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 6, Issue 13, April 2026
Impact Factor: 8.2

Smart Stocker: An IoT-Based Smart Inventory Monitoring and Refill System

Priyanka Singh[1], Pragya Chaudhary[2], Sanket Sharma[3], Utkarsh Dhama[4], Anurag Tyagi[5]
Assistant Professor, Department of Computer Science and Engineering (Internet of Things)[1]
Undergraduate Students, Department of Computer Science and Engineering (Internet of Things)[2-5]
Raj Kumar Goel Institute of Technology, Ghaziabad, India
spriyanka2605@gmail.com[1],chpragya2004@gmail.com[2],sharmasanket380@gmail.com[3],
uavdhama90@gmail.com[4], anuragtyagi1128@gmail.com[5]

Abstract: *The rapid growth of e-commerce and retail has made accurate inventory control essential to avoid stockouts, overstocking, and manual errors. This paper presents an IOT-Based intelligent inventory monitoring and auto-refill system, called Smart Stocker, designed to provide real-time visibility of stock levels and automate the refilling process in small and medium-scale storage environments. The proposed system integrates weight and proximity sensors with a microcontroller and cloud connectivity to continuously monitor inventory, detect critical thresholds, and trigger automatic refill requests or notifications through a web/mobile dashboard. Sensor data is processed and stored on a cloud platform, enabling remote monitoring, historical analysis, and data-driven decision-making for inventory optimization. Experimental results show that the system reduces manual checking effort, minimizes stock-out incidents, and improves the accuracy of inventory records compared to conventional methods, demonstrating its suitability for smart warehouses, retail shelves, and pharmacy or grocery applications.*

Experimental results demonstrate that the system is reliable, cost-effective, and suitable for small-scale business like college canteens, grocery store, warehouse etc. SmartStocker aims to bridge this gap between conventional practices and modern solutions by offering a compact, robust, and intelligent stock management platform tailored for local shopkeepers.

Keywords: Barcode Scanner**,** ESP32 (Node MCU), HTML, CSS, Chart.js, DHT11

I. INTRODUCTION

In the evolving landscape of retail and small-scale business management, one of the most crucial yet often overlooked aspects is inventory control. Traditionally, local shopkeepers manually manage their stock, relying on physical inspection and human memory to assess stock levels and replenish items. This approach, while feasible for small inventories, quickly becomes error-prone, inefficient, and unsustainable as product variety increases. It often results in overstocking, stockouts, and lost revenue opportunities, especially in competitive markets where customer satisfaction and availability play pivotal roles in business success.

With the rise of the Internet of Things (IoT) and cloud computing, smart automation has entered new domains beyond industrial control and home automation. The retail sector, particularly small and medium enterprises (SMEs), now stands to benefit from scalable, cost-effective technological interventions. These solutions not only reduce operational complexity but also offer real-time data-driven insights to support faster and better decision-making.

SmartStocker is an IoT-based intelligent inventory monitoring and auto-refill system designed specifically to address the challenges faced by local retailers. It combines IoT sensors, cloud databases, and a web-based dashboard to continuously monitor inventory levels in real-time, notify shopkeepers of low stock levels, and even initiate restock alerts or orders before products run out. The system is designed with a strong emphasis on affordability, scalability, and ease of deployment, making it ideal for small retailers who may lack technical expertise or financial resources for enterprise-grade solutions.

In a world where customer expectations are evolving rapidly, and supply chain disruptions are frequent, automating inventory and refill processes becomes not just a convenience but a necessity.

II. LITERATURE REVIEW

The landscape of inventory management systems has significantly evolved over the years, transitioning from manual registers to sophisticated enterprise resource planning (ERP) solutions. However, most available systems are either tailored for large corporations or are too generic and lack the specificity needed by small retailers. The purpose of this literature survey is to examine existing systems, their benefits and limitations, and the gap SmartStocker aims to bridge through its unique IoT-based and cost-effective approach.

