

Design and Implementation of a Stock Purchase System for Printing Businesses Using the Waterfall Method

Mega Henia Br Ginting¹, Francka Sakti Lee²

^{1,2} Information Systems Department, Bunda Mulia University, Jakarta, Indonesia
Email: ¹megaginting15@gmail.com, ²flee@bundamulia.ac.id

Abstract

Efficient stock availability is essential for the seamless operation of business processes within a company. However, stock management often encounters several critical challenges, including discrepancies between warehouse inventory and logbook records, as well as mismatches between ordered and received quantities. These issues frequently lead to overstocking or stockouts overstocking increases operational costs and risks quality degradation or expiration of goods, while stockouts disrupt sales and customer service. To address these challenges, this study proposes the design of a stock purchasing management application aimed at optimizing inventory tracking and enhancing operational efficiency within a printing shop. The system is developed using the Waterfall methodology, a structured software development model that helps minimize errors throughout the design process. To validate the system's functionality, black box testing is employed, ensuring that the application performs as intended. The resulting application offers an effective solution to stock management issues, reducing inventory imbalances and supporting more efficient business operations.

Keywords: Stock Management, Inventory Optimization, Waterfall Methodology, Black Box Testing, Printing Business System

1. INTRODUCTION

In the era of technological development, information systems are one of the things that cannot be separated in various fields, one of which is in a business or business. Information system consists of the word system which means a collection of components where each component has a different function from other components and is interconnected to achieve a task[1]. Information is data that has been processed and has benefits for those who receive it [2]. From the above understanding, the information system is a combination of technology, information, various activities that support business processes and decision

making based on the data owned [3], storing and processing data for company information[4].

Currently, the growth of printing shops is accelerating because there are many needs in offices or schools related to printing, so business owners also improve the management of their stock of goods. Purchasing can simply be interpreted as a process of ordering goods or services related to the quality of goods, prices, quantities to meet needs or for resale [5]. Management can be interpreted as an activity of planning, and organizing [6], besides that stock management can affect operational costs and service to customers[7]. Currently, stock purchases are still an important thing to manage, where the amount of purchase, purchase time is one of the things that is considered to support business operational processes and operational costs[8].

In addition, the number of customer requests is often not fulfilled on time due to insufficient stock so that it can have an impact on the quality of service to customers [9], besides that there is a difference between the recording of incoming stock purchases recorded by the admin and the amount contained in the warehouse causing business operations to experience overstock or out stock problems. Stock recording is still done using a manual system, namely by using a notebook so that the risk of errors can occur. From the data collection that has been done, stock recording errors can occur on 1-3 items depending on the number and variety of items ordered. The more the number and variety, the higher the risk of error.

The urgency of this research raises the problem of purchasing stock that is experiencing problems in managing and recording stock information so that overstock and out stock occur from the printing shop business process. Overstock can cause stock accumulation so that goods can expire and cause losses to the company. Meanwhile, out stock can reduce the level of service to customers because of insufficient stock.

2. LITERATURE REVIEW

Literature study is an activity that involves collecting data related to the research conducted from various relevant written sources[9], with the aim of comparing research that has been done before and as a reference for making features in the system created [10]. In a study conducted by Rafli Fadillah Agustioa, Ahnaf Irfan Bahariantob, Riyan Pratama Muliac, Wasis Haryono entitled Design of Web-Based Inventory System and Stock Purchase Transactions Using the Waterfall Method at the Raffa Rafli Photocopy Shop in the Informatics and Computer Engineering Research Journal (Restikom)[11], it can be concluded that this study creates a web-based inventory and purchase transaction system, there are features

for recording stock, accessing stock reports in real time, and tracking transactions that can increase the speed of the work process, and the accuracy of inventory management. In its design, the waterfall method is used as a development and testing method carried out using the blackbox testing method. The results of this application design have several features such as tracking transactions, purchases, adding items, providing a stock purchase menu, and user settings.

Another study was conducted by Haikal Wahyudi, Zaeniah, Salman with the title Design and Construction of a Drug Stock Management Information System at Karya Husada Pharmacy in the EXPLORE journal.[12]. From this study, a website is designed to be used for stock management, where the website is made for pharmacists, admins, and leaders. These three users have differences and similarities in accessing the menu on the website. With this system, it is easy to check drug stock, drugs that will expire, make it easier to make sales, purchase, and stock reports. On this website there are menus such as stock of goods, purchase of goods, sales, supplier data menus, and notification features to see expired drugs, and those that will run out, and use pharmacies. Using the waterfall method for the system development method and using blackbox to conduct research.

The next research is entitled Analysis and Development of Stock Monitoring System, Sales and Purchase of BBM (Fuel Oil) (Case Study at Gas Station 85.988.02 Wadio) conducted by Gunawan Prayitno, Novita Paraga, Usman Arfan in the journal Technology and Informatics[10]. In this study, the calculation of fuel reserves at SPBU85 is less than optimal because sales transactions are recorded manually so that the process of searching and storing sales transaction data, inventory, and monthly transaction reports is considered less accurate. From this problem, a fuel oil monitoring system is created to determine the amount of stock owned. This study uses a prototype method to improve stock monitoring, sales and purchases of fuel to improve the quality of service in the energy industry.

