

Type of
Contribution:

▶ Research Paper
Review Paper
Case Study

ENERGY: JURNAL ILMIAH
ILMU-ILMU TEKNIK
Vol. 16, No. 1 (2026) pp 88-105
DOI: 10.51747/energy.v16i1.1616



E-ISSN: 2962-2565

This article
contributes to:



9 INDUSTRY, INNOVATION
AND INFRASTRUCTURE



Development of Inventory and Goods Lending Systems Using QR Code at Universitas Muhammadiyah Kotabumi

Risanda Ardela^{1*}, Ryan Aji Wijaya¹

¹ Department information system and technology, University Muhammadiyah Kotabumi, 34517, Indonesia

*2259201069@umko.ac.id

Abstract

Asset and inventory management in higher education institutions is a crucial aspect of supporting academic operational efficiency. At Universitas Muhammadiyah Kotabumi (UMKO), the current inventory recording system and lending procedures are still conducted conventionally, leading to risks of data inaccuracy, slow information retrieval, and the potential loss of physical lending forms. This study aims to design and develop an integrated inventory and goods lending information system utilizing Quick Response (QR) Code technology. The system development method employed is the Waterfall model. This method was selected based on the characteristics of inventory systems that require structured development stages and comprehensive documentation. QR Code technology is implemented to automate item identification and accelerate lending transactions through mobile device scanning. The result of this research is a web-based application capable of providing real-time asset data, facilitating item tracking, and generating accurate inventory reports. The implementation of this system is expected to minimize human error, increase asset management transparency, and support the modernization of digital infrastructure within Universitas Muhammadiyah Kotabumi for more professional services.

Keywords: Inventory System, QR Code, Waterfall Method, Asset Management

1. Introduction

Asset management efficiency is a vital pillar in supporting the smoothness of operational and academic activities [1]. From a management perspective, this requires a systematic approach to managing the asset lifecycle, ranging from the

Article Info

Submitted:

2026-05-11

Revised:

2026-06-12

Accepted:

2026-06-18

Published:

2026-06-19



This work is
licensed under a
Creative
Commons
Attribution-
NonCommercial
4.0 International
License

Publisher

Universitas
Panca Marga

stages of planning, organizing, utilizing, and maintaining to the disposal process, ensuring that each asset is optimally managed throughout its useful life [2]. In the current era of digital transformation, the availability of accurate and real-time inventory data has become an absolute necessity for university management to conduct oversight and strategic decision-making [3]. However, in practice, many institutions still face obstacles in synchronizing physical item counts with administrative records, which often triggers inefficiencies in asset maintenance and procurement [4].

Universitas Muhammadiyah Kotabumi (UMKO), as an evolving educational institution, manages a high volume of assets and a significant intensity of goods lending. Currently, the inventory recording process and lending procedures at UMKO tend to be conducted conventionally or through manual bookkeeping, which poses a high risk of human error. The primary obstacles frequently encountered include the risk of losing lending forms, difficulties in rapid item tracking, and a time-consuming monthly report recapitulation process due to decentralized data.

To address these issues, a technological innovation in the form of an integrated inventory information system utilizing Quick Response (QR) Code instruments is required. This technology is capable of transforming textual data into an efficient two-dimensional code format and is designed to facilitate rapid data retrieval through electronic devices such as smartphones [5]. By embedding QR Codes on every asset, the item identification process and lending transactions can be executed automatically and accurately while minimizing manual data entry errors, thereby enhancing asset management transparency within Universitas Muhammadiyah Kotabumi.

Previous studies have explored the implementation of inventory management systems using various technologies and approaches. Developed a QR Code-based asset tracking system that focuses on general inventory logging through a web platform [6]. While the system successfully facilitates asset identification and tracking, it lacks a dedicated role-based permission mechanism, allowing inventory data to be accessed through publicly scannable QR codes. Another study [7] proposed a web-based inventory management system with barcode tracking functionality; however, the system relies on a static database structure and does not support real-time decentralized lending transactions or automatic PDF label generation. To address these limitations, the proposed UMKO Inventory System introduces a high-security internal scanning mechanism that restricts data access exclusively to authorized staff and administrators. In addition, the system incorporates an integrated lending transaction module and an automated digital PDF label generator, thereby improving inventory control, security, and operational efficiency.

In developing this system, the researcher implements the Waterfall Method as the software development framework. The selection of this method is based on the characteristics of the inventory system requirements, which are clearly defined and static. The systematic approach of the Waterfall model enables application development to proceed in a structured and well-documented manner [8]. To guarantee functional quality, the system will be validated through Blackbox testing. This testing focuses on the external functions of the software through input and output scenarios to ensure that every command generates the correct and consistent response according to the initial design, without requiring an understanding of the system's internal process complexities [9].

