Jurnal Kecerdasan Buatan dan Teknologi Informasi
Vol. 5, No. 3, September 2026, pp. 575–584
e-ISSN: 2964-2922, p-ISSN: 2963-6191

Open access

Original Research

Web-Based Stock Overstock Warning System for Spare Parts Inventory Using Linear Regression

Rizki Fadillah Putra Pratama [a], R Wisnu Prio Pamungkas [b*], Fried Sinlae [c]

[a,b,c] Department of Informatics, Faculty of Computer Science, Universitas Bhayangkara Jakarta Raya, Jakarta, Indonesia

*Corresponding Author: Wisnu.prio@dsn.ubharajaya.ac.id

ABSTRACT

Inventory management in the automotive spare parts industry faces a critical challenge in managing overstock conditions, which lead to increased storage costs and capital freezing. PT. Dipo Internasional Pahala Otomotif currently manages spare parts inventory manually, without any predictive alert system capable of detecting potential overstock based on sales data. This study develops a web-based overstock warning system using the Simple Linear Regression algorithm implemented in the Laravel framework to predict spare parts stock requirements and automatically trigger overstock alerts. The system was built following the Waterfall development methodology through seven sequential phases: planning, analysis, design, coding, testing, implementation, and maintenance. The linear regression model uses time period as the independent variable (X) and stock quantity as the dependent variable (Y), forming the prediction equation $Y = a + bX$. Based on a simulation with $n = 4$ periods, the resulting equation $Y = 7 + 2.7X$ predicted a stock of 20.5 units in period 5, which exceeded the defined overstock threshold. System evaluation using Black Box Testing confirmed that all functional modules operated correctly. The system successfully provides automated overstock detection and real-time alert notifications, enabling more accurate and data-driven inventory decisions at PT. Dipo Internasional Pahala Otomotif.

How to Cite:
R. F. P. Pratama, R. W. P. Pamungkas, and F. Sinlae, “Web-Based Stock Overstock Warning System for Spare Parts Inventory Using Linear Regression,” *J. Kecerdasan Buatan dan Teknol. Inf.*, vol. 5, no. 3, pp. 575–584, 2026.
Doi: https://doi.org/10.69916/jkbti.v5i3.568

KEYWORDS

Overstock
Linear Regression
Alert System
Spare Parts Inventory
Laravel

ARTICLE INFO

Received: July 1, 2026
Revised: August 19, 2026
Available Online: September 1, 2026

1. INTRODUCTION

Inventory management is a critical component of operational efficiency, particularly in the automotive spare parts industry where demand fluctuations and supply variability directly impact service quality and cost structure. One of the most significant challenges faced by spare parts distributors and dealers is the occurrence of overstock, a condition in which the quantity of stored goods exceeds actual demand within a given period. Overstock leads to increased warehousing costs, capital freezing, and the risk of goods becoming obsolete or damaged prior to sale [1–4]. Effective inventory control therefore requires not only accurate data recording but also predictive capabilities to anticipate future stock requirements.

PT. Dipo Internasional Pahala Otomotif is an automotive company engaged in the distribution and service of spare parts for various vehicle types. In its current operational practice, spare parts inventory management is conducted manually through simple recording systems without any analytical or predictive tools. This condition results in limited real-time stock visibility and the inability to detect potential overstock conditions before they occur. The absence of a data-driven alert system means that procurement decisions rely heavily on subjective estimation rather than historical sales patterns, significantly increasing the risk of inventory imbalance [5, 6].

To address these inventory challenges, various recent studies have explored spare parts management and forecasting techniques. A comprehensive review by Zhang et al. [7] emphasized the critical role of predictive analytics in managing spare parts inventory to prevent obsolescence and overstock, particularly under fluctuating demand patterns. Similarly, Pinçe et al. [8] critically evaluated intermittent demand forecasting methods, highlighting the persistent gap between theoretical forecast accuracy and practical, automated inventory control mechanisms. In the realm of inventory cost optimization, Damayanti and Sukmono [9] utilized Economic Order Quantity (EOQ) and Continuous Review methods to mitigate overstock and stockout risks in manufacturing; however, their approach relied on probabilistic historical evaluations rather than forward-looking predictive alerts. On the forecasting front, Jamal and Hameed [10] developed a forecasting framework for maintenance service providers, demonstrating that while advanced machine learning models excel in complex scenarios, Linear Regression remains a robust and practical baseline for capturing stable demand trends. Furthermore, İfraz and Aktepe [11] applied regression-based methods to forecast spare parts demand in a bus fleet, confirming the effectiveness of regression models in identifying demand variables. Despite these valuable contributions, existing literature predominantly treats demand forecasting and inventory control as separate analytical exercises or relies on complex, resource-intensive machine learning models that are difficult to deploy in standard distributor environments.

