Volume 10, Issue 4, April – 2025 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25apr681

AI for Inventory Management

G Punith Sai[1]; G Nagavallika[2]; A V S Sai Babu[3]; A Satish[4]; Y Vinay Kumar[5]; P Sunny Jaswanth[6]; Ch Venkatesh[7]

[2] Asst. Professor
[1,3,4,5,6,7] B.Tech Student

[1,2,3,4,5,6,7] Sri Vasavi Engineering College(A), Pedatadepalli, Tadepalligudem

Publication Date: 2025/04/23

Abstract: This project involves the development of an AI-driven inventory management system designed to simplify stock tracking and restocking for small businesses. It combines traditional inventory methods for products with stable demand and a machine learning model to predict restocking needs for items with fluctuating demand. The machine learning model is pre-trained on standard datasets, ensuring accurate forecasts without requiring training from user data.

Developed using Django, MySQL, and Bootstrap, the system is web-based and accessible from any device. Key features include vendor management, automated restocking alerts via email, and a billing module for managing in-store sales. Users can categorize products, track stock levels in real time, and view a dashboard that highlights low-stock items. With a user-friendly interface and intelligent automation, this system supports small business owners in making efficient, data-driven decisions.

How to Cite: G Punith Sai; G Nagavallika; A V S Sai Babu; A Satish; Y Vinay Kumar; P Sunny Jaswanth; Ch Venkatesh (2025), AI for Inventory Management. *International Journal of Innovative Science and Research Technology*, 10(4), 1027-1033. https://doi.org/10.38124/ijisrt/25apr681

I. INTRODUCTION

Managing inventory efficiently is vital for business success, especially for small businesses that must maintain the right balance between stock availability and cost control. Traditional inventory systems often rely on manual methods or basic tools that lack intelligent forecasting, which can lead to stock shortages or overstock situations.

This project proposes a smart inventory management system powered by artificial intelligence. It combines machine learning techniques to anticipate restocking needs for products with unpredictable demand and uses standard tracking for items with steady demand. Key functionalities include live inventory tracking, supplier management, billing integration, and automated restock alerts via email. The system is developed using Django, MySQL, and Bootstrap, offering a responsive and accessible platform for small business users.By utilizing predictive analytics, the system helps reduce inventory waste, supports better decision-making, and boosts overall operational productivity, leading to improved cost efficiency and enhanced customer satisfaction.

II. LITERATURE SURVEY

- *Shamita Deshmukh (2022) – Inventory Management System*

This work presents a basic yet functional inventory management system designed to simplify stock control for businesses. Developed using web technologies such as PHP, MySQL, and JavaScript, the system includes features like product entry, editing, and deletion, alongside inventory classification using ABC analysis and JIT principles. Its objective is to minimize manual errors and provide an adaptable platform that can be enhanced with additional features depending on business needs.

- *J.B. Munyaka & V.S.S. Yadavalli (2022) – Inventory Management Concepts and Implementations: A Systematic Review*

The authors conducted a detailed review of traditional and advanced inventory strategies. They examined models such as EOQ, MRP, JIT, ERP, and DRP, highlighting how these techniques can improve supply chain operations. The study advocates for blending classic inventory practices with modern tools to improve responsiveness, reduce inventory costs, and support better service delivery. Future improvements, the authors suggest, lie in adopting AI and real-time systems.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25apr681

➢ *Osman Çaylı & Zeki Oralhan (2024) – Artificial Intelligence-Driven Inventory Management*

This study investigates the use of artificial intelligence to optimize inventory operations. It incorporates various machine learning methods—LSTM for forecasting demand patterns, GBM for improving inventory planning, SVM for stock grouping, and RL for dynamic decision-making. The findings reveal that AI integration can significantly benefit small and medium enterprises by improving accuracy, reducing losses, and adapting to real-time business environments.

III. METHODOLOGY

➢ *Requirement Analysis:*

The initial phase involves collecting and analyzing the needs of end-users and defining the scope of the system to ensure it aligns with small business inventory workflows.