This research produces an application titled "Inventory and Goods Lending System Using QR Code," which is capable of automating the infrastructure management workflow at Universitas Muhammadiyah Kotabumi. With this system, it is expected that the asset monitoring process will become more controlled, lending procedures will become more practical for the academic community, and inventory reporting can be generated instantly to support more professional service standards.

2. Methods

This study adopts a methodological approach that combines several data collection techniques, specifically observation, interviews, and literature review. Meanwhile, the system development itself is conducted by implementing the Waterfall method.

2.1 Data Collection Methods

a. Observation

Observation is a planned sensory-based monitoring process [10]. Based on field observations, it was evident that the asset management process in the infrastructure department still relies on recording in ledgers and accumulated physical form sheets, making it difficult to retrieve item data quickly.

b. Interview

interview is a dynamic interaction process involving direct verbal communication between the researcher (interviewer) and the informant (resource person). Through this face-to-face dialogue, an exchange of information occurs aimed at gathering in-depth data from primary sources [11]. The results of the interviews with the inventory management staff indicated that the primary obstacles frequently encountered are data duplication and the loss of lending history due to an unintegrated administrative system.

c. Literature review

Literature review is the process of gathering information through the review of various written sources, ranging from books to technical reports. The primary focus of this method is to explore documents and records that have direct relevance to the research problem in order to gain a comprehensive understanding [12].

2.2 System Development Method

This study applies the Waterfall method (in **Figure 1**) as the software development model. The Waterfall model consists of several sequential phases, namely Requirements Analysis, System Design, Implementation (Coding), Testing, Deployment, and Maintenance [13]. The process begins with requirements analysis through observation and interviews to identify inventory management problems at UMKO. The identified requirements are then translated into system and database designs using UML. Next, the design is implemented into a web-based inventory application with QR Code functionality. After implementation, the system is tested using Black Box Testing to ensure that all features operate according to the requirements. The system that passes the testing phase is then deployed for use by authorized staff, followed by maintenance activities to fix errors and improve system performance.

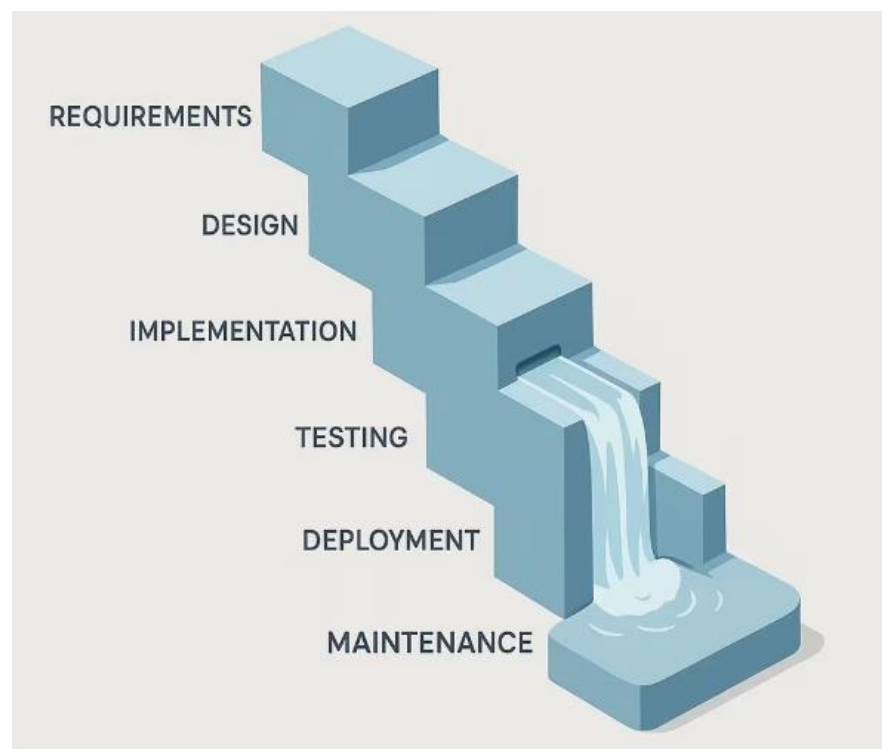


Figure 1. Waterfall method

3.Results and Discussion

3.1 UML System Modeling

Unified Modeling Language (UML) is a graphical modeling standard used to describe software blueprints through visual representations. The primary functions of UML include visualizing structures, specifying requirements, constructing models, and comprehensively documenting all elements within a software system [14].

3.1.1 Use Case

A Use Case Diagram is a modeling instrument that functions to visualize the functional behavior of the system under development. This diagram specifically represents the interactions occurring between one or more external actors and the services provided by the system [15].