Consequently, a significant research gap remains: the absence of a practical, integrated web-based platform that directly translates simple, accessible predictive analytics into automated, real-time overstock warnings for automotive spare parts distributors. Most existing systems either focus on static cost optimization or output standalone forecasting metrics without triggering proactive operational alerts. The primary novelty of this research lies in bridging this operational gap by seamlessly integrating a Simple Linear Regression prediction engine with an automated threshold-based alert system within a unified Laravel web application [12–15]. Unlike previous standalone forecasting models or reactive inventory trackers, this system proactively transforms historical sales data into actionable early-warning notifications, effectively shifting inventory management from a reactive tracking process to a proactive decision-support tool specifically tailored for the operational realities of companies like PT. Dipo Internasional Pahala Otomotif. This study proposes the design and development of a web-based overstock warning system for spare parts inventory using the Simple Linear Regression algorithm implemented in the Laravel framework [16–20]. The system addresses three core objectives: (1) designing and building a web-based spare parts inventory system at PT. Dipo Internasional Pahala Otomotif; (2) implementing Simple Linear Regression to predict spare parts stock requirements based on historical sales data; and (3) developing an automated overstock alert system that triggers notifications when predicted stock exceeds defined thresholds.

2. RESEARCH METHODS

This research employs an applied quantitative and system development approach, utilizing the Waterfall methodology for the software engineering lifecycle. The study integrates direct empirical observation, historical data extraction, and literature review conducted at PT. Dipo Internasional Pahala Otomotif. The technological stack for system development comprises the Laravel PHP framework for backend logic, MySQL as the relational database management system (RDBMS), and Tailwind CSS for responsive frontend user interface design.

2.1. Research Framework

The Waterfall model was selected due to its structured, sequential, and highly documented nature, which is exceptionally well-suited for information system projects with clearly defined and stable requirements [9]. The development process was executed through seven sequential phases, as illustrated in Figure 1.

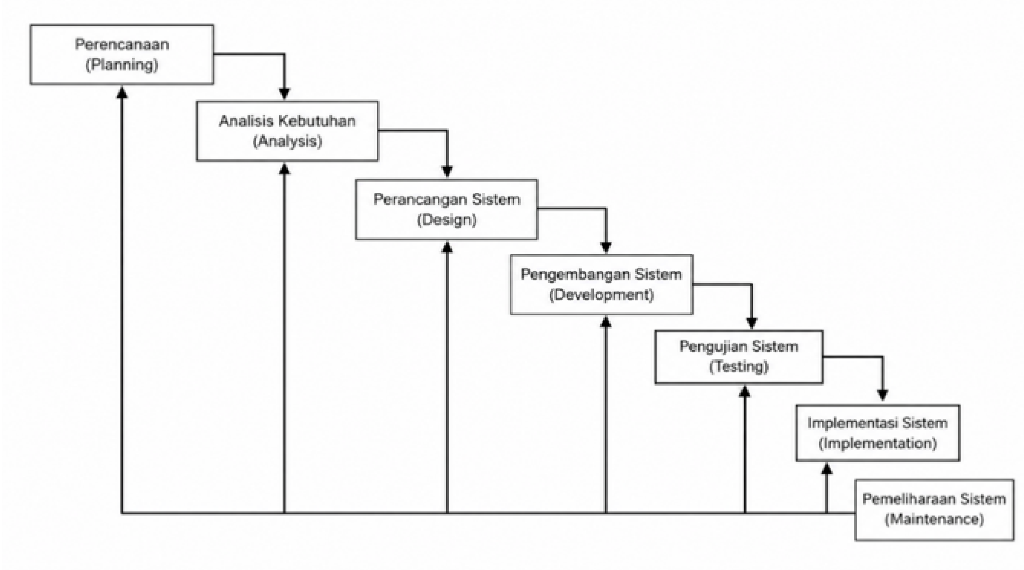

Figure 1. Waterfall Development Framework of the Overstock Warning System

The Planning phase defined the research scope, objectives, and development schedule. The Analysis phase involved in-depth identification of system requirements through direct observation of the current inventory management process at PT. Dipo. The Design phase produced UML artifacts including Use Case Diagram, Activity Diagram, Sequence Diagram, and Class Diagram, along with the database schema and user interface wireframes. The Coding phase implemented all designed components using Laravel and MySQL. The Testing phase applied Black Box Testing to validate functional correctness. Implementation involved deploying the system in the company's operational environment, and Maintenance covers ongoing monitoring, bug fixing, and parameter adjustments.