➢ *System Architecture Design:*

A structured layout is planned, detailing the interaction between the frontend (user interface), backend (logic and processing), and MySQL database. The design includes modules for real-time tracking, vendor details, billing records, and prediction outputs.

➢ *Machine Learning Integration:*

A Gradient Boosting algorithm is implemented to forecast inventory requirements. This model is trained on historical product movement data to predict future demand, especially for items with irregular sales patterns.

➢ *Testing and Validation:*

The system undergoes thorough testing at various levels, including component testing, integration checks, and complete system evaluation to confirm that all features operate correctly and reliably.

➢ *System Deployment:*

Once tested, the application is deployed to a web-based environment, making it accessible from different devices. Security measures and performance optimization are also implemented for a smooth user experience.

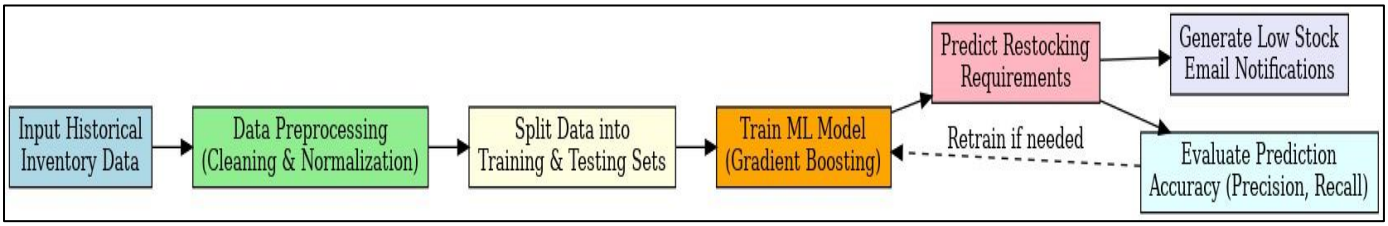

Fig 1 System Architecture

➢ *Data Acquisition and Preparation*

In the initial stage, inventory-related data such as item names, stock quantities, sales history, and vendor details are gathered. The dataset is then cleaned by eliminating missing, redundant, or incorrect entries. Important attributes are selected, and normalization is applied to ensure consistent scaling across all values, which improves model accuracy.

➢ *Splitting Data for Training and Testing*

To evaluate model performance effectively, the refined dataset is divided into two parts: one for training the machine learning model and the other for testing it. Generally, 75% of the data is used for training, while the remaining 25% is used to validate the model's predictions.

➢ *Forecasting Using Gradient Boosting Algorithm*

A Gradient Boosting algorithm is employed due to its strong predictive capabilities. It analyzes historical stock patterns to predict when a particular product is likely to run low. This allows the system to generate timely restocking alerts, especially for items with inconsistent demand.

➢ *Generating Inventory Predictions*

Once trained, the machine learning model evaluates current inventory data to forecast future stock requirements. This helps ensure optimal stock levels are maintained, minimizing both shortages and surplus inventory, which in turn supports better operational planning.

➢ *Model Accuracy Assessment*

To confirm the effectiveness of the predictions, the system uses performance evaluation metrics like RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error). These metrics help ensure the model's reliability before deploying it for real-time decision-making.

➢ *Machine Learning Model – Gradient Boosting (For Inventory Prediction)*

Gradient Boosting is used in the project to forecast future stock requirements based on historical inventory and sales trends. It is a sequential ensemble learning technique where each new model attempts to minimize the errors made by the previous one. This layered improvement process makes it highly effective in learning complex patterns in fluctuating demand.

How Gradient Boosting Works in This System:

- Initial Estimate: The model begins with a basic prediction, typically the mean stock level or average sales from past records.
- Error Identification: It then calculates the difference between the actual values and the model's current prediction, known as the residuals.
- New Tree Training: A new decision tree is trained to predict these residuals. This tree focuses on where the previous prediction was inaccurate.
- Model Update: The new tree's output is combined with

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25apr681

the earlier prediction to make a more accurate estimate.
- Final Output: After repeating this process several times, the combined output of all models results in a strong, accurate prediction of when and how much stock needs replenishing.