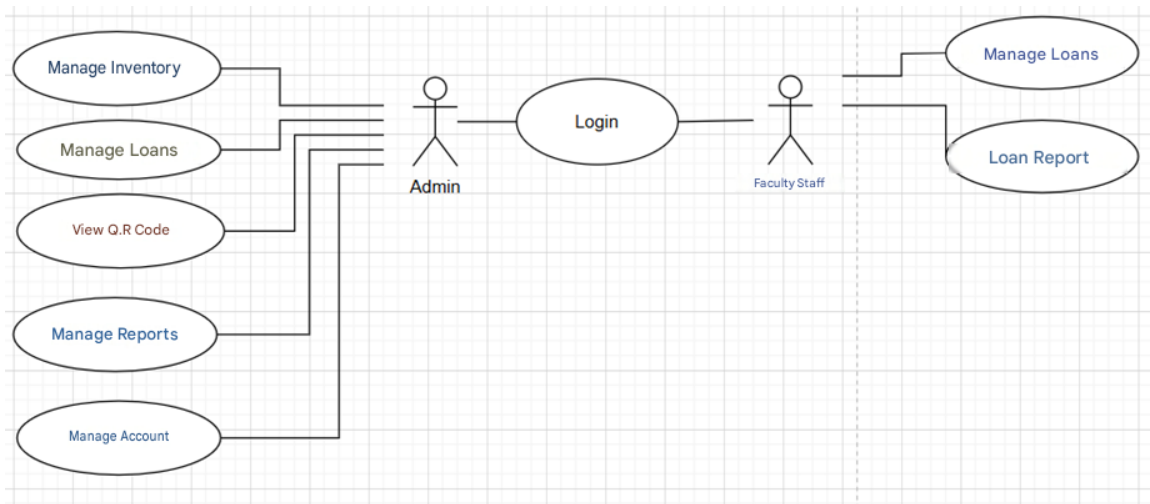


Figure 2. Use Case diagram concept

The Use Case Diagram (in **Figure 2**) illustrates the inventory and lending management system involving two primary actors, namely the Admin and Faculty Staff, both of whom are required to undergo a login process to access the system. The Admin holds full authority, which encompasses managing inventory, lending transactions, reports, user accounts, and a specific feature for viewing QR Codes. Conversely, the Faculty Staff has more restricted access focused on daily operations, specifically managing lending transactions and viewing lending reports only. Overall, this diagram demonstrates a clear division of access rights between the system administrator and operational users in handling the inflow and outflow of goods or facilities.

3.1.2 Activity Diagram

Activity diagrams (in **Figure 3**) are used to represent the workflow or activities that occur dynamically. The primary focus of this diagram is to provide a visual

overview of the procedures and operational steps performed by the system in executing its functions [16].