This research was conducted by Iwan Setiawan with the title Design and Construction of Stock Forecasting Application Using the Weighed Moving Average (WMA) Method at XYZ Goods Store in the Informatics Engineering Journal.[13]. From this study it is known that changing demand can affect the inventory of goods in this store, in addition, stock purchase transactions have difficulty in determining the amount purchased for the next period. This problem results in a shortage of stock or remaining stock. From this problem, forecasting or prediction is made using the Weighed Moving Average (WMA) method with the aim of helping to determine the inventory of goods. The results of the calculations carried out showed that the WMA results carried out on September data were 69.8 which was almost the same as the original data, in addition the

results of the Mean Forecast Error were 0.2, Mean Absolute Deviation 0.2, and Mean Squared Error 0.0, where the smaller the error value, the more appropriate it is to use.

Further research was conducted by Andi M Nurul Aksa, Riskayani with the title Stock Management Information System Using the Rapid Application Development Method at the Sentral Jaya Soppeng Store in the Scientific Journal of Information Systems and Informatics Engineering (JISTI) [14]. In this study, the store manages and records data on goods, suppliers, and sales in a ledger, which can be said to still be done manually and is prone to errors. Based on this condition, an application design is made for this problem that can monitor stock, and help in seeing goods coming in and out of the store. This design uses the SDLC method to help produce high-quality applications, with a short time, low cost and according to user needs. This design is expected to help facilitate the store in managing their stock.

3. METHODS

The Waterfall method is a classic and widely adopted system development approach characterized by a sequential workflow where each stage must be completed before progressing to the next [15]. This structured methodology is particularly beneficial in projects requiring thorough documentation and well-defined objectives. In the initial phase, ideas are gathered and refined based on user needs. Following this, a design process is executed in alignment with the specific business requirements. The third phase is implementation, and the subsequent phase involves security and support, where users test the system and ensure compatibility and security within their operational environment [16].

This phased approach aids researchers in designing a system supported by clear and consistent documentation at every level. Moreover, it minimizes the likelihood of design errors and establishes a strong foundation for the initial development of the system. The stages of the Waterfall method include [17]:

- 1) Requirement Analysis: This involves comprehensive information gathering to identify user needs and conducting a thorough analysis of data and software specifications.
- 2) System Design: Based on the requirements, this stage involves creating an application design tailored to user expectations, including interface and architectural planning.
- 3) Implementation: Here, the approved design is translated into functional code by programmers using preselected software tools.
- 4) Testing: Post-implementation, testing ensures that the developed system meets user needs and adheres to the original specifications. It also helps identify and resolve potential errors [18].

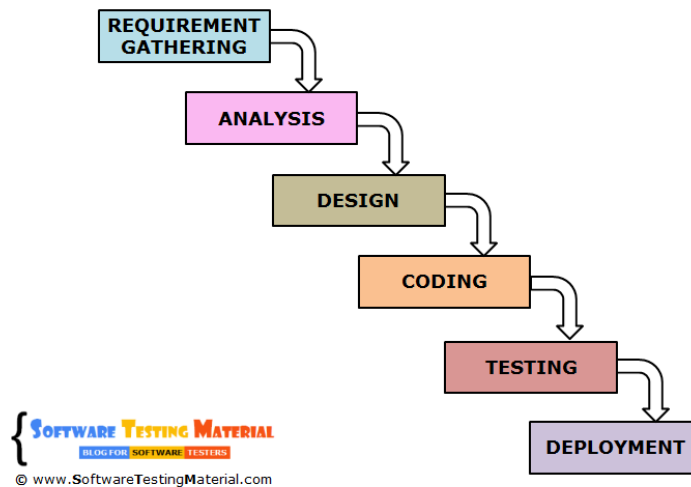


Figure 1. Waterfall Method [16]

Figure 1 visually outlines each phase of the Waterfall method, which are further detailed as follow.

1) **Requirement Definition**

This foundational phase aims to ensure that the proposed system design aligns with the identified challenges and provides an effective solution [19]. Researchers begin by identifying operational issues and inefficiencies within existing business processes. Data is collected through interviews with the printing shop owner to understand current operations, specific pain points, and desired outcomes. Observations are also conducted to gain direct insight into business workflows. The data collected is then analyzed to determine the most effective features and functionalities needed in the new system, laying the groundwork for its architecture.

2) **System and Software Design**

Following the analysis phase, the system design process begins. This involves developing a detailed blueprint for a computer-based solution, including interface layouts, database structures, and overall system architecture [20]. This phase ensures that every aspect of the system is visualized before any code is written. In this research, tools like use case diagrams, BPMN models, and sequence diagrams are employed to map out the functional and procedural elements of the stock purchasing management system, ensuring the final application meets operational goals.

3) **Implementation and Unit Testing**

At this stage, the previously crafted system design is transformed into a working application. Developers use PHP and MySQL as the programming languages to bring the system to life. Unit testing is

conducted simultaneously to verify that individual components of the system function correctly and are aligned with the system's objectives. This coding phase is critical as it represents the actualization of all prior planning.