IV. EXISTING SYSTEM

➢ *Lack of Demand Forecasting:*

Most traditional inventory systems are reactive rather than predictive. They only track current stock levels and past sales, but cannot anticipate future demand, especially for products with seasonal or fluctuating sales. This often results in either excess inventory, which increases holding costs, or stockouts, which lead to missed sales and dissatisfied customers.

➢ *Manual Data Entry and Human Errors:*

A large number of small businesses continue to depend on manual methods or legacy software for tracking inventory. These methods are highly susceptible to errors such as inaccurate stock quantities, repeated entries, or untimely updates. Such mistakes can negatively impact decision-making, cause financial setbacks, and lower confidence in the system's reliability.

➢ *Absence of Real-Time Inventory Tracking:*

In many existing systems, stock levels are updated only after certain transactions or at the end of the day. Without real-time tracking, it becomes difficult for business owners to monitor stock status instantly, which is critical for fast-moving products. Delayed updates can disrupt order fulfillment and customer satisfaction.

➢ *No Automated Notifications or Alerts:*

Older systems often lack the ability to send automatic alerts for low-stock, out-of-stock, or expiring products. This means store owners must constantly check inventory levels manually, increasing the chance of missing important restocking opportunities. This inefficiency can result in lost sales or excess unused stock**.**

➢ *Limited Integration with Other Business Operations:*

Inventory systems that do not integrate with billing, supplier management, or analytics tools force businesses to switch between multiple platforms. This disconnect causes workflow interruptions, data inconsistencies, and added workload. An integrated system is essential for streamlining operations and maintaining accurate records across all departments.

V. PROPOSED SYSTEM

The proposed system introduces an AI-integrated solution for inventory management, specifically tailored to the needs of small businesses. It blends conventional stock tracking methods with intelligent demand forecasting, ensuring efficient handling of products with both predictable and unpredictable sales patterns. A pre-trained machine learning model is embedded to forecast restocking needs for items with variable demand, removing the need for users to train the model manually.

Developed as a web-based platform using Django, MySQL, and Bootstrap, the system provides seamless access across devices. It offers essential features including supplier management, live stock level updates, and an inbuilt billing system for managing retail transactions. Additionally, it sends automated email alerts when stock levels drop below a predefined threshold and presents a user-friendly dashboard highlighting inventory status.By automating key inventory tasks and offering data-driven insights, the system supports small business owners in maintaining optimal stock levels, reducing manual workload, and improving decision-making.

VI. IMPLEMENTATION

➢ *Step 1: Setting Up Your Development Environment*

Install required packages using requirements.txt (Django, MySQL, scikit-learn, joblib). Configure the MySQL database, and create Django apps for inventory, vendors, billing, and ML.

➢ *Step 2: Designing the Web Interface*

Create a user-friendly, responsive web interface using Django templates and Bootstrap. Design forms for adding products, vendors, sales, and navigation between modules. Develop a dashboard to display real-time stock updates and predictions.

➢ *Step 3: Integrating Machine Learning for Restocking*

Train a predefined ML model using historical product data to predict restocking needs. Integrate the trained model into the system to run predictions when inventory is updated. Show restocking suggestions on the dashboard to assist in planning.

➢ *Step 4: Managing Vendors and Billing*

Allow users to manage vendor information and link them to specific products. Enable in-store billing with stock adjustment after every sale. Store and display billing history for tracking sales trends.

➢ *Step 5: Sending Automated Email Alerts*

Set up automatic email notifications for low-stock items. Trigger alerts when product quantity falls below the reorder level. Include product and vendor details in the email for quick restocking.