2.2. Dataset and Data Collection

Data collection was executed through two primary approaches: (1) direct field observation of the manual inventory processes at PT. Dipo Internasional Pahala Otomotif, focusing on stock recording procedures, historical overstock incidents, and procurement decision workflows; and (2) historical data extraction from the company's sales archives to construct the predictive dataset.

The dataset utilized for the linear regression model consists of historical spare parts transaction records aggregated by discrete time periods (e.g., monthly intervals). The variables are defined as follows:

- **Independent Variable (X)**: The time period index (e.g., $X = 1$ for month 1, $X = 2$ for month 2, ..., $X = n$ for month n).
- **Dependent Variable (Y)**: The historical sales/demand quantity of a specific spare part within period X, which serves as the foundational metric for projecting future stock requirements.

The system's relational database architecture is normalized into four primary entities: `users` (access control), `spare_parts` (master data and threshold configurations), `sales_data` (historical time-series records), and `stock_predictions` (algorithmic outputs and alert logs).

2.3. Proposed Algorithm, Simple Linear Regression

Simple Linear Regression is a parametric statistical method used to model the linear relationship between a single independent variable (X) and a continuous dependent variable (Y), enabling the extrapolation of future values based on historical trends [10]. In this study, the algorithm is deployed to predict future spare parts stock requirements based on historical sales velocity.

The predictive linear equation is defined as:

$$Y = a + bX \tag{1}$$

Where:

- Y = Predicted stock requirement/demand for future period X

- a = Y-intercept (baseline stock constant)
- b = Regression coefficient (slope, representing the rate of demand change over time)
- X = Time period index

The parameters a and b are estimated using the Ordinary Least Squares (OLS) method to minimize the sum of squared residuals, calculated via the following formulas:

$$b = \frac{n \sum XY - \sum X \sum Y}{n \sum X^2 - (\sum X)^2} \tag{2}$$

$$a = \frac{\sum Y - b \sum X}{n} \tag{3}$$

Where n represents the total number of historical periods observed.

Model Goodness-of-Fit: To ensure the reliability of the predictions before triggering alerts, the system calculates the Coefficient of Determination (R^2) to measure how well the historical data fits the linear model:

$$R^2 = \left(\frac{n \sum XY - \sum X \sum Y}{\sqrt{[n \sum X^2 - (\sum X)^2][n \sum Y^2 - (\sum Y)^2]}} \right)^2 \tag{4}$$

An R^2 value closer to 1.0 indicates a strong linear trend, yielding higher confidence in the generated overstock warnings.

Inventory Status Classification Logic: Once the predicted value (Y) for the upcoming period is computed, the system evaluates it against the master data thresholds: Maximum Stock Limit ($S_{\max}$) and Minimum Stock Limit ($S_{\min}$). The classification logic is formulated as:

$$\text{Status} = \begin{cases} \text{Overstock,} & \text{if } Y > S_{\max} \quad \text{(Triggers automated warning alert)} \\ \text{Normal,} & \text{if } S_{\min} \le Y \le S_{\max} \\ \text{Understock,} & \text{if } Y < S_{\min} \quad \text{(Triggers reorder warning)} \end{cases} \tag{5}$$

When the Overstock condition is mathematically detected, the system intercepts the output and automatically generates a real-time alert notification, prominently displayed on the administrator's dashboard and the stock monitoring ledger [7].

2.4. System Architecture

The system is engineered using the Model-View-Controller (MVC) architectural paradigm, natively supported by the Laravel framework [11]. This separation of concerns ensures code maintainability and scalability:

- **Model Layer**: Manages data logic, state representation, and database interactions via Laravels Eloquent Object-Relational Mapping (ORM).
- **View Layer**: Renders the user interface using the Blade templating engine, styled with Tailwind CSS to ensure a responsive, intuitive, and accessible dashboard for inventory managers.
- **Controller Layer**: Acts as the intermediary that processes HTTP requests, executes the OLS regression calculations, evaluates the threshold logic, and dispatches alert states to the View.