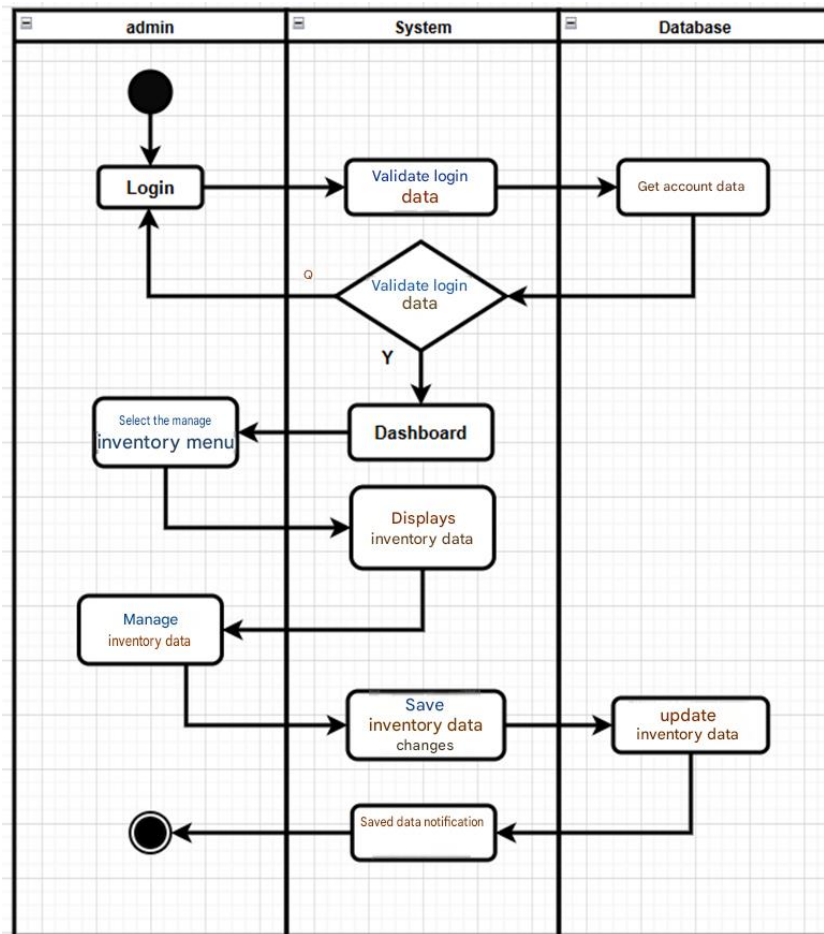


Figure 3. Inventory Management Activity Diagram

The activity diagram above illustrates the inventory management workflow, where the process begins with the Admin performing a login that is validated by the system against the database. Upon successfully accessing the dashboard, the Admin can manage the inventory by updating item data in the database. The entire sequence of activities concludes with the system providing a success notification to the user after the data changes have been permanently saved to the database.

3.1.3 Sequence Diagram

A sequence diagram (in [Figure 4](#)) is a type of UML diagram that specifically visualizes the interaction between objects and the exchange of messages within a specific timeframe. This diagram plays a crucial role in mapping the activities found in a use case into technical operations within the object classes in a class diagram [17].

The Sequence Diagram for Lending Reports above illustrates the systematic interaction when the Faculty Staff accesses report data. The process begins with an access request for the report menu from the actor to the System, which is then followed by a data retrieval instruction (get data) to the Database. After the Database

returns the relevant collection of report data, the System processes and presents that information to the user in the form of a report list display. The entire sequence concludes with a visual confirmation or notification that the report has been successfully loaded and is ready for review by the user.