➢ *Step 6: Deployment*

Run the project locally for final testing.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25apr681

VII. RESULT AND ANALYSIS

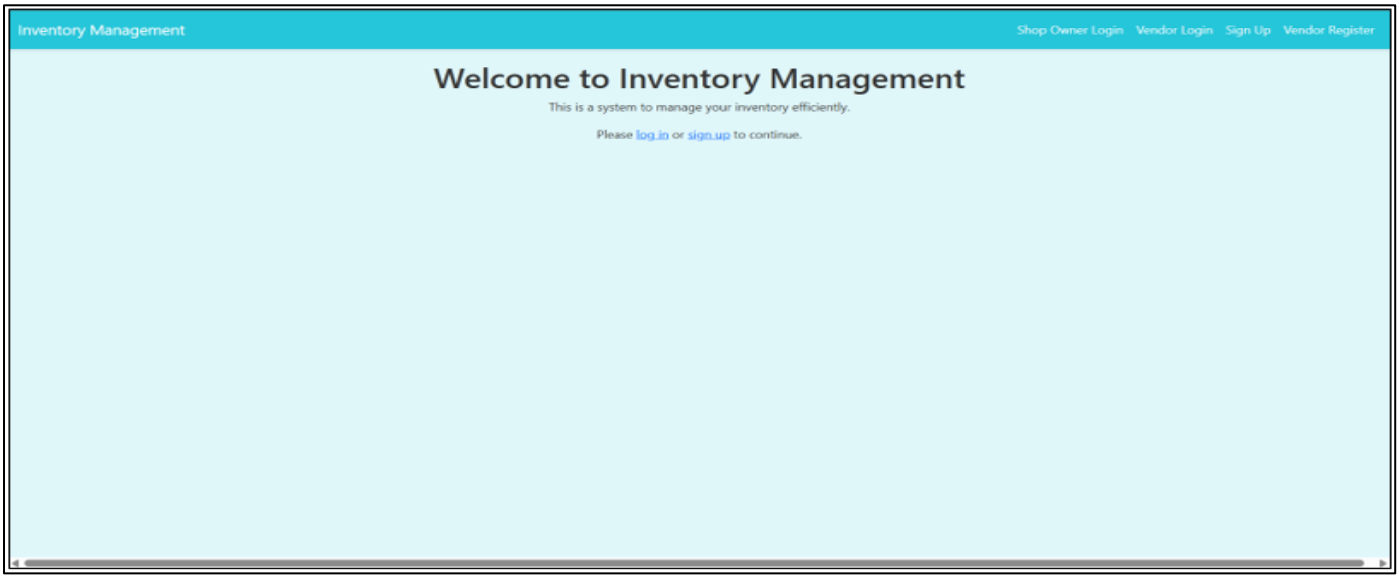

Fig 2 Home Page

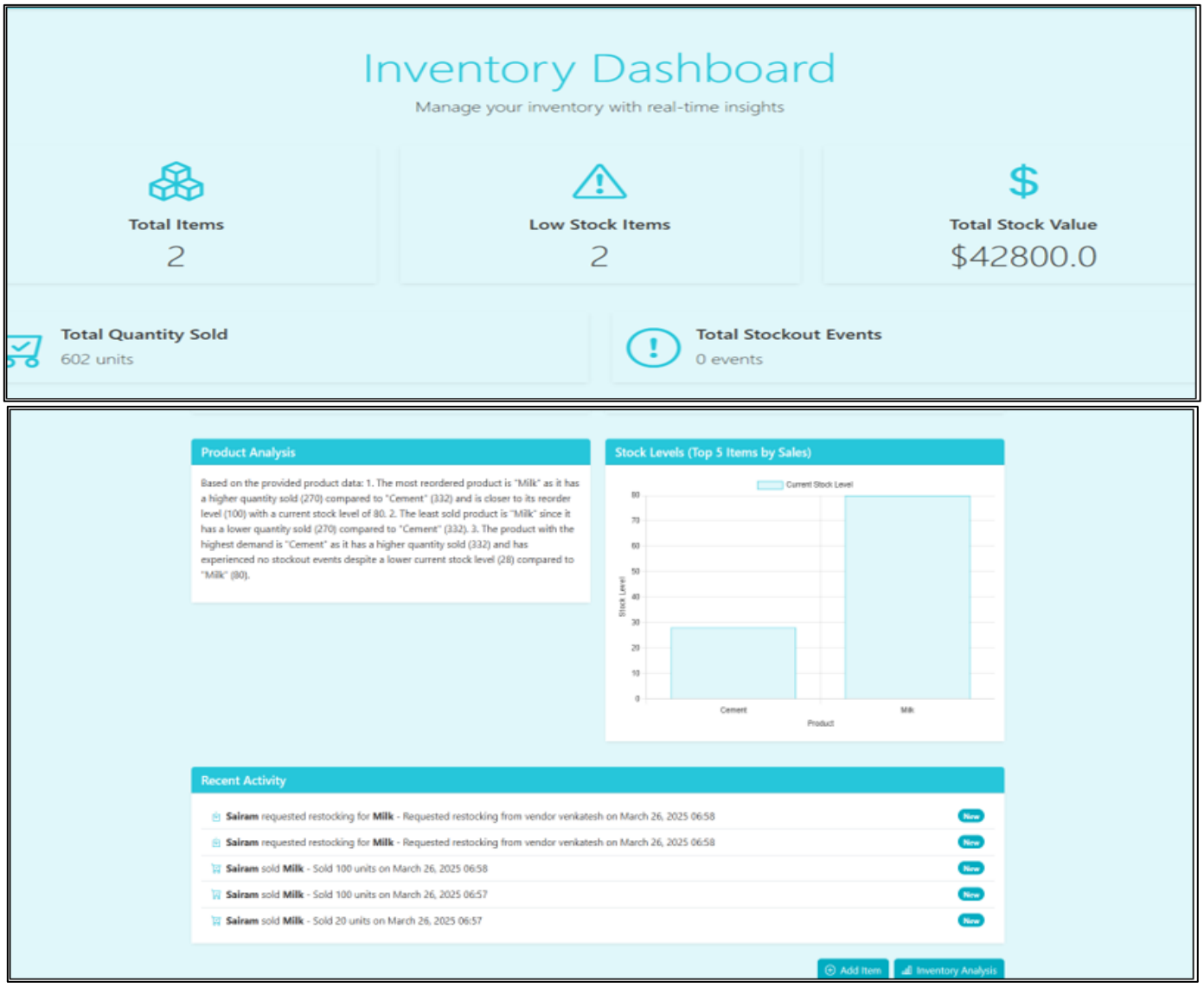

Fig 3 Inventory Dashboard Page

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25apr681

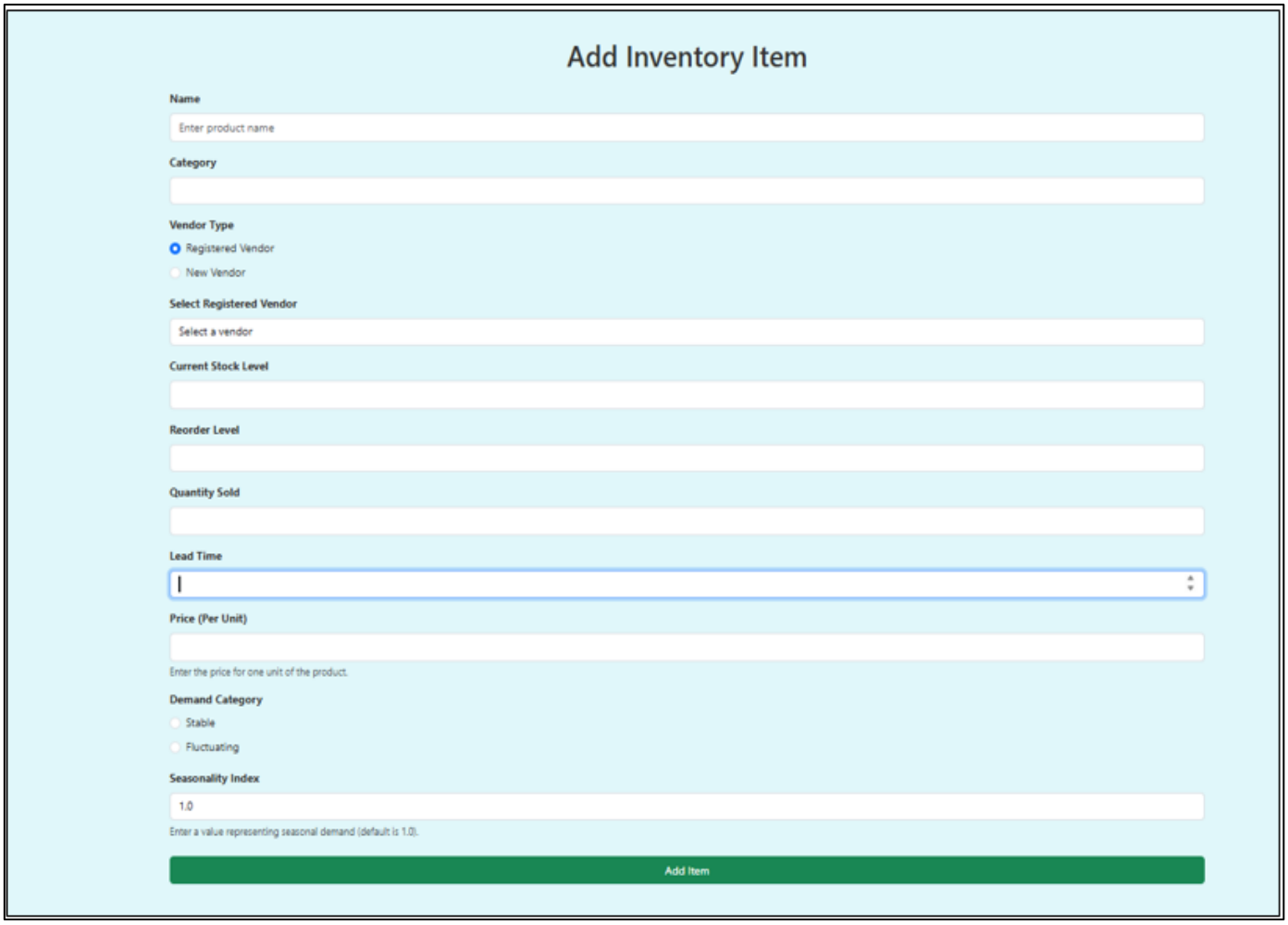

Fig 4 Add Inventory Item Page

Fig 5 Inventory Analysis Page

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25apr681

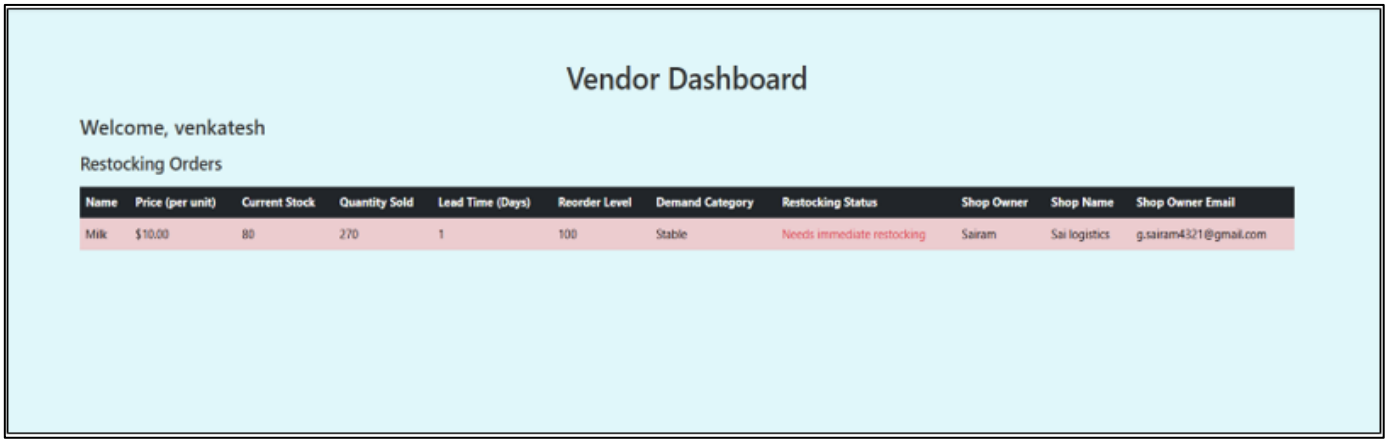

Fig 6 Vendor Dashboard Page

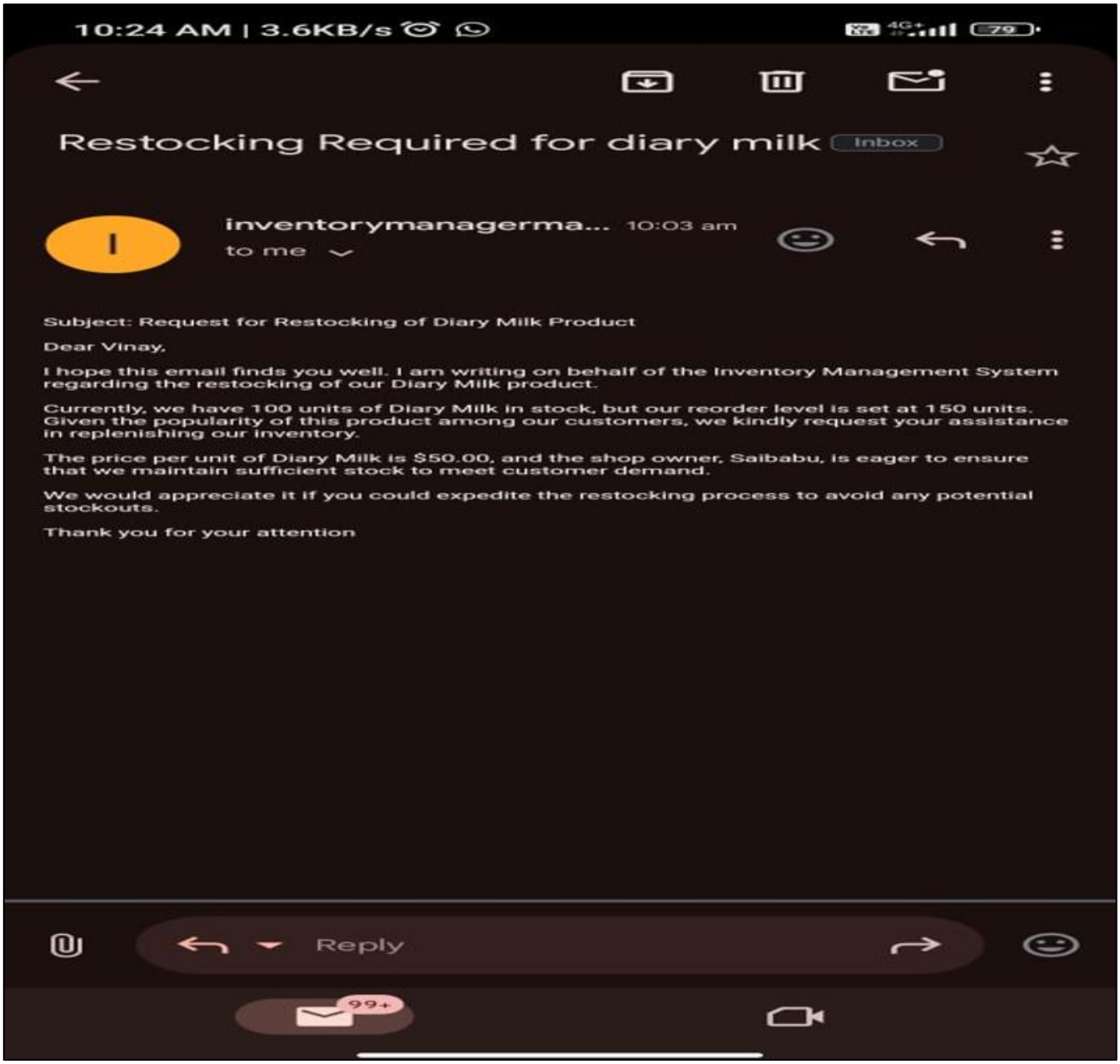

Fig 7 Email Notification

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25apr681

- *Result Analysis:*

The AI-based Inventory Management System has delivered effective performance across critical areas, supporting small business inventory processes with improved accuracy and reduced manual effort.