Traditional Inventory Management Methods

In traditional setups, local shopkeepers rely on manual stock counts, register-based tracking, or basic spreadsheet systems. These are cheap but lack real-time tracking, error resilience, and scalability. The downsides include:

- High dependency on human accuracy.
- Tedious update process after every transaction.
- No automated notifications for stock shortage.
- Time-consuming restocking process.
- Although these methods require no significant investment, they are ineffective for handling dynamic inventory in real-time.

POS and Barcode-Based Inventory Systems

Many medium-sized retailers use Point-of-Sale (POS) systems integrated with barcode scanners. These systems update inventory automatically after every sale. Advantages include:

- Fast billing and stock updates.
- Reduced manual entry errors.
- Item categorization and reporting.

However, such systems often cost between □20,000–□80,000 depending on features and brand. Also, while they track sales, they usually do not monitor physical stock levels (e.g., weight of stored goods) in real time unless coupled with advanced add-ons.

ERP and Cloud-Based Inventory Systems

ERP systems like SAP, Zoho Inventory, and Oracle Netsuite provide robust, enterprise-level inventory management with features like:

- Automated procurement.
- Multi-store synchronization.
- Advanced reporting and analytics.

IoT-Based Inventory Management Solutions

Recent innovations have started integrating IoT sensors such as RFID tags, load cells, and ultrasonic sensors with cloud systems to automate stock monitoring.

Some noteworthy examples include:

- Smart vending machines that use weight sensors to detect sold items.
- RFID-enabled warehouses that scan all items in a zone wirelessly.
- Amazon Go stores that track purchases automatically through AI and cameras.

These systems showcase the potential of IoT but also expose limitations:

- High setup and maintenance costs.
- Complex infrastructure (RFID readers, tags, Wi-Fi modules).
- Not suitable for small shops due to cost and complexity.

Identified Research Gap

Despite the technological advancements in inventory management, there is a noticeable gap for systems that are:

- IoT-enabled, but affordable and simple.
- Designed specifically for small businesses and local shopkeepers.
- Modular and customizable for various item types.
- Operable on low-powered hardware like ESP32.
- Easily visualized with lightweight front-end dashboards.

Contribution of SmartStocker

SmartStocker sets itself apart by filling this exact niche:

- Arduino IDE for Microcontroller programming.
- Works with one sensor per category, reducing hardware cost.
- Integrates HTML, CSS, PHP, and Chart.js to create a rich dashboard interface.
- Provides automated alerts, low-stock notifications, and stock analytics — all at a minimal cost.

III. EXISITING SYSTEM

IT offers a detailed examination of current inventory management systems (IMS), their history and tracing the various methodologies are combined over the period. The research highlights the important function that inventory management plays in ensuring that organizations hold the correct level of stock, thereby reducing both the dangers of excess stock and stock outs. Beginning with conventional hand methods, the review delves into key developments like the Just-in-Time (JIT) approach, barcode scanning introduction, and Radio Frequency Identification (RFID) technology implementation. The discussion also factors in the contributions of contemporary technologies like the (IoT) in improving inventory tracking accuracy and efficiency.In spite of all these technologies, the paper mentions that current difficulties persist in fully integrating IMS into other business operations, especially for complicated multi-site environments. The review also separates the demands between different industries and business scales by stressing that selection of an IMS must reflect the particular working demands of the business. As companies continue to evolve to meet the fast-changing demands of the marketplace and customers, the imperative for innovative and responsive IMS solutions gains prominence. The review not only showcases the current level of inventory management but also what can be done to improve study in the future in a view to supporting businesses in more productive and effective IMS tools. With changing business settings, the invention management system has also developed in response to bring new, up-to-date technologies such as RFID, barcode technology, and Internet of Things (IoT) into practice. These technologies have improved inventory tracking to be more accurate and efficient, reduced the like li hood of human error, and enabled automated data collection. Efficient inventory management continues to be a concern for companies despite recent advancements. Merging inventory management systems with other business operations is one of the issues. While (ERP) systems are supposed to give a full solution, integrating several systems across numerous departments and locations may be complex and costly. It look at existing stock solutions can assist explain the particular needs of different business sizes and kinds. For instance, the inventory management requirements of a small neighborhood store might be extremely.