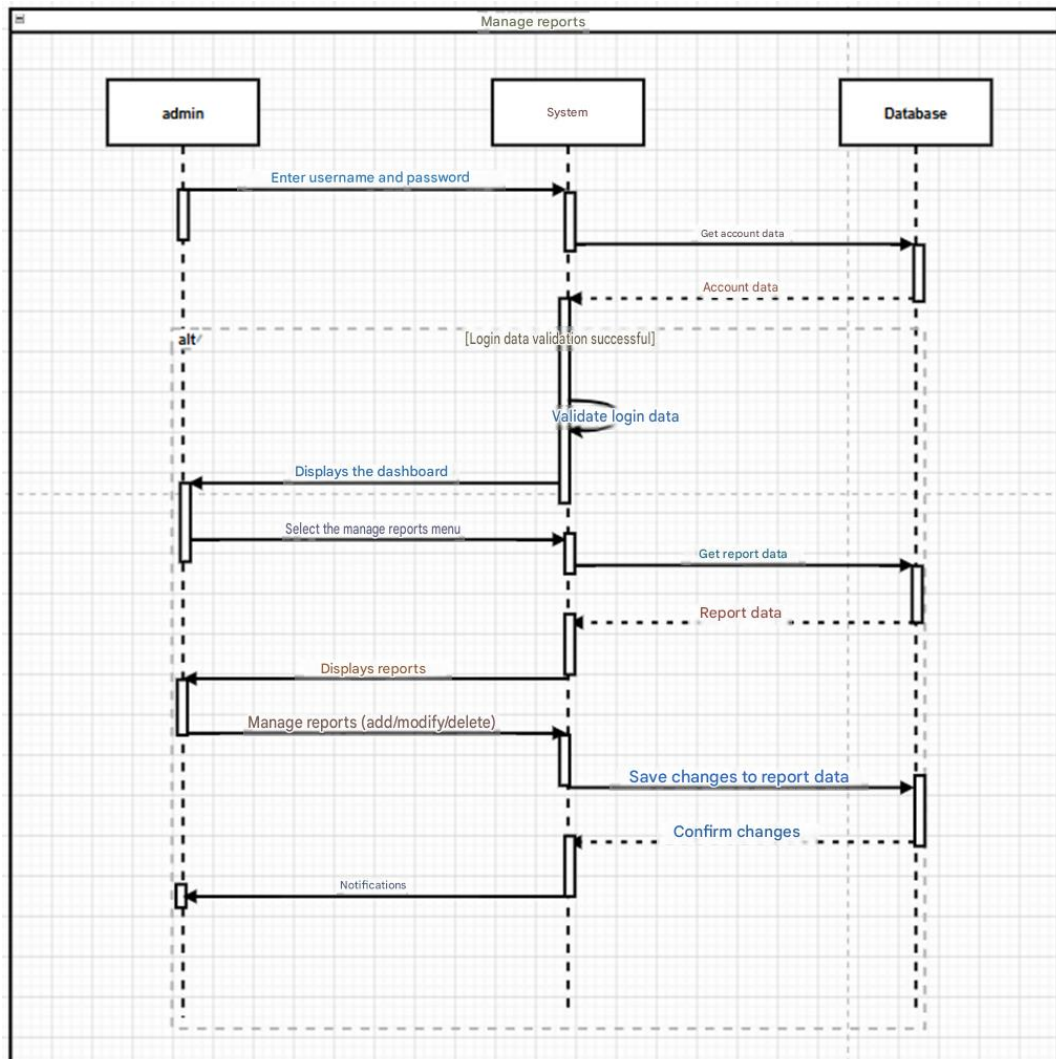


Figure 4. Sequence Diagram for Reporting

3.1.4 Database

The Entity Relationship Diagram (ERD) underpinning the inventory and lending management system consists of seven primary entities: Faculty, Admin, Inventory, Lending, Brand, Building, and Room (in [Figure 5](#)). The core of these relationships centers on the Inventory table, which stores technical asset details such as item codes, brands, and conditions. This table is directly linked to the Lending table to track the circulation of goods, covering statuses from application to return. Furthermore, this data structure organizes the location hierarchy through the interconnectedness of the Building, Room, and Faculty tables, while the Admin table records the management of user authorization responsible for overseeing the asset data.

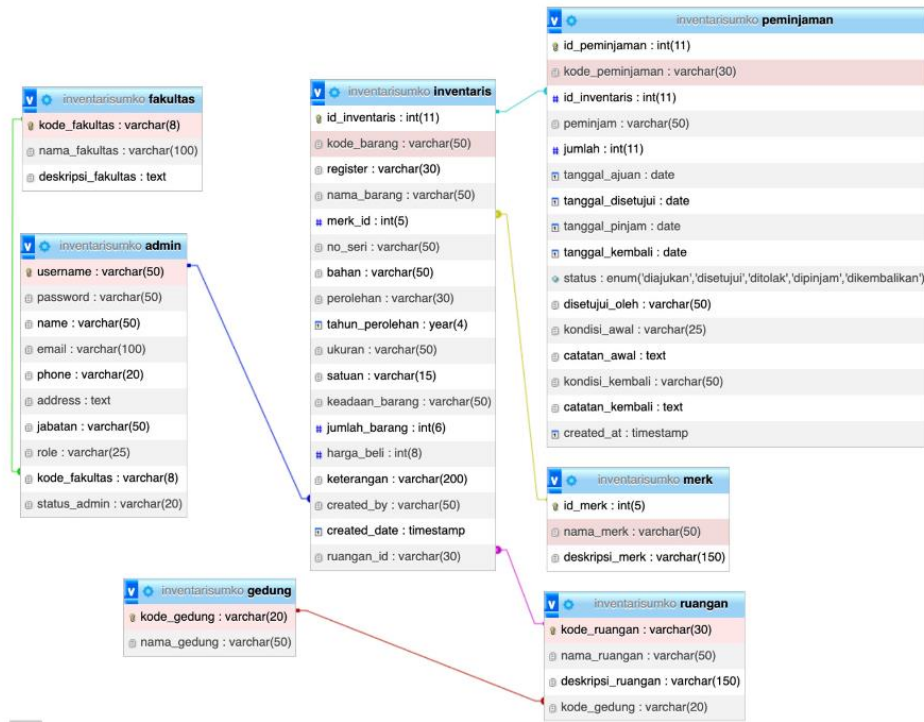


Figure 5. Database system management

3.2 Implementation

The implementation of the integrated inventory information system utilizing QR Code instruments at Universitas Muhammadiyah Kotabumi (UMKO) yields significant advancements in two critical parameters: operational efficiency and the reduction of human error.

From an operational perspective, the transition from conventional bookkeeping to a QR Code-based digital system radically accelerates the asset lifecycle management. In the previous manual workflow, staff spent substantial time manually locating physical ledger entries, verifying asset numbers, and filling out physical lending forms. By integrating QR Codes, the time required for asset identification and lending logging is reduced from several minutes to a few seconds per item. A simple scan via an authorized smartphone instantly retrieves or updates the asset status in the cloud database. Furthermore, the decentralized data nature of the proposed system optimizes the monthly report recapitulation. What previously took days of cross-referencing paper forms can now be generated instantaneously via the system's automated report module, thereby enhancing institutional agility and strategic decision-making readiness for the university management.

Human error in conventional inventory tracking remains a primary source of data discrepancy and asset vulnerability. The manual entry of long, complex alphanumeric strings such as the Kode Barang and No Register is highly susceptible to typos, misreadings, and omissions. The proposed system mitigates this risk by embedding these data structures directly into a two-dimensional QR Code format.