- *Speed:*

The system performs operations like stock updates, vendor management, and billing with minimal delay. Email alerts for low-stock items are triggered instantly, ensuring timely restocking.

- *User Experience:*

With a simple and responsive interface designed using Bootstrap, users can easily navigate between modules. Features like product categorization and stock monitoring are organized for smooth access and usability.

- *Automation:*

A major highlight of the system is the automated email notification feature. When stock for any item falls below a predefined threshold, the system automatically sends an alert to the registered email. This helps business owners take quick action without constantly monitoring stock levels.

- *Personalization:*

The integrated machine learning model analyzes product demand trends and predicts restocking needs based on item behavior, offering smarter inventory control without needing manual intervention.

- *Security and Reliability:*

Built using Django's secure framework and a well-structured MySQL database, the system maintains data integrity and ensures safe access for authorized users only.

- *Data Handling:*

The platform supports exporting stock data, helping users keep external records or reports for business analysis or offline access.

VIII. CONCLUSION

The AI-Driven Inventory Management System significantly improves stock monitoring and restocking processes through intelligent automation. By implementing the Gradient Boosting algorithm, the system accurately forecasts future inventory requirements based on historical usage patterns, helping businesses maintain optimal stock levels. Developed using Django and MySQL, the platform offers an intuitive user experience for