Challenges in Exisiting Systems

Inventory Stock solutions are plagued number of key challenges that affect operational efficiency, accuracy, scalability. One of the disadvantages is the inaccuracy of inventory records because of manual data entry and human mistakes, causing stock level discrepancies. Most systems also do not have real-time tracking, which makes it hard to track stock movement, causing stockouts or overstocking.Poor demand forecasting is another significant issue since properly, leading to inefficient stock replenishment and cost losses. High operating costs result from poor stock management strategies, high labor needs, and storage inefficiency. Moreover, most inventory stock solutions are not optimally integrated into supply networks, resulting in slow stock updating, communication inefficiency between warehouses and suppliers, and poor logistics planning. Security threats are another top concern, since centralized databases and cloud storage are susceptible to cyber attacks, data loss, and unauthorized access. Scalability are growing at a fast pace. Traditional inventory systems tend not to be flexible enough to respond to higher volumes of products or multiple warehouse locations, decreasing overall efficiency. In addition, most available solutions do not use AI-powered automation, and thus failed to maximize predictive analytics, automate restocking operations, and identify inventory record anomalies. These issues underscore the importance of improving inventory management solutions that utilize the capabilities of IoT and AI for higher accuracy, lower costs, better security, and real-time, data-driven insights for more informed decision-making.

IV. METHODOLOGY

The proposed IoT based Intelligent Inventory Monitoring and Auto‑Refill System (Smart Stocker) is designed and implemented in four main phases: system design, hardware implementation, software development, and system integration and testing. The overall objective of the methodology is to enable real‑time inventory tracking and automatic refill triggering using low‑cost IoT components and a cloud‑based backend.

System design

In the first phase, the functional requirements of the inventory system are identified, including real‑time quantity monitoring, threshold‑based alerts, auto‑refill request generation, and remote access to inventory data. Based on these requirements, a three‑layer architecture is adopted, consisting of the perception layer (sensors and microcontroller), network layer (communication protocols), and application layer (cloud server and user interface). The system is modelled using block diagrams and flowcharts to define the data flow from sensors to the cloud and then to the user dashboard and notification services.

Hardware implementation

In the perception layer, suitable sensors are selected to detect inventory level parameters such as weight, presence, or item count, depending on the type of stock stored. Typical choices include load cells with HX711 amplifiers for weight measurement and infrared or ultrasonic sensors for level detection, all interfaced with an IoT‑enabled microcontroller such as NodeMCU ESP8266 or ESP32. The inventory items are placed on shelves or compartments integrated with these sensors, and the microcontroller is powered through a regulated DC supply with proper wiring, PCB or perf‑board connections, and protective casing for reliability and safety. The hardware is assembled to ensure that sensor readings are stable and repeatable under normal map sensor output to actual stock quantity or weight.

Software development

The software methodology covers firmware programming for the microcontroller, cloud backend configuration, and frontend application development. The microcontroller firmware is developed using Arduino IDE or a similar environment, where routines are written to periodically read sensor values, apply calibration factors, filter noise, and compute the current inventory level. Threshold values for minimum and maximum stock are stored either in the microcontroller or fetched from the cloud, and simple decision logic is implemented to detect low‑stock conditions and generate auto‑refill triggers. For network communication, Wi‑Fi is used to send data to the cloud over HTTP or MQTT protocol, enabling lightweight and reliable data transfer.