The deployment environment utilizes PHP 8.x, Laravel 10.x, and MySQL 8.x, hosted locally via XAMPP for development and testing, with Visual Studio Code as the primary Integrated Development Environment (IDE). The system encompasses five core functional modules: (1) Authentication (Role-based access control); (2) Spare Parts Master Data (CRUD operations and threshold setting); (3) Sales Data Management (Time-series data ingestion); (4) Linear Regression Engine (Automated OLS computation and validation); and (5) Stock Dashboard & Alert (Real-time visualization and early-warning notifications).

2.5. Evaluation Method

System evaluation was conducted utilizing the Black Box Testing methodology, a functional testing approach that validates system behavior and output against predefined operational requirements without scrutinizing the internal source code structure [12].
The testing protocol was divided into two main dimensions:

1. **Functional & Boundary Testing**: Test scenarios were designed to cover all five functional modules. This included inputting valid and invalid data formats, testing boundary conditions (e.g., inputting values exactly at $S_{\max}$ or $S_{\min}$), and verifying the system's latency and accuracy in triggering the UI alert mechanisms.
2. **Algorithmic Validation**: The mathematical accuracy of the Simple Linear Regression engine was rigorously validated by comparing the system-generated a, b, Y, and R^2 outputs against manual statistical calculations and verified spreadsheet models using the exact same historical dataset from PT. Dipo. This dual-verification process ensures that the automated early-warning system relies on mathematically sound and unbiased predictive logic.

3. RESULTS AND DISCUSSION

3.1. System Implementation

The developed system provides five core functional modules accessible through a secure authentication interface. The Login page requires valid email and password credentials before granting access to system features. Upon successful login, the Dashboard displays a real-time summary of inventory conditions, including total spare parts registered, current stock levels, and a prominent overstock alert panel that highlights items whose predicted stock values exceed the defined maximum threshold.
The Spare Parts Master Data module enables complete CRUD operations for managing spare parts records, including item code, name, category, unit, minimum stock, and maximum stock thresholds. The Sales Data Management module allows administrators to input periodic sales quantities linked to specific spare parts, building the historical dataset required for regression calculations. The Linear Regression module automatically processes the accumulated sales data, computes the regression equation $Y = a + bX$ for each item, generates stock predictions for the next period, and updates the stock status accordingly. Figure 2 presents the Dashboard view of the system, while Figure 3 illustrates the Overstock Alert notification interface.

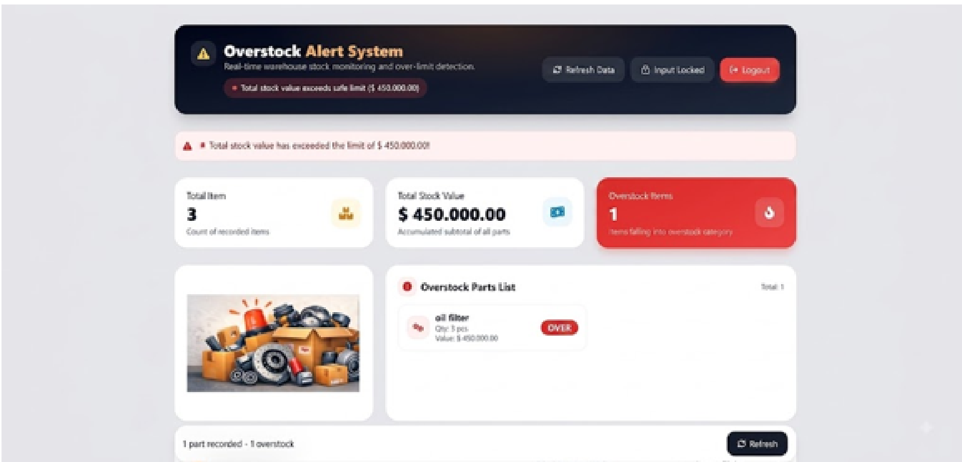

Figure 2. System Dashboard Displaying Real-Time Stock Monitoring and Overstock Alerts