This automation ensures 100% accuracy during data retrieval and transaction logging, as the system eliminates the need for manual data entry at the point of action. Additionally, the risk of losing physical lending forms is entirely neutralized, as all lending histories and user liabilities are recorded digitally and securely within the internal database. By restricting scanning capabilities strictly to authorized staff and administrators, the system prevents unauthorized data tampering, maintaining absolute data integrity throughout the asset's useful life.

a. Dashboard interface

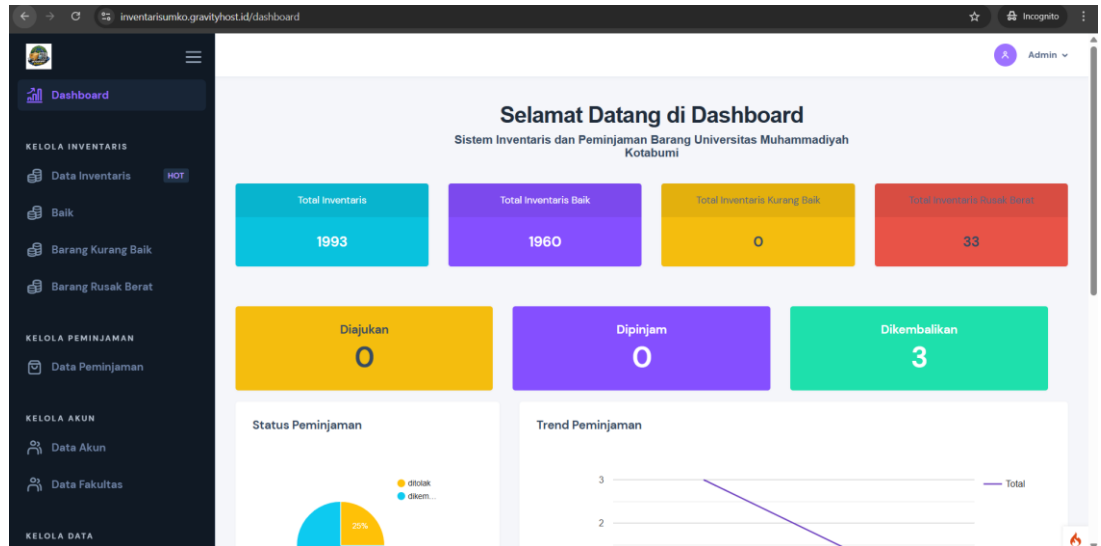


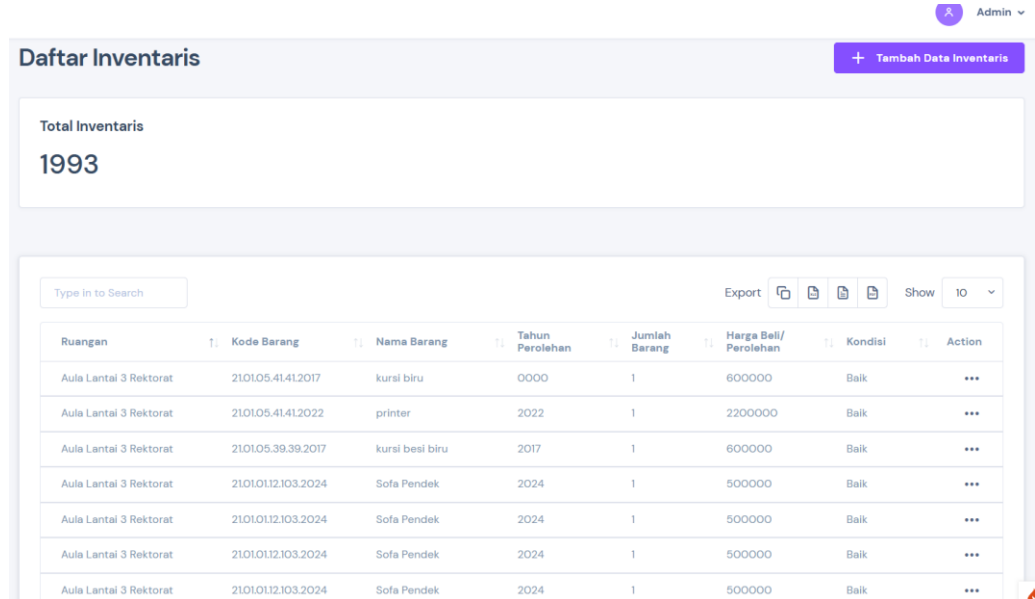
Figure 6. Dashboard Interface page of the application