4) **Integration and System Testing**

Once implementation is complete, rigorous testing is conducted to validate the overall system. This includes checking the system's workflow, verifying the functionality of each feature, and ensuring it performs as expected under real-world conditions [21]. The research utilizes black box testing to focus on functionality without delving into internal code logic. This method ensures that the user interface, operational commands, and processing features are all functioning correctly. If issues are identified, they are promptly addressed to enhance performance and reliability.

5) **Operation and Maintenance**

After successful testing and deployment, the system enters the maintenance phase. This includes ongoing support and updates to correct any errors that may arise during long-term use [22]. Maintenance also involves monitoring the system's performance, conducting periodic checks, and implementing enhancements or new features as business needs evolve. Continuous maintenance ensures that the system remains robust, secure, and relevant over time, sustaining operational efficiency in the printing business.

4. RESULTS AND DISCUSSION

4.1. Analysis

In the process of purchasing stock that does not yet have a system, there are several obstacles faced in managing the company's stock purchases. Some of the obstacles faced are inaccurate purchase reporting data, where there can be differences in the number of goods in the admin's notebook with the number in the warehouse. This difference can cause overstock, a condition in which there is a buildup of goods due to continuous purchases that can cause the piled up goods to be damaged or expired so that this condition can cause losses to the company. Another condition faced from differences in stock records is out of stock, where this condition can reduce the level of service to customers because in its condition the number of stocks in the notebook is still available while in the warehouse the goods are no longer there.

By using the application, it can help the admin in managing his stock purchases. Where the admin can enter data into the application that has been created, then the amount will be added to the stock data where this stock data will be updated

if there is a sale to the customer and stock purchase. From the implementation of this application, it is expected to minimize overstock or outstock so that it can improve the business process and quality of service to customers.

In this study, BPMN (Business Process Modeling Nation) was used to describe the activities that occurred. BPMN is a business process diagram used to analyze and describe how activities occur [23]. The flow of activities in the goods purchasing application contains several activities carried out by customers and admins. In the customer section, they will order goods, then the admin will receive the order and process it. If the order is complete, the customer can receive their order and the stock data will decrease. Furthermore, they will order stock from the supplier, in the second activity the goods will be sent by the supplier and the admin will receive the goods. In the third activity, the admin will receive an invoice from the supplier as proof of purchase and the price of the goods.

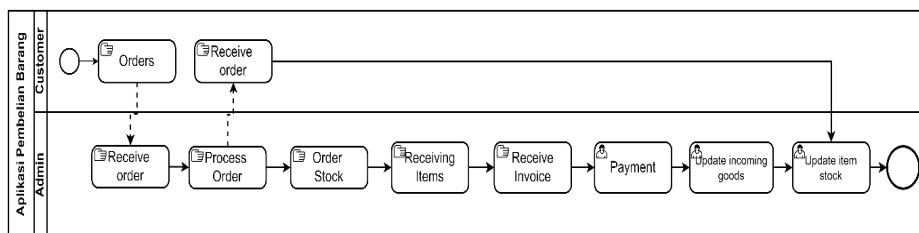


Figure 1. BPMN Process

After that, in the fourth activity, the admin will make a payment according to the price and quantity of goods, in the fifth activity, the admin will input the data of goods that have been received on the website that has been created, in this activity, the admin will input a number of data requested on the website, and in the sixth activity, the system will update the stock amount according to the amount entered by the admin in the incoming goods data with the stock amount before being updated if any, as shown in Figure 2.

4.2. Planning

At this stage will explain how the system process occurs in the stock purchase application and how the flow in managing stock purchases into the application. In this study using a use case diagram to describe the flow that occurs. Use case diagrams provide an overview of what interactions occur and can be performed by actors [24]. Use case diagrams help system developers to understand the interactions that occur. In this diagram, actors can perform many use cases and uses can have many actors [25]. Use cases are used to find out what functions occur and who can use these functions [26]. At this stage using two actors,

namely customer and admin. In the customer actor makes an order provided by the printing shop.