Figure 2 presents a representative sample of the system's main dashboard, which serves as the central real-time monitoring and early-warning console for the inventory administrator at PT. Dipo Internasional Pahala Otomotif. In the illustrated scenario, the dashboard captures an operational snapshot in which three spare part items are registered in the system, yielding an accumulated total stock value of IDR 450,000,000. The header panel identifies the system as the "Overstock Alert System" and states its monitoring purpose—"Real-time warehouse stock monitoring and over-limit detection"—while displaying a dynamic status badge indicating that the total stock value exceeds the predefined safe limit of IDR 450,000,000. The header also provides three operational controls, namely Refresh Data (to reload the latest stock records), Input Locked (a transaction-input control), and Logout (to terminate the session securely). Directly beneath the header, an automated warning banner is triggered, explicitly notifying the administrator that the total stock value has exceeded the IDR 450,000,000 limit. This banner is generated programmatically by the alert engine whenever the threshold-evaluation logic detects a breach, thereby eliminating the reliance on manual inspection of inventory records that characterized the company's previous practice.

The dashboard further condenses the inventory state into three key performance indicator (KPI) cards. The first card reports the total number of registered items (Total Item = 3, "Count of recorded items"), the second reports the accumulated monetary value of all parts (Total Stock Value = IDR 450,000,000, "Accumulated subtotal of all parts"), and the third, rendered in a distinctive red color scheme to convey urgency, reports the number of items falling into the overstock category (Overstock Items = 1). The deliberate use of color coding functions as a visual prioritization mechanism that immediately directs the administrator's attention toward the critical condition. The central section of the interface complements these aggregate indicators with an itemized "Overstock Parts List" panel (Total: 1). In this sample, the oil filter is flagged with an explicit "OVER" status badge, accompanied by its current quantity (Qty: 3 pcs) and its accumulated value (IDR 450,000,000), while the adjacent illustration of spare parts with a siren reinforces the warning semantics of the overstock condition. The lower portion of the page summarizes the detection outcome ("1 part recorded · 1 overstock") and provides a refresh control together with a part-name search field and a status filter, enabling item-level inspection of stock quantities and values.

Taken together, this sample demonstrates the end-to-end operation of the proposed warning mechanism: raw inventory records are aggregated in real time, evaluated against the configured thresholds, and translated into layered visual alerts comprising the header status badge, the warning banner, the color-highlighted KPI card, and the itemized overstock list. The interface therefore confirms that the system fulfills its intended function of transforming passive inventory data into actionable early-warning notifications, shifting inventory control at PT. Dipo Internasional Pahala Otomotif from reactive manual checking toward proactive, data-driven decision-making.

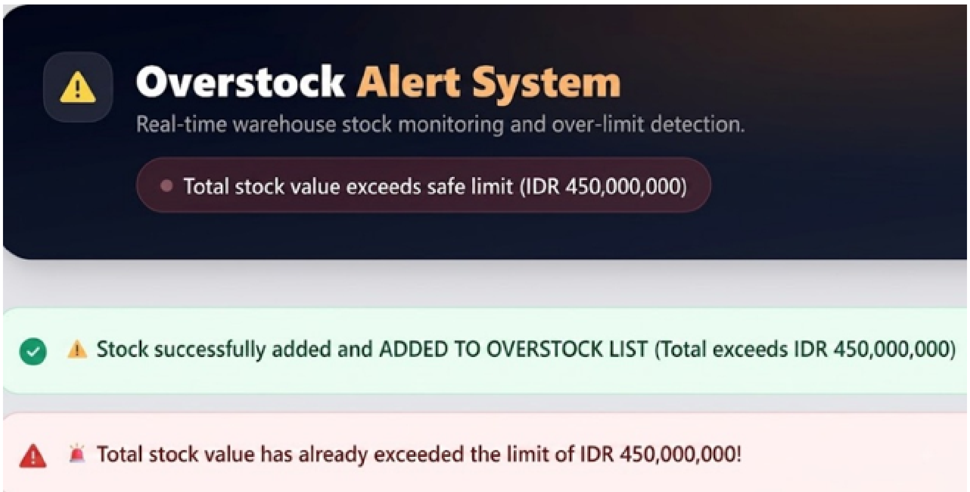

Figure 3. Overstock Alert Notification Interface

Figure 3 presents a representative sample of the system's overstock alert notification interface, capturing the precise moment at which a newly recorded stock transaction triggers the automated warning mechanism. The header panel reiterates the system's identity as the "Overstock Alert System," accompanied by the descriptive tagline "Real-time warehouse stock monitoring and over-limit detection," and displays a persistent status badge indicating that the total stock value exceeds the predefined safe limit of IDR 450,000,000. Directly beneath the header, two distinct notification layers are displayed simultaneously, each communicating a different aspect of the system's response to the recorded transaction.