The Dashboard of the Inventory and Lending System at Universitas Muhammadiyah Kotabumi (in [Figure 6](#)) serves as an integrated center for monitoring assets and lending transactions. It provides a statistical summary of the inventory, encompassing total items (1,993), goods in good condition (1,960), and severely damaged items (33). Furthermore, the dashboard presents real-time data regarding lending statuses, such as the number of pending applications (0), items currently borrowed (0), and returned items (3). Featuring a comprehensive sidebar navigation for managing inventory data, faculties, and user accounts, this interface is designed to assist administrators in monitoring campus logistical trends and operational statuses within a single, informative visual display.

b. Inventory List Interface

The Inventory List page (in [Figure 7](#)) presents a detailed asset database with a total of 1,993 recorded items. The primary table displays comprehensive information for each asset, including room location, item code, name, acquisition year, quantity, purchase price, and the condition status, the majority of which are recorded as being in 'Good' condition. This interface is also equipped with managerial features such as an 'Add Inventory Data'

button, a search function, data export options, and an action menu in each row to facilitate regular updates or maintenance of asset data.

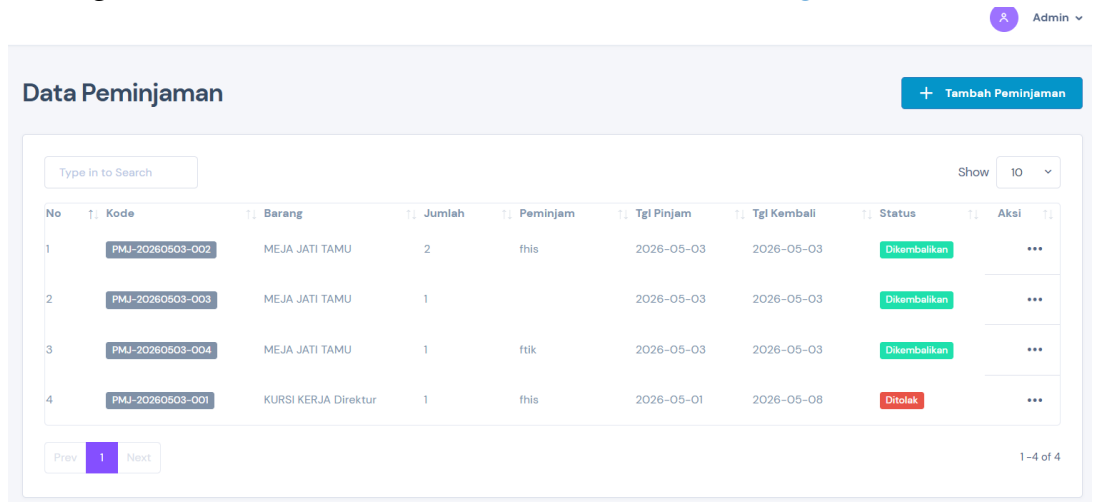


Ruang	Kode Barang	Nama Barang	Tahun Perolehan	Jumlah Barang	Harga Beli/Perolehan	Kondisi	Action
Aula Lantai 3 Rektorat	210105.41.41.2017	kursi biru	0000	1	600000	Baik	...
Aula Lantai 3 Rektorat	210105.41.41.2022	printer	2022	1	2200000	Baik	...
Aula Lantai 3 Rektorat	210105.39.39.2017	kursi besi biru	2017	1	600000	Baik	...
Aula Lantai 3 Rektorat	21010112.103.2024	Sofa Pendek	2024	1	500000	Baik	...
Aula Lantai 3 Rektorat	21010112.103.2024	Sofa Pendek	2024	1	500000	Baik	...
Aula Lantai 3 Rektorat	21010112.103.2024	Sofa Pendek	2024	1	500000	Baik	...
Aula Lantai 3 Rektorat	21010112.103.2024	Sofa Pendek	2024	1	500000	Baik	...

Figure 7. Inventory Page Interface of the application

c. Borrower List Interface

The Lending Data module functions to manage and monitor all inventory lending activities in a structured manner, as shown in **Figure 8**.



No	Kode	Barang	Jumlah	Peminjam	Tgl Pinjam	Tgl Kembali	Status	Aksi
1	PMJ-20260503-002	MEJA JATI TAMU	2	fhis	2026-05-03	2026-05-03	Dikembalikan	...
2	PMJ-20260503-003	MEJA JATI TAMU	1		2026-05-03	2026-05-03	Dikembalikan	...
3	PMJ-20260503-004	MEJA JATI TAMU	1	ftik	2026-05-03	2026-05-03	Dikembalikan	...
4	PMJ-20260503-001	KURSI KERJA Direktur	1	fhis	2026-05-01	2026-05-08	Ditolak	...