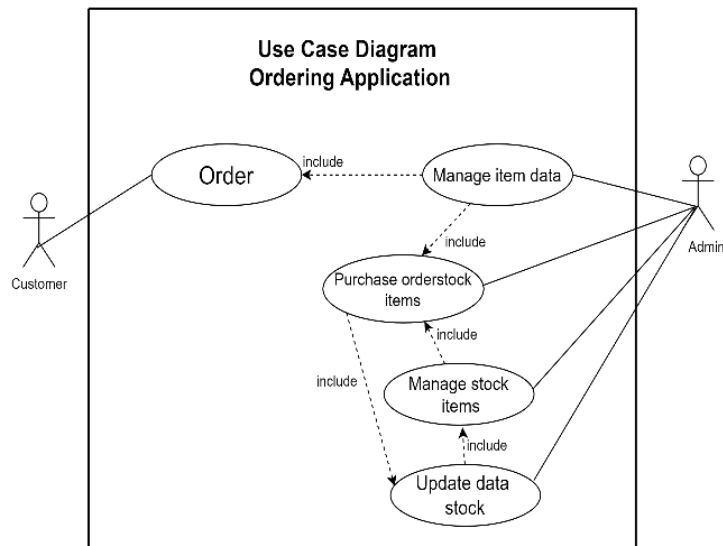


Figure 2. Use case diagram ordering application

In the customer actor can do several activities, namely the admin can manage the goods data, this will be related to what can be ordered by the customer. In addition, the admin can also order stock of goods to the supplier to ensure that the stock is still available, for ordering stock of goods the admin can access stock data to find out how much is available so that it can reduce overstock. Then the admin can input data if the goods ordered have been received and are in accordance, after inputting this data, the stock data will change according to the amount entered in the incoming goods data with the remaining amount before the stock of goods is input, from Figure 3.

4.3. Design

At this stage, a sequence diagram of the stock purchase application is designed. Sequence diagram is a diagram that provides an overview of the activity of a use case by defining the timing of events, messages received or sent by objects[27]. The sequence diagram must describe all use cases, interactions that occur in each use case, and the separate process of each use case so that the more use cases, the more sequence diagrams are made [28]. In Figure 4, the customer will order goods and will receive the ordered goods. In Figure 5, it starts from the admin receiving the order then processing the order. Then the admin can order stock and wait for the goods to arrive. When the goods arrive, the admin will receive

the goods according to the amount ordered, then the admin can input incoming goods data and the system will update the data that has been input by the admin. When the goods data has been inputted, the system will update the amount of stock in the stock data according to the amount inputted in the incoming goods data.

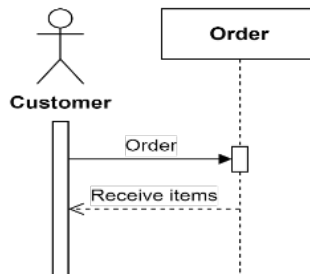


Figure 3. Customer sequence diagram

In the incoming goods data, the admin can edit the data if there is a discrepancy, then it can be saved and the system will update the incoming goods data and stock data. In addition, the admin can also delete incoming goods data, then the system will delete the data and in the stock data the goods data will be deleted as well.

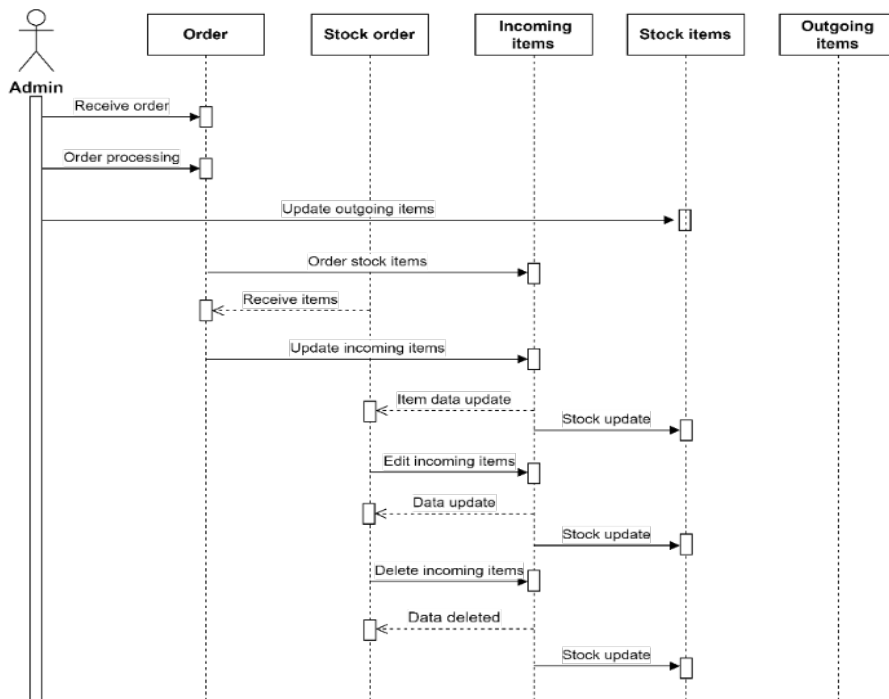


Figure 4. Sequence diagram admin

4.4. Implementation

The system implementation in this study uses prototype results, to determine whether the design results have met the objectives and uses a dashboard for the stock purchasing design system.

1) Purchase Order View

On the purchase order page, users can see a table of incoming goods containing several data and images of the goods. In addition, other menus can be seen such as stock of goods, goods data, sales, and other menus. On the purchase order, there are several actions that can be performed by the user, namely viewing details, editing, deleting, searching, and adding data. On this page, users can see details of an item, where details of the item such as the name of the item, proof of payment, proof of invoice, and other data will be displayed. Furthermore, there is an edit menu that can be used to change data that is not appropriate.

When changing data, a form containing information about the item will be displayed, then the user can make changes to the data, then the user can save or cancel the data changes, if the user chooses save then the system will save the data changes and display them in the purchase order data. In addition to editing, users can also delete items, when clicking the delete button, the user will get a pop-up message containing confirmation to delete the message. The user can choose to cancel or delete the data and the system will delete the data, as shown in Figure 6.