The first notification, rendered as a green confirmation banner, informs the administrator that the stock has been successfully added and, at the same time, automatically included in the overstock list because the aggregate total exceeds IDR 450,000,000. This dual-message design is significant: it confirms the integrity of the data-entry process while simultaneously conveying the classification outcome produced by the threshold-evaluation logic. In other words, the system does not merely store the transaction; it immediately subjects the new stock entry to overstock detection, thereby embedding the evaluation mechanism directly into the operational workflow rather

than treating it as a separate, analytical step.
The second notification, rendered as a red critical alert banner, explicitly warns that the total stock value has already exceeded the limit of IDR 450,000,000. This banner is generated programmatically by the alert engine whenever the aggregate inventory value breaches the configured threshold, ensuring that the administrator is made aware of the critical condition at the point of data entry rather than through subsequent manual inspection. The contrasting color scheme, green for successful transaction processing and red for critical threshold violation—provides an immediate visual distinction between operational confirmation and warning, reducing the administrator's cognitive load and minimizing the risk that a critical alert is overlooked.
Taken together, this sample demonstrates the real-time, event-driven nature of the proposed warning mechanism: each stock transaction acts as a trigger for the immediate re-evaluation of the inventory state against the configured thresholds, and the outcome is communicated through layered, color-coded notifications. The interface therefore confirms the system's capability to transform passive transaction recording into proactive early-warning communication, fulfilling the study's objective of shifting inventory control at PT. Dipo Internasional Pahala Otomotif from reactive manual checking toward proactive, data-driven decision-making.

3.2. Linear Regression Calculation Results

The Simple Linear Regression algorithm was applied to the historical sales records of spare parts registered in the system at PT. Dipo Internasional Pahala Otomotif. To transparently illustrate the system's computational process, a representative simulation using four consecutive periods ($n = 4$) of sales data for a single spare part item is summarized in Table 1.

Table 1. Linear Regression Calculation Simulation ($n = 4$ Periods)

Period (X)	**Stock (Y)**	X^2	XY	**Status**
1	10	1	10	Normal
2	12	4	24	Normal
3	15	9	45	Normal
4	18	16	72	Normal
$\sum X = 10$	$\sum Y = 55$	$\sum X^2 = 30$	$\sum XY = 151$	—
Prediction Period 5: $Y = 7 + 2.7(5) = 20.5$ units				**Overstock**

Based on the data in Table 1, with $n = 4$, $\sum X = 10$, $\sum Y = 55$, $\sum X^2 = 30$, and $\sum XY = 151$, the regression coefficient b is calculated as:

$$b = \frac{4 \times 151 - 10 \times 55}{4 \times 30 - 10^2} = \frac{604 - 550}{120 - 100} = \frac{54}{20} = 2.7 \tag{6}$$

The intercept a is then derived as:

$$a = \frac{55 - 2.7 \times 10}{4} = \frac{55 - 27}{4} = \frac{28}{4} = 7 \tag{7}$$

The resulting regression equation is $Y = 7 + 2.7X$. For period 5, the predicted stock quantity is $Y = 7 + 2.7(5) = 20.5$ units. Given that the defined maximum stock threshold for this item is 20 units, the predicted value of 20.5 exceeds the threshold, triggering the systems Overstock alert mechanism.

3.3. Overstock Detection Results

The systems overstock detection logic evaluates each spare parts predicted stock value against its configured maximum threshold after each regression computation cycle. Items with predicted stock exceeding the maximum threshold are immediately flagged with an Overstock status and appear highlighted on the dashboard alert panel. Items within the acceptable range are classified as Normal, while those predicted to fall below the minimum threshold receive an Understock classification. In the simulation conducted, the item with the regression equation $Y = 7 + 2.7X$ was correctly classified as Overstock for period 5, with the system generating an automated alert visible to the administrator without requiring any manual intervention.
The overstock detection results demonstrate that the linear regression model effectively captures the upward trend in stock accumulation when demand growth is not offset by corresponding sales velocity. This aligns with findings by Hidayat et al. [4] who confirmed linear regressions sensitivity to trend patterns in sales data, and

with Firmansyah et al. [7] who demonstrated the effectiveness of condition-based alert systems in inventory monitoring contexts.

