFID technology enables automatic identification and real-time monitoring of inventory items, significantly reducing human error in warehouse operations [8]. Whenever an RFID tag was brought near the reader, the system identified its unique ID (UID) and updated the inventory database (refer to Figure 8). The system successfully tracked nine different items, including screws, nuts, bolts, and electronic chips, displaying their specific stock counts on the serial monitor and Blynk dashboard (refer to Figure 9). A critical part of these results was the "Low Stock Alert" feature; when an item's quantity dropped to 5 units or fewer, the dashboard indicator turned bright Red and displayed a "LOW STOCK" status (refer to Figure 10). This automated identification reduces the human error typically found in manual counting and significantly speeds up the tracking process.



(a) (b) (c)

Fig. 8: *(a) Inventory System activated when an RFID tag is scanned (b) The UID number printed in the Arduino IDE (c) Inventory Database updated in the Arduino IDE*



(a) (b) (c) (d) (e) (f) (g)

Fig. 9: Items that synchronize with the Arduino IDE and Blynk in the NORMAL condition

(a) Screw (b) Nut (c) Bolt (d) Relay (e) Wire (f) Resistor (g)Capacitor

(a) (b)

Fig. 10: Items that synchronize with the Arduino IDE and Blynk in the LOW STOCK condition

(a) Sensor (b) Chip

3.2 Functionality of Hardware Prototype

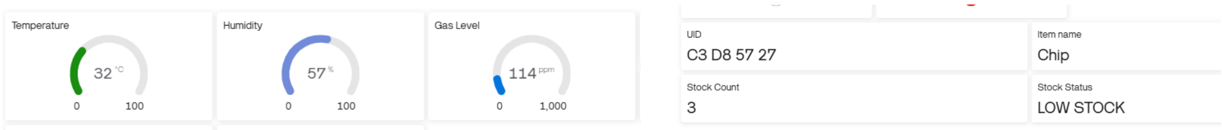
This section will cover the project's functionality tests and troubleshooting. To ensure that every component can function properly, tests have been conducted. Every component used in this project has its functionality shown in the figures below.

3.2.1 Functionality of All Components in Smart Warehouse Monitoring System

Each hardware component functioned exactly as intended within the integrated system. The NodeMCU ESP32 served as the brain, processing data from multiple sensors and maintaining a stable Wi-Fi connection for cloud communication. It was selected for its integrated Wi-Fi, low power consumption, and suitability for IoT applications [9]. The DHT11 and MQ-2 sensors provided the necessary environmental inputs, while the MFRC522 RFID reader handled the contactless identification of inventory items. All components were put in a custom-designed structure that produces a real warehouse environment, providing protection for the electronics while ensuring the sensors could effectively sample the surrounding air.

3.2.2 Functionality of Blynk Web Console

The Blynk web console served as the primary remote interface, enabling users to monitor warehouse conditions from any location with internet access. It successfully translated all sensor data into easy-to-read gauges for temperature and humidity, as well as text labels for inventory status (refer to Figure 11). The platform's most important function was its alerting system, which provided immediate visual warnings, such as the Red LED for low stock and environmental alert (refer to Figure 12). It ensures that problems can be addressed without a user needing to be physically present in the warehouse.



(a) **(b)**

Fig. 11: All data synchronized in Arduino IDE in Blynk *(a) Environmental Control data (b) Inventory Status*

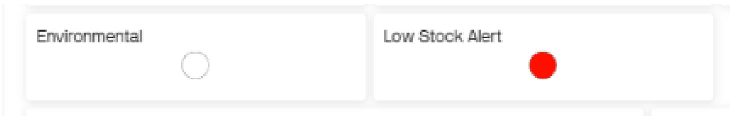


Fig. 13: *Environmental Alert System and Low Stock Alert System*

3.2.3 Environmental Control Graph

The environmental control graphs displayed on the Blynk dashboard provided a visual history of the warehouse conditions. These charts illustrate that the temperature, humidity, and gas levels remained stable over time, demonstrating that the ESP32 could process and transmit data without significant delays or crashes. By observing these trends, warehouse operators can identify patterns, such as gradual temperature increases, which may indicate environmental risks.