Id Barang	Barang	Nama Supplier	Tanggal Masuk	Jumlah	Total Pembayaran	Action
00A1		Toko Serba Barang	10/09/2024	10 dus	Rp. 3.000.000,00	Detail Edit Delete
00B2		Toko Serba Barang	13/09/2024	20 pcs	Rp. 1.700.000,00	Detail Edit Delete

Figure 5. Purchase Order View

2) Purchase Order View Add Data

When the user adds an item, it can be done via the plus button in the upper right corner. When clicking this button, a form will be displayed to add data. In this

form, the user will be asked to fill in some of the requested data such as the name of the item, date of entry, quantity, upload proof of payment, and other data. After completing the requested data, the user can choose to save and the system will save and display it in the purchase order table or cancel the data, as shown in Figure 7.

Figure 6. Add data

3) Stock View

After the admin saves the added items, the stock items table. On this page contains items with the amount of stock of each item. When the admin adds data on incoming goods, it will be updated according to the amount in the purchase order table and the remaining stock and will decrease if the admin adds data on sales in the stock table as shown in Figure 8.

Id Barang	Barang	Nama Supplier	Tanggal Masuk	Stok	Total Pembayaran	Action
00A1		Toko Serba Barang	10/09/2024	20 dus	Rp. 3.000.000,00	Detail Edit Delete
00B2		Toko Serba Barang	13/09/2024	20 pcs	Rp. 1.700.000,00	Detail Edit Delete

Figure 7. Stock of items

4.5. System Testing

This study employs black box testing as the primary method for evaluating the functionality of the developed stock management system. Black box testing is designed to assess whether the system meets its functional requirements without examining the internal code structure [29]. The focus is on user interactions and outputs based on various input conditions, ensuring that each component functions as expected from an end-user perspective. This includes testing both individual features (unit testing) and broader system functionalities [30]. During the testing phase, users are invited to engage with the system by performing typical actions such as adding, updating, and deleting item data. These scenarios are selected to mimic real operational tasks and evaluate whether the system responds accurately and effectively to user commands [31]. The primary objective is to confirm that the user interface behaves consistently with design expectations and business requirements.

The black box testing process begins with the Purchase Order menu. This test evaluates how data is displayed on the interface and ensures that each data field, including item images, names, and details, is rendered correctly. Next, the search function is tested. Users enter item names or keywords into the search bar located above the data table. The system is expected to filter and display relevant results instantly, accurately reflecting all matching entries stored in the database. Subsequent testing involves the Add Item menu. When the user clicks the add (+) button, a pop-up form should appear, allowing them to input new item details such as the name, quantity, entry date, proof of payment, and related documentation. Once submitted, the system should save the new data and update the purchase order table accordingly.

Another critical test involves the View Details feature. This function allows users to access a more detailed view of a selected item. Information such as the invoice and payment proof is displayed, often in image format, and is available for download. The goal is to provide users with all essential purchase verification details in a streamlined interface without navigating away from the main view. The Edit Item menu is then tested to ensure data modification is seamless. Users access this function by clicking the edit button on an item entry. A pre-filled form appears, allowing the user to update the item's information. Upon saving, the system should immediately reflect the updated data in the main view, confirming successful modification.

Finally, the Delete Item functionality is tested. When selected, the system prompts the user with a confirmation pop-up to prevent accidental deletions. If the user confirms, the system removes the item from the database and refreshes the view. If canceled, the data remains intact. Table 1 summarizes the test results

across all features. Overall, the black box testing demonstrates that the stock management application aligns with user expectations, fulfills the intended business functions, and adheres to system design requirements. This method validates that the application can effectively support inventory management operations in a printing business context.

Table 1. Black Box Test Results

No.	Functionality	Test Case	Expected	Actual	Conclusion
1	Access Purchase Order Menu	Click the "Purchase Order" menu	System displays all purchase order data	System successfully displays purchase order data	Valid
2	Use Search Function	Click the search button	System filters and displays items based on input keyword	System displays matching item data	Valid
3	Add New Item	Click the "Add" button	System opens a data entry form to input new item details	Form appears, data is saved and displayed correctly in the table	Valid
4	View Item Details	Click the "Details" button	System displays full item details including images and documents	System shows item information correctly	Valid
5	Edit Existing Item	Click the "Edit" button	Opens pre-filled form; user can save changes or cancel to discard	Changes saved and updated on table if saved; no changes if cancelled	Valid
6	Delete Item	Click the "Delete" button	Pop-up confirmation appears; item is deleted if confirmed	Item deleted upon confirmation, remains if cancelled	Valid
7	View Proof of Invoice	Click on "PO Proof"	Image of the invoice is displayed in a preview window	Invoice image is displayed correctly	Valid
8	View Proof of Payment	Click on "Proof of Payment"	Displays image/document of the payment proof	Payment proof displayed as expected	Valid
9	Submit Empty Form	Click "Add" and	System should show error	System saved empty data	Invalid

		then "Save" without filling in required fields	message and prevent saving	without error message	
10	Upload Invalid File Format for Invoice	Upload image format instead of PDF in the invoice proof field	System rejects file and shows warning	System accepted image format without warning	Invalid

In this test there are two invalid tests. The first is clicking the Add button without filling in anything and clicking save, the expected result is that there is a notification that the data must be filled in but the result is that the system saves empty data. This is invalid because the form should have validation from to ensure the data must be filled in. Second, uploading files in non pdf form, the expected result is that the system refuses to save, but the result is that the system accepts the data. This is invalid because the system should limit the file format and if it is out of format then a warning will be displayed. To overcome these two things, further checks and improvements can be made from the backend side to ensure this feature can run according to its rules.