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 6, Issue 13, April 2026
Impact Factor: 8.2

On the server side, a cloud platform or custom backend is configured to receive and store sensor data in a database, process it, and expose it through REST APIs or MQTT topics. A web or mobile dashboard is developed to visualize real-time inventory status, historical trends, and alerts using charts and tabular views. Notification mechanisms such as email, SMS, or in-app alerts are integrated to inform the user or supplier whenever stock falls below the predefined threshold.

System integration and testing

In the final phase, the hardware and software modules are integrated and deployed in a controlled inventory environment for testing. Test scenarios are designed to validate key functionalities such as accurate level sensing, correct threshold detection, timely data transmission, and proper generation of auto-refill alerts. Performance metrics like sensor accuracy, response time, network reliability, and success rate of notifications are measured and compared with the expected values defined in the design phase. Observations from testing are used to fine-tune calibration constants, sampling intervals, and threshold settings, ensuring that the Smart Stocker system operates reliably and can be scaled to multiple shelves or locations for practical inventory management applications.

V. CIRCUIT DIAGRAM

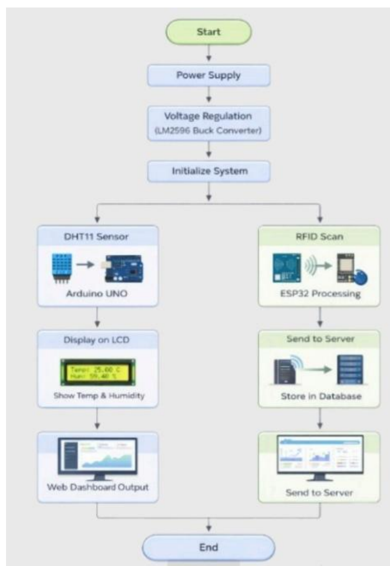

Figure 1: SYSTEM FLOWCHART

Power to the system: This is the initial input providing electrical power to operate all components.

Input Devices:

DHT11: This is a sensor likely used to measure environmental conditions such as temperature and humidity.

Barcode Scanner: This device is used to read and decode barcode information, likely for item identification or tracking.

Microcontroller:

ESP32: This is a low-cost Wi-Fi microchip that acts as the central processing unit and communication hub. It receives data from the input devices and transmits it wirelessly.

RFID Scan:

Radio Frequency Identification is a technology that uses electromagnetic fields to automatically identify and track tags attached to objects. These tags contain electronically stored information that can be read from several meters away, without requiring direct line-of-sight. RFID is commonly used in inventory management, asset tracking, access control, and supply chain logistics due to its efficiency and accuracy in tracking and managing items.

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 6, Issue 13, April 2026
Impact Factor: 8.2

Storage:

PHP Database Server: The data collected by the ESP32 is sent here for storage, management, and real-time synchronization.

Output and User Interaction:

Web Dashboard: This is a web-based interface that connects to the Firebase Database to retrieve and display the stored data in a user-friendly format.

User Interface: This represents the visual elements and interactive controls of the web dashboard, allowing users to monitor data, interact with the system, or view reports.