Fig. 14: Environmental Control Related Graph

3.3 Discussion

During the testing and development phase, several issues and solutions are identified to ensure the system has stable functionality. First, the unstable power supply issue was resolved by ensuring all components had sufficient and stable power and that all pins were correctly connected. Wiring errors have been corrected, particularly for RFID pins and ESP32 pins, by verifying the connections against the schematic. Bugs in the Blynk configuration settings were resolved by using a private Wi-Fi hotspot.

Problems with hardware in the Smart Warehouse Monitoring System include issues with RFID UIDs and software development. When detecting the card reader with an RFID sensor, the Arduino IDE cannot print the RFID's UID. A workable solution is to test the RFID sensor's usability by connecting the port to the ESP32 and running the code in the Arduino IDE. However, when the Arduino IDE indicates that the RFID is not connected to the ESP32, an old RFID has been replaced. Once the Serial Monitor in the Arduino IDE displays the UID number, the inventory database has been successfully displayed. Therefore, it shows that the recent research has demonstrated that deep learning techniques can enhance inventory optimization and anomaly detection by learning demand patterns from historical data [7].

Progress in Engineering Application and Technology Vol. 7 No. 1 (2026) p. 290-297

4. Conclusion

The AI-powered smart warehouse monitoring system successfully presented the modernization of traditional storage management through the integration of Internet of Things (IoT) hardware and intelligent monitoring. By utilizing a NodeMCU ESP32 microcontroller to coordinate data from DHT11 environmental sensors, MQ-2 air quality sensors, and RFID tracking modules, the project has established a reliable hardware prototype capable of protecting sensitive goods from spoilage and hazards. The system can identify inventory items through unique RFID tags and provide real-time updates to a remote Blynk web console represents a significant improvement over slow and error-prone manual stock-checking methods. This research demonstrates that a cost-effective combination of sensors and cloud-based dashboards can provide warehouse managers with constant visibility and automated alerts, even when they are not physically on-site. Ultimately, this project serves as a strong foundation for future advancements, such as the full integration of complex machine learning algorithms to predict stock needs and equipment failures before they occur, further maximizing the efficiency and safety of smart warehousing operations

Acknowledgement

The authors would like to thank the Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia for their support along the way.

References

[1] M. A. Jabbar, R. B. Madhavi, and B. L. Laxmi, "Warehouse Management System Using IoT and Artificial Intelligence," 2020 International Conference on Computer Science and Its Application in Agriculture (ICOSICA), pp. 1-6, 2020.

[2] M. K. Hasan, M. M. Islam, and M. R. Islam, "IoT-Based Warehouse Management System," Int. Conf. Cyber Resilience (ICCR), pp. 1-5, 2022.

[3] M. Ghazali, N. A. Ibrahim, and A. Ismail, "IoT-driven Smart Warehouses with Computer Vision for Enhancing Inventory Accuracy and Reducing Discrepancies in Automated Systems," IRE Journals, 2024.

[4] T. Y. Chai, S. C. Haw, M. Jahangir, K. B. Hoe, L. E. Heng, and M. Vellaisamy, "Advancing Retail Operations: A Customizable IoT-Based Smart Inventory System," International Journal of Membrane Science and Technology, vol. 10, no. 3, pp. 1885-1897, 2023.

[5] T. K. Sowmya, H. A. Vinutha, and M. P. Aruna, "Implementation of IoT based Smart Warehouse Monitoring System," International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET), vol. 6, no. 13, pp. 1-4, 2018.

[6] K. Dewi and R. N. Shofa, "Development of Warehouse Management System to Manage Warehouse Operations," International Journal of Applied Information Systems and Informatics (JAISI), vol. 1, no. 1, pp. 1-10, 2023.

[7] C. Deng and Y. Liu, "A Deep Learning-Based Inventory Management and Demand Prediction Optimization Method for Anomaly Detection," Wireless Communications and Mobile Computing, vol. 2021, no. 1, 11 pages, 2021.

[8] S. B. Erlangga, A. Yunita, and S. R. Satriana, "Development of Automatic Real Time Inventory Monitoring System using RFID Technology in Warehouse," JOIV: International Journal on Informatics Visualization, vol. 6, no. 3, pp. 636-641, 2022.

[9] Espressif Systems, "ESP32 Series Datasheet," 2023.