Figure 9 is a summary of the results of black box testing in the form of a bar chart. From the diagram, it can be seen that the number of valid tests is 8 tests and 2 invalid tests. From the tests that have been carried out, respondents state that they are satisfied and easy to use the system that has been designed. Respondents easily use the features in the system, especially the stock purchase management feature which helps in managing stock items to be more optimal. Although there are still opportunities for further system development such as reminders if the amount of stock is low, overall the system designed has met user needs and is good to use.

The existence of a stock purchase management system can help printing business owners to manage stock to be more optimal. Business owners can use these features to view their stock data, input stock purchase data so as to reduce the occurrence of overstock. In addition, business owners can find out information about their stock items such as date purchased, quantity purchased. In addition to inputting stock purchase data, users can also delete or edit data if at any time there is damage to the item so that it can no longer be used or other conditions. By doing this, the stock information will be updated and the amount of stock purchased can be optimized. In addition, users can also input invoices or proof of payment to ensure the stock purchase process is appropriate.

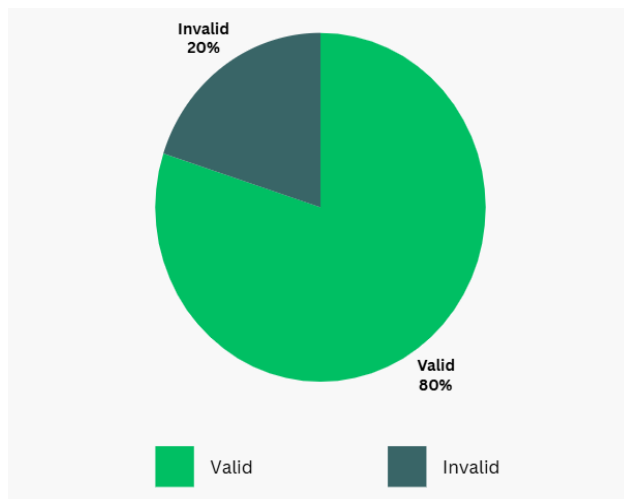


Figure 9. Black box testing

4.6. Discussion

The stock purchase management system developed in this study has proven effective in supporting business operations, particularly within the printing industry. By centralizing and automating stock purchasing, the system helps prevent critical inventory issues such as overstocking—which can lead to increased operational costs and product expiration—and stockouts, which hinder customer service and disrupt business continuity. The structured design of this system ensures that inventory is managed accurately and efficiently, significantly improving day-to-day business processes.

Beyond its successful application in the printing sector, this system demonstrates strong potential for scalability and adaptability across other industries with similar inventory management needs. For example, in the pharmaceutical industry, accurate tracking of drug stock, including expiration dates and current inventory levels, is crucial. Similar to the printing industry, pharmacies must manage sensitive inventory, and the system's features—such as automated stock entry, alerts, and detailed tracking—are well-suited to meet these requirements.

Moreover, the system can be effectively implemented in daily needs retail businesses such as grocery stores or staple food providers. These businesses rely heavily on accurate stock levels to meet customer demand consistently. Integrating this management system would enable timely restocking, reduce waste, and ensure high customer satisfaction through the continuous availability of essential items.

This study utilized the Waterfall development methodology, which offers a linear and systematic approach, suitable for projects with well-defined requirements. However, other research highlights that Rapid Application Development (RAD) can also be an effective alternative. RAD allows for faster iterations and continuous user feedback, making it ideal for dynamic environments or businesses with evolving stock management needs.

For system validation, black box testing was employed—an industry-standard method that focuses on evaluating system functionality based on user expectations without diving into internal code structures. This testing approach is particularly effective in systems like this one, where usability, interface response, and process accuracy are critical to business success. The stock purchase management system designed in this study has not only demonstrated its value in the printing industry but also holds promising applicability in various other fields such as healthcare and retail. Its scalability, coupled with a reliable development and testing framework, makes it a viable solution for any business that requires streamlined, accurate, and responsive stock management.

5. CONCLUSION

The stock purchase management system developed in this study has demonstrated its effectiveness in improving business operations, particularly in inventory handling and procurement data accuracy. By enabling administrators to input purchase data along with supporting documents such as invoices and payment proofs the system ensures that stock levels are accurately updated, minimizing the risk of errors commonly found in manual recording. The use of the Waterfall development methodology provided a clear, step-by-step framework that contributed to the system's structured implementation. Each phase from requirements analysis to testing was carefully executed, resulting in a reliable and user-friendly application that meets the operational needs of a printing business. The system not only facilitates better tracking of incoming goods but also allows for easier data management through features like add, edit, delete, and view functions, all of which enhance overall inventory visibility and control.