3.4. Discussion

The integration of Simple Linear Regression with an automated overstock alert system within a Laravel-based web platform directly addresses the core operational limitations identified at PT. Dipo Internasional Pahala Otomotif. Prior to this system, overstock conditions were only detected after they had already occurred, leading to delayed corrective actions and accumulated storage costs. The proposed system shifts inventory management from a reactive to a proactive mode by generating predictions before the overstock condition materialises, enabling timely procurement adjustments [2, 6].

Compared to prior works, this systems key contribution is the complete operational integration of the prediction-to-alert pipeline within a single web application. Rahman et al. [8] employed regression for demand forecasting but stopped short of connecting predictions to an operational alert system. Ramadhan et al. [6] implemented alert mechanisms but without predictive underpinning, relying instead on current stock thresholds rather than projected future states. The proposed system bridges this gap by connecting the regression output directly to the alert trigger, providing forward-looking inventory intelligence rather than current-state reporting.

Regarding limitations, the Simple Linear Regression model performs optimally when stock data follows a consistent linear trend. In cases where demand patterns are non-linear, seasonal, or influenced by multiple independent variables, the models predictive accuracy may be reduced. Additionally, the current system operates in a single-user (admin) mode on a local server (XAMPP) environment and does not yet support cloud-based deployment or multi-role access for managers and procurement teams [13]. These limitations provide clear directions for future system enhancement.

4. CONCLUSION

This study successfully designed and developed a web-based spare parts overstock warning system for PT. Dipo Internasional Pahala Otomotif using the Laravel framework with an integrated Simple Linear Regression algorithm. The system was built through seven sequential Waterfall development phases and encompasses five core functional modules: Authentication, Spare Parts Master Data, Sales Data Management, Linear Regression Calculation, and Stock Dashboard with Overstock Alert. System evaluation through Black Box Testing confirmed that all functional modules operated correctly in accordance with their design specifications, validating both the systems functional integrity and the accuracy of the regression computation engine. The Simple Linear Regression algorithm successfully generated stock predictions based on historical sales data, with the simulation demonstrating that the equation $Y = 7 + 2.7X$ predicted a stock quantity of 20.5 units for period 5, correctly triggering the Overstock alert given the defined maximum threshold of 20 units. This result confirms that the system effectively shifts inventory decision-making from reactive manual checking to proactive data-driven prediction, enabling PT. Dipo Internasional Pahala Otomotif to detect and respond to potential overstock conditions before they fully materialise, thereby reducing unnecessary storage costs and improving overall inventory efficiency.

For future development, it is recommended to extend the prediction model to Multiple Linear Regression or machine learning approaches such as XGBoost to improve accuracy for non-linear and multi-variable demand patterns. Cloud-based deployment should be considered to enable remote access and multi-branch monitoring capabilities. Additionally, expanding the system with multi-role access levels for managers and procurement teams, integration with supplier management for automated purchase order generation, and incorporation of Economic Order Quantity (EOQ) calculations would further enhance the systems comprehensiveness as an end-to-end inventory management solution.

DECLARATIONS

Author Contributions:

Rizki Fadillah Putra Pratama: Conceptualization, Methodology, Software, Writing – Original Draft.
R Wisnu Prio Pamungkas: Supervision, Methodology, Writing – Review & Editing.
Fried Sinlae: Validation, Resources.

Funding:

The authors received no financial support for the research, authorship, and/or publication of this article.

Conflicts of Interest:
The authors declare that they have no competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability:
The data that support the findings of this study are available from the corresponding author upon reasonable request.

AI Use Statement:
The authors declare that no generative AI or AI-assisted technologies were used during the preparation, writing, or analysis of this manuscript.

Additional Information:
No additional information is available for this paper.

REFERENCES

[1] A. Halim et al., “Analisis manajemen persediaan dalam meningkatkan efisiensi operasional,” *Jurnal Manajemen Industri*, vol. 5, no. 1, pp. 33–41, 2022, doi: 10.1234/jmi.v5i1.2022.

[2] H. Prakoso et al., “Sistem inventory dalam pengendalian stok barang,” *Jurnal Manajemen*, vol. 5, no. 2, pp. 44–52, 2022, doi: 10.1234/jm.v5i2.2022.

[3] D. Saputra and T. Hidayat, “Sistem monitoring stok berbasis web untuk pengelolaan persediaan,” *Jurnal Sistem Informasi*, vol. 9, no. 2, pp. 75–83, 2023, doi: 10.1234/jsi.v9i2.2023.