managing inventory data, vendor details, and restocking operations. Features like real-time stock tracking, automated email alerts, and predictive analytics reduce manual workload and support timely decision-making. This approach not only increases efficiency but also enhances accuracy, making inventory management more reliable, proactive, and cost-effective for small businesses.

FUTURE SCOPE

In the future, the AI for Inventory Management system can be enhanced by incorporating more advanced machine learning models and deep learning techniques to further improve demand forecasting accuracy. Integration with IoT devices and smart sensors can enable real-time tracking of inventory levels across warehouses, ensuring more precise stock control. Voice-enabled and NLP-based interfaces can be developed to allow users to interact with the system more naturally and efficiently. Additionally, the system can be extended to support mobile platforms for remote access and real-time notifications. Strengthening data privacy through advanced security protocols and aligning with data protection regulations will also be a key focus to ensure the safe handling of sensitive business information. Expanding the system to handle multi-location inventory and vendor analytics will increase its utility for growing businesses.

REFERENCES

[1] Shamita Deshmukh, Asst. Prof. Sana Tak (2022). Inventory Management System. Journal of Scientific Research & Engineering Trends.
[2] Caylı, O., & Oralhan, Z. (2024). Artificial Intelligence-Driven Inventory Management: Optimizing Stock Levels and Reducing Costs Through Advanced Machine Learning Techniques.
[3] J.B. Munyaka & V.S.S. Yadavalli (2022). Inventory Management Concepts and Implementations. South African Journal of Industrial Engineering.