However, the system is not without its limitations. It currently lacks real-time integration, depends on manual data entry, and may not fully support the demands of large-scale enterprises. These constraints present opportunities for future development. Integrating automated alerts for low stock levels, real-time data synchronization, and scalable architecture would significantly improve its performance and usability across various industries. In essence, while the current system offers substantial benefits for small to mid-sized businesses, its full potential can be realized through continued enhancement. With further

development, it can become a comprehensive solution not only for the printing industry but also for other sectors requiring efficient and accurate stock purchase management.

REFERENCES

- [1] I. Jauhari, "Sistem Informasi Manajemen Pendidikan Islam," *Tarbawi Ngabar J. Educ.*, vol. 2, no. 2, pp. 190–208, 2021, doi: 10.55380/tarbawi.v2i2.130.
- [2] H. Saputri, U. Kusnaedi, and Y. Asmana, "Pengaruh Sistem Informasi Akuntansi Terhadap Kualitas Laporan Keuangan Perusahaan Jasa di Jakarta Utara," *J. Ilm. Multidisiplin*, vol. Volume 1, no. 4, pp. 102–109, 2023, doi: 10.5281/zenodo.7932454
- [3] I. Juarsyah and H. Mulyono, "Analisis Dan Perancangan Sistem Informasi Pengaduan Masyarakat Berbasis Android Pada Dinas Komunikasi Dan Informatika Kota Jambi," *Manaj. Sist. Inf.*, vol. 6, no. 1, pp. 142–152, 2021.
- [4] Y. Fitriani, S. Utami, and B. Junadi, "Perancangan Sistem Informasi Human Capital Management Berbasis Website," *J. Inf. Syst. Applied, Manag. Account. Res.*, vol. 6, no. 4, pp. 792–803, 2022, doi: 10.52362/jisamar.v6i4.919.
- [5] M. Idwal *et al.*, "Perancangan Sistem Stok Barang Dengan Metode Fifo," *Semin. Nas. Inov. Teknol.*, pp. 320–325, 2022.
- [6] J. Beno, A. . Silen, and M. Yanti, "No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する共分散構造分析Title," *Braz Dent J.*, vol. 33, no. 1, pp. 1–12, 2022.
- [7] E. Penggunaan and A. Bima, "Efektivitas Penggunaan Aplikasi Bima Terhadap Pengelolaan Stok Opname B21," vol. 10, no. 204, pp. 827–837, 2025.
- [8] A. Nurkholis and P. S. Oktora, "Sistem Persediaan Obat Menggunakan Metode Moving Average Dan Fixed Time Period With Safety Stock," *J. Sains Komput. Inform.*, vol. 6, no. 2, pp. 1134–1145, 2022.
- [9] F. S. Lee, F. Nurprihatin, H. Honni, A. P. Santoso, and F. F. Tampinongkol, "Aplikasi Pelaporan Kerja Cleaning Service Dengan Metode Waterfall," *Infotech J. Technol. Inf.*, vol. 10, no. 1, pp. 61–70, 2024, doi: 10.37365/jti.v10i1.248.
- [10] N. Paraga, G. Prayitno, U. Arfan, and P. S. Informatika, "Analisis Dan Pengembangan Sistem Monitoring Stok ," vol. 1, no. 2, pp. 58–74, 2024.
- [11] R. F. Agustio, "Perancangan Sistem Inventory dan Transaksi Pembelian Stok Barang Berbasis Web Dengan Metode Waterfall," vol. 6, no. 3, pp. 554–564, 2024.
- [12] Z. Zaeniah, "Rancang Bangun Sistem Informasi Pengelolaan Stok Obat Pada Apotek Karya Husada," *Explore*, vol. 11, no. 2, p. 146, 2021, doi: 10.35200/explore.v12i1.542.