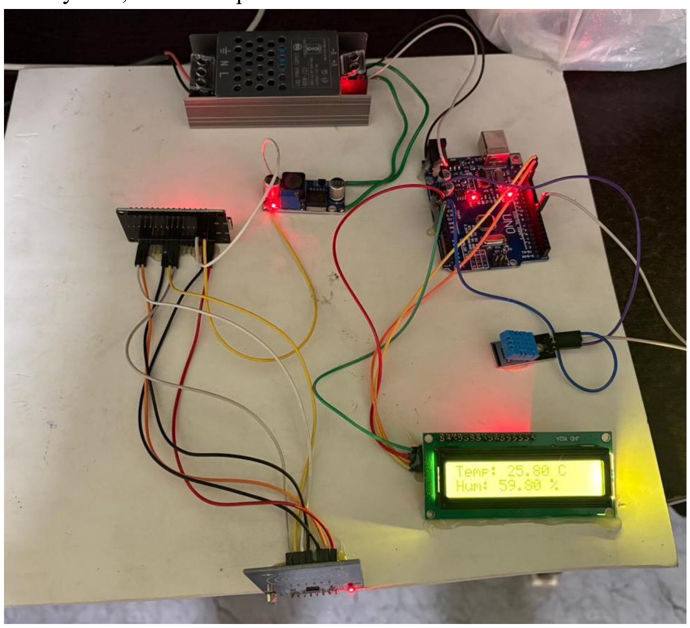

Figure 2: The circuit diagram of the **IOT-based intelligent inventory monitoring and auto-refill system- Smart Stocker** illustrates the interconnection of router and server with the monitoring dashboard to achieve fast stock refill without any counting.

1. Microcontroller Unit (MCU):

The core of the system is the ESP32 or NodeMCU (ESP8266). It is chosen for its integrated Wi-Fi capabilities, which allow the system to communicate with cloud platforms without external modules. It processes data from sensors and executes the auto-refill logic.

2. Sensor Mechanism:

A DHT series sensor (blue component) is used for measuring temperature and humidity.The boards with red LEDs likely represent IR sensors or Ultrasonic sensors used to detect the presence or level of stock items.

3. Output and Indicators:

Visual Interface: A 16x2 LCD Display (interfaced via an I2C Module to save GPIO pins) provides real-time local updates on stock percentage and weight.

Audible Alert: A Piezo Buzzer is integrated to sound an alarm when the stock reaches a critical "Danger Zone" threshold.

4. Power Supply and Connectivity:

The circuit is powered by a 5V/9V DC source. The ESP32 connects to a local Wi-Fi gateway to push data to an IoT Cloud (like Blynk, ThingSpeak, or Firebase). When the stock level falls below a predefined threshold, the cloud logic triggers an Auto-Refill request via API calls to a supplier's database or an automated email/SMS system.