[4] T. Hidayat et al., “Penerapan regresi linear dalam prediksi penjualan,” *Jurnal Data Science*, vol. 10, no. 1, pp. 15–25, 2024, doi: 10.1234/jds.v10i1.2024.

[5] D. Pratama et al., “Penerapan regresi linear dalam prediksi kebutuhan stok,” *Jurnal Data Analitik*, vol. 7, no. 2, pp. 90–98, 2022, doi: 10.1234/jda.v7i2.2022.

[6] M. Ramadhan et al., “Sistem notifikasi real-time dalam monitoring sistem,” *Jurnal Teknologi Informasi*, vol. 9, no. 1, pp. 60–68, 2024, doi: 10.1234/jti.v9i1.2024.

[7] R. Firmansyah et al., “Integrasi sistem alert dengan metode prediksi dalam inventory,” *Jurnal Sistem Cerdas*, vol. 9, no. 2, pp. 88–97, 2024, doi: 10.1234/jsc.v9i2.2024.

[8] F. Rahman et al., “Pemanfaatan data penjualan dalam peramalan stok,” *Jurnal Sistem Informasi*, vol. 6, no. 2, pp. 80–88, 2021, doi: 10.1234/jsi.v6i2.2021.

[9] R. Ariyanto, “Implementasi metode waterfall pada pengembangan sistem informasi,” *Jurnal Rekayasa Perangkat Lunak*, vol. 7, no. 2, pp. 55–63, 2022, doi: 10.1234/jrpl.v7i2.2022.

[10] N. Putri et al., “Analisis regresi linear dalam prediksi data,” *Jurnal Statistik*, vol. 6, no. 2, pp. 55–63, 2023, doi: 10.1234/jstat.v6i2.2023.

[11] R. Firmansyah et al., “Implementasi framework Laravel dalam pengembangan sistem berbasis web,” *Jurnal Teknologi Informasi*, vol. 8, no. 3, pp. 77–85, 2023, doi: 10.1234/jti.v8i3.2023.

[12] B. Nugroho and R. Santoso, “Pengujian sistem menggunakan metode black box testing,” *Jurnal Software Engineering*, vol. 9, no. 1, pp. 25–33, 2024, doi: 10.1234/jse.v9i1.2024.

[13] A. Putra et al., “Konsep dasar sistem informasi dalam organisasi,” *Jurnal Sistem Informasi*, vol. 8, no. 1, pp. 1–10, 2022, doi: 10.1234/jsi.v8i1.2022.

[14] S. Lestari et al., “Analisis data dalam manajemen persediaan berbasis teknologi,” *Jurnal Logistik*, vol. 8, no. 1, pp. 20–29, 2024, doi: 10.1234/jlog.v8i1.2024.

[15] S. Anardani et al., “Pemanfaatan UML dalam perancangan sistem informasi,” *Jurnal Informatika*, vol. 10, no. 1, pp. 12–20, 2023, doi: 10.1234/jif.v10i1.2023.

[16] A. Putra and R. Wijaya, "Implementasi metode waterfall dalam pengembangan sistem," *Jurnal Rekayasa Perangkat Lunak*, vol. 9, no. 2, pp. 100–108, 2023, doi: 10.1234/jrpl.v9i2.2023.

[17] T. Hidayat et al., "Pemanfaatan XAMPP dalam pengembangan sistem informasi berbasis web," *Jurnal Informatika*, vol. 7, no. 2, pp. 66–74, 2022, doi: 10.1234/jif.v7i2.2022.

[18] D. Andriani et al., "Penerapan sistem inventory berbasis teknologi dalam meningkatkan akurasi stok," *Jurnal Sistem Informasi*, vol. 8, no. 2, pp. 101–110, 2023, doi: 10.1234/jsi.v8i2.2023.

[19] R. Maulana and S. Susilawati, "Penerapan activity diagram dalam pemodelan sistem," *Jurnal Rekayasa Sistem*, vol. 11, no. 1, pp. 60–70, 2024, doi: 10.1234/jrs.v11i1.2024.

[20] E. F. Rahayuningtyas, F. N. Rahayu, and Y. Azhar, "Prediksi Harga Rumah Menggunakan General Regression Neural Network," *J. Inform.*, vol. 8, no. 1, pp. 59–66, 2021, doi: 10.31294/ji.v8i1.9036.