- [13] I. Setiawan, “Rancang Bangun Aplikasi Peramalan Persediaan Stok Barang Menggunakan Metode Weighted Moving Average (Wma) Pada Toko Barang Xyz,” *J. Tek. Inform. Vol. 13, No. 3, Agustus 2021*, vol. 13, no. 3, pp. 1–9, 2021.
- [14] A. M. N. Aksa and R. Riskayani, “Sistem Informasi Pengelolaan Stok Barang Menggunakan Metode Rapid Application Development Pada Toko Sentral Jaya Soppeng,” *J. Ilm. Sist. Inf. dan Tek. Inform.*, vol. 5, no. 2, pp. 87–96, 2022, doi: 10.57093/jisti.v5i2.132.
- [15] L. Fauziah, A. Firmansyah, and A. Aguswin, “Sistem Informasi Sekolah Berbasis Web Menggunakan Metode Waterfall. Studi Kasus: SMPI Al-Hudri Walibrah,” *REMIK Ris. dan E-Jurnal Manaj. Inform. Komput.*, vol. 8, no. 1, pp. 274–285, 2024.
- [16] F. S. Lee, K. Aprilia, D. F. Dinata, W. Fernando, and J. F. Andry, “Aplikasi Pengelolaan Stok Bahan Baku dengan Metode Waterfall Pada Pabrik Plastik,” *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 6, no. 2, pp. 258–265, 2024, doi: 10.47233/jteksis.v6i2.1273.
- [17] S. Supiyandi, M. Zen, C. Rizal, and M. Eka, “Perancangan Sistem Informasi Desa Tomuan Holbung Menggunakan Metode Waterfall,” *JURIKOM (Jurnal Ris. Komputer)*, vol. 9, no. 2, p. 274, 2022, doi: 10.30865/jurikom.v9i2.3986.
- [18] J. F. Andry, F. S. Lee, Y. M. Geasela, R. Kamila, S. Meyliana, and S. Winata, “Rancang Bangun Aplikasi Member Parkir Terintegrasi dengan Kartu Tanda Mahasiswa,” vol. 4, no. 2, pp. 1–13, 2024.
- [19] R. R. Ibnu Choldun, “Penerapan Metode Waterfall Pada Aplikasi Pembelajaran Seni Budaya Berbasis Website Menggunakan Framework Reactjs,” vol. 9, no. 13, pp. 335–348, 2023.
- [20] A. Duma and E. A. Pusvita, “Pengembangan Sistem Informasi Data Siswa Berbasis Web Pada Smpn 09 Nabire Dengan Metode Waterfall,” *J. Inf. Syst. Manag.*, vol. 5, no. 1, pp. 70–76, 2023, doi: 10.24076/joism.2023v5i1.1115.
- [21] B. Fachri and R. W. Surbakti, “Perancangan Sistem Dan Desain Undangan Digital Menggunakan Metode Waterfall Berbasis Website (Studi Kasus: Asco Jaya),” *J. Sci. Soc. Res.*, vol. 4, no. 3, p. 263, 2021, doi: 10.54314/jssr.v4i3.692.
- [22] E. A. Giofandi, A. Novalinda, D. Sekarjati, M. A. Pratama, and C. E. Sekarrini, “Analisis Aksesibilitas Fasilitas Kesehatan di Kota Pekanbaru, Indonesia,” *J. Inf. Syst. Dev.*, vol. 8, no. 1, pp. 1–6, 2023, doi: 10.19166/isd.v8i1.581.
- [23] N. E. M. Herlina, S. T. Abdulhafizh, I. A. Ghafur, and ..., “Evaluating and Implementing Business Model Based on BPEL Structure: A Systematic Literature Review,” *Informasi, Tek. dan*, vol. 1, no. 1, pp. 41–56, 2024.
- [24] I. Muhammad, M. Masnur, and A. G. Syam, “Aplikasi Qr Code Sebagai Sarana Penyampaian Informasi Pohon Dikebun Raya Jompie,” *J. Sintaks*

- Log.*, vol. 1, no. 1, pp. 33–41, 2021, doi: 10.31850/jsilog.v1i1.694.
- [25] L. Setiyani, “Desain Sistem : Use Case Diagram Pendahuluan,” *Pros. Semin. Nas. Inov. Adopsi Teknol. 2021*, no. September, pp. 246–260, 2021.
- [26] Renaldy and A. Rustam, “Perancangan Sistem Informasi Inventory Berbasis Web Pada Gudang Di Pt. Spin Warriors,” *Aisyah J. Informatics Electr. Eng.*, vol. 4, no. 1, pp. 27–32, 2022.
- [27] R. S. Dias and M. Muhallim, “Sistem Informasi Penjualan Berbagai Macam Produk Berbasis Android Di Toko De Ari Palopo,” *Indones. J. Educ. Humanit.*, vol. 2, no. 1, pp. 34–50, 2022.
- [28] A. Ramadani, "Sistem Informasi Cuti Kepegawaian pada Rumah Sakit Umum Daerah Kabupaten Batu Bara," *Modem: Jurnal Informatika dan Sains Teknologi*, vol. 3, no. 1, pp. 67–75, 2025.
- [29] M. T. Abdillah, I. Kurniastuti, F. A. Susanto, and F. Yudianto, “Implementasi Black Box Testing dan Usability Testing pada Website Sekolah MI Miftahul Ulum Warugunung Surabaya,” *J. Comput. Sci. Vis. Commun. Des.*, vol. 8, no. 1, pp. 234–242, 2023, doi: 10.55732/jikdiskomvis.v8i1.897.
- [30] S. T. Welhelmina, A. P. Thenata, and B. Hakim, “Aplikasi Voting Naskah Dan Pre-Order Buku Berbasis Web Menggunakan Framework Laravel (Studi Kasus: Penerbit Loveable),” *JBASE - J. Bus. Audit Inf. Syst.*, vol. 6, no. 2, pp. 46–57, 2023, doi: 10.30813/jbase.v6i2.4674.
- [31] H. Nurfauziah and I. Jamaliyah, “Perbandingan Metode Testing Antara Blackbox Dengan Whitebox Pada Sebuah Sistem Informasi,” *J. Vis.*, vol. 8, no. 2, pp. 105–113, 2022.