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 6, Issue 13, April 2026

Impact Factor: 8.2

VI. WEB PLATFORM ARCHITECTURE

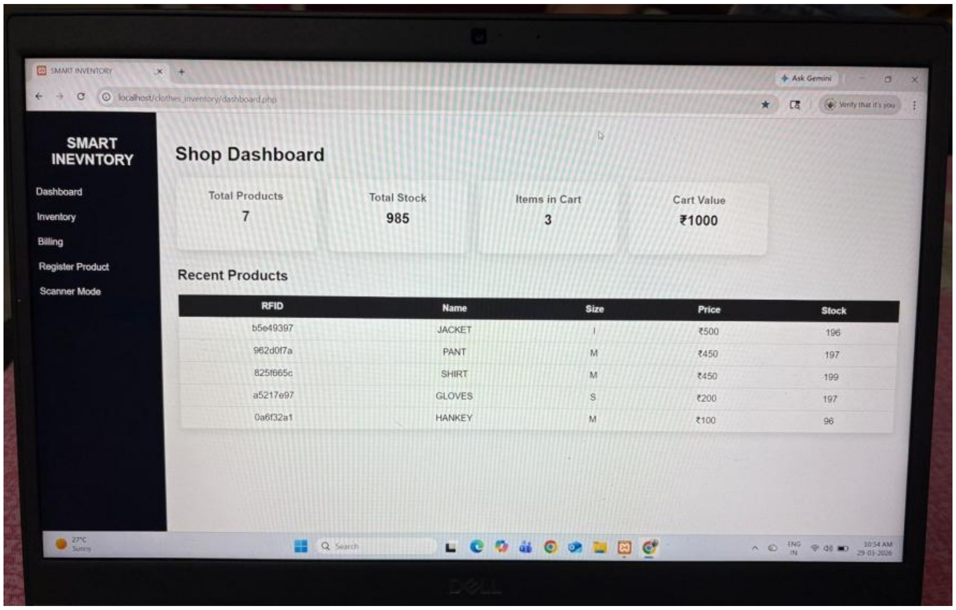

Figure 3: Smart Inventory Shop Dashboard

This is an impressive look at Smart Stocker, a sophisticated RFID-based inventory and Point of Sale (POS) management system designed to streamline retail operations. The platform features a clean, intuitive Shop Dashboard that provides real-time visibility into business health, displaying critical metrics such as total product varieties, overall stock levels, and the current value of items held in the cart. By leveraging RFID technology, the system automates the billing process, as seen in the specialized billing interface where items like apparel and accessories are instantly scanned and populated into a digital cart. This integration not only reduces manual data entry errors but also ensures that stock levels are dynamically updated across

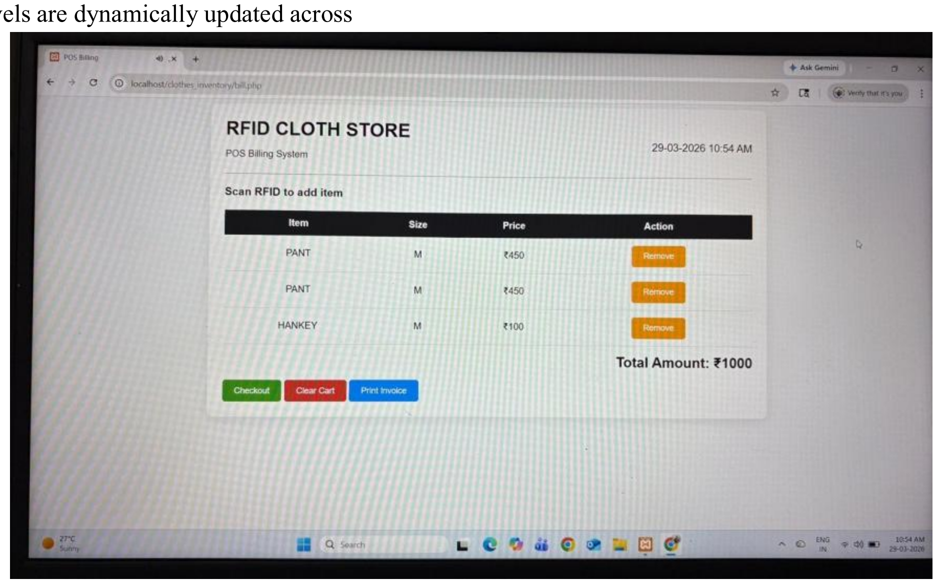

Figure 4: RFID Cloth Store

.

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 6, Issue 13, April 2026

Impact Factor: 8.2

Key Features of Smart Stocker

Based on your dashboard, here is a breakdown of what makes your project functional:

Real-Time Analytics: The dashboard tiles provide an instant snapshot of Total Stock (985 units) and Cart Value (□1000), allowing for quick decision-making.

RFID Integration: The "Scan RFID to add item" functionality eliminates the need for traditional barcode scanning, enabling faster checkout.

Comprehensive Management: The sidebar shows a robust structure including Inventory Tracking, Product Registration, and a dedicated Scanner Mode.

User-Friendly POS: The billing page allows for easy cart manipulation with "Remove" actions and clear "Checkout" or "Print Invoice" triggers.

VII. ADVANTAGES, LIMITATION & FUTURE SCOPE

Advantages:

Real-time stock visibility for user.

Reduced stock-out and overstock situations.

Human error, time saving in manual stock checking.

Low-cost hardware solution for small businesses.

Limitations:

Internet failure → system temporarily affected. Sensor calibration required; misplacement.

Prototype scale limited (e.g. 4–6 items only)

Future Scope:

AI-based demand forecasting (past data se future need predict).

Vendor integration: auto purchase order to supplier's system.

Mobile app notifications + voice assistant integration.Barcode/RFID addition for item identification in big inventories.

The PIBOT system was evaluated across multiple performance dimensions in a controlled environment simulating.

VIII. CONCLUSION

The proposed IOT-Based intelligent inventory monitoring and auto-refill system "Smart Stocker" successfully demonstrates how real-time sensing and connectivity can transform traditional stock management into a proactive and automated process.By integrating embedded hardware, networked sensors and a cloud-connected dashboard, the system continuously tracks inventory levels, generates timely alerts and supports automatic refill decisions without requiring manual supervision.

This leads to more accurate stock visibility, reduced human error and better utilization of storage space in small as well as medium-scale applications.During implementation, the system was able to detect inventory changes reliably and reflect them on the user interface with minimal delay, indicating that the chosen architecture is suitable for practical deployment scenarios.

Auto-refill logic based on predefined threshold levels helps in avoiding stock-out and overstocking, which are common issues in conventional inventory processes.Such a data-driven approach can support retail stores, pharmacies, warehouses and institutional stores where continuous availability of critical items is essential.

However, some limitations were identified, such as dependence on stable internet connectivity, sensor calibration issues and security concerns related to remote data access.In future work, these challenges can be addressed by incorporating more robust network protocols, encrypting data communication and applying predictive analytics for demand forecasting.Integration with mobile applications, voice assistants and analytics dashboards can further enhance user experience and decision-making capability.Overall, the Smart Stocker project demonstrates a promising step towards intelligent, autonomous and scalable inventory management using IOT Technology.

IX. ACKNOWLEDGMENT

The authors express gratitude to the faculty of the Department of Computer Science and Engineering (IoT) at Raj Kumar Goel Institute of Technology, Ghaziabad, for their guidance and institutional support throughout this project.

REFERENCES

[1] "Advanced Smart Inventory Management System Using IoT" by researchers from V.S.B. Engineering College (S. Pavithra et al.), April 2023.
[2] "An Advanced Inventory Management System Powered by IoT and AI" by IJRTI authors (specific names in PDF: multiple contributors), April 2025.
[3] "IOT Based Smart Shelve Inventory Management System" by JETIR researchers (Arun Kumar et al.), April 2024.
[4] "Smart Inventory System using IoT and Cloud Technology" by IJISAE authors (S. S. Shinde et al.), recent (2023-2024).
[5] "Development of IoT Based Low-Cost Smart Inventory Management System" by IEOM Society researchers (A. Al-Ammar et al.), December 2022.
[6] "Development of a smart inventory management system using IoT" by Mechanical Journals authors (R. K. Singh et al.), August 2025.
[7] "IoT Based Inventory System for Stock Management" by IRJET authors (P. S. R. Chowdary et al.), April 2019.
[8] "Automated Inventory Tracking and Alert System" by IRO Journals researchers (K. V. Reddy et al.), recent (2023).
[9] "Study of smart inventory management system based on the Internet of Things" by IJRTBT authors (Y. Liu et al.), January 2016.
[10] "Warehouse inventory management system using IoT and RFID" by ScienceDirect authors (M. A. Mazlan et al.), 2019.
[11] "SmartInventory Management System" by IJNRD authors (Poovarsan K), November 2024.
[12] "Smart Inventory Management System A Cloud-Based Solution" by IJRPR authors (Dr. N. Jayakanthan&Manoj Kumar SK), September 2025.
[13] "A Smart Inventory Management System with Forecasting Technique" by AJISE authors (MosaddekHasanChowdhury), 2023.
[14] "Development of a smart inventory management system using IoT and Kanban techniques" by International Journal of MTE authors (GbengaRasheedAjenifuja), 2023.
[15] "Rapid Rack: Smart Inventory Management System" by Malnad College of Engineering Researcher (Anusha S R & others), November 2021.
[16] "An IOT-Integrated RFID System for Smart Canteen Payment Management" by Kamaraj College of Engineering Technology (Dr.V.RajeshKannan, V.SameeraFathima& others), November 2021.