Figure 8. Borrower page interface of the application

d. Brand List Interface

The Brand List interface serves as a centralized hub for managing brand data of inventory items, currently recording a total of 53 registered brands, as shown in **Figure 9**. This interface presents a table containing brand names, such as 'acer', along with their descriptions to facilitate asset categorization. The Admin can efficiently manage this database through search features, data export options to various formats, and an '+ Add Brand' button to update the brand list according to the university's new inventory requirements.

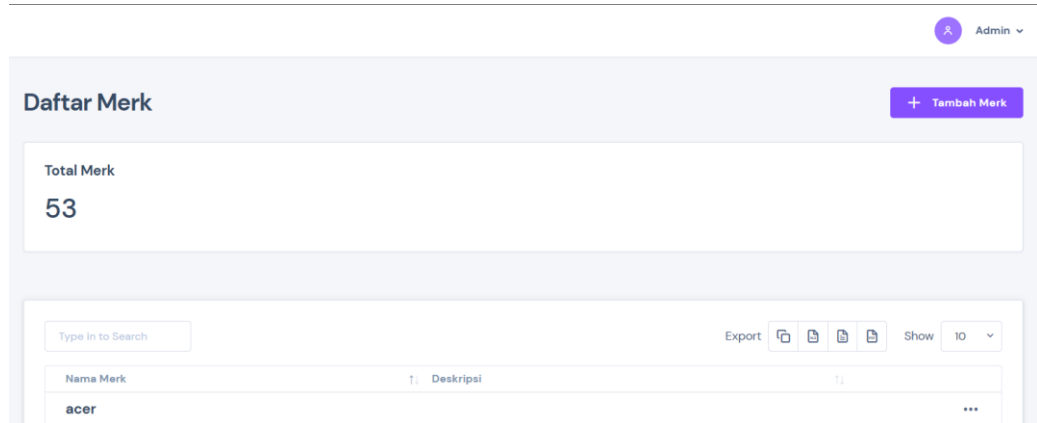


Figure 9. Brand List page interface of the application

e. QR Code

The digital-based inventory system dashboard page (KIB/Asset Inventory Card), as shown in **Figure 10**. The QR Code and all detailed asset information displayed on this page are strictly internal. Only authorized staff or administrators have access to log into the system, view this data, or scan the code to manage the inventory. External users or the public cannot access the information herein

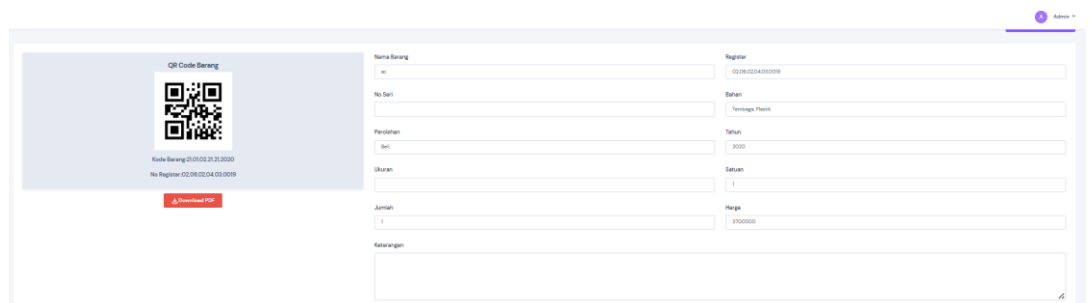


Figure 10. QR Code result information

3.3 Testing

3.3.1 Black-box Testing

Black-box testing is conducted to ensure that all functions (features) within the application operate correctly (in **Tabel 1**). The Functional Testing Results demonstrate that all system modules, including Dashboard, Inventory Management, Lending Module, QR Scanner, Brand Management, and Report Module, achieved a 100% success rate. This indicates that every functional requirement defined in the system specification has been successfully implemented without detectable functional failure.

The uniform success rate across all modules reflects high functional stability and consistency of the system. In particular, critical modules such as QR Scanner and Lending Module show strong reliability, confirming that real-time asset tracking and transaction processes operate correctly. From a software engineering

perspective, these results indicate that the system satisfies functional correctness criteria and is ready for operational deployment in real-world academic environments, as shown in **Figure 11**.

Tabel 1. Blackbox Testing Result

No	Module	Test Scenario	Expected Result	Outcome
1	Dashboard	Access system dashboard	System displays inventory statistics	Passed
2	Inventory Management	Add new item data	Data stored in database correctly	Passed
3	Lending Module	Create lending transaction	Transaction recorded successfully	Passed
4	QR Code Scanner	Scan asset QR code	System redirects to asset detail	Passed
5	Brand Management	Add new brand	Brand saved into system	Passed
6	Report Module	Generate report	System generates valid report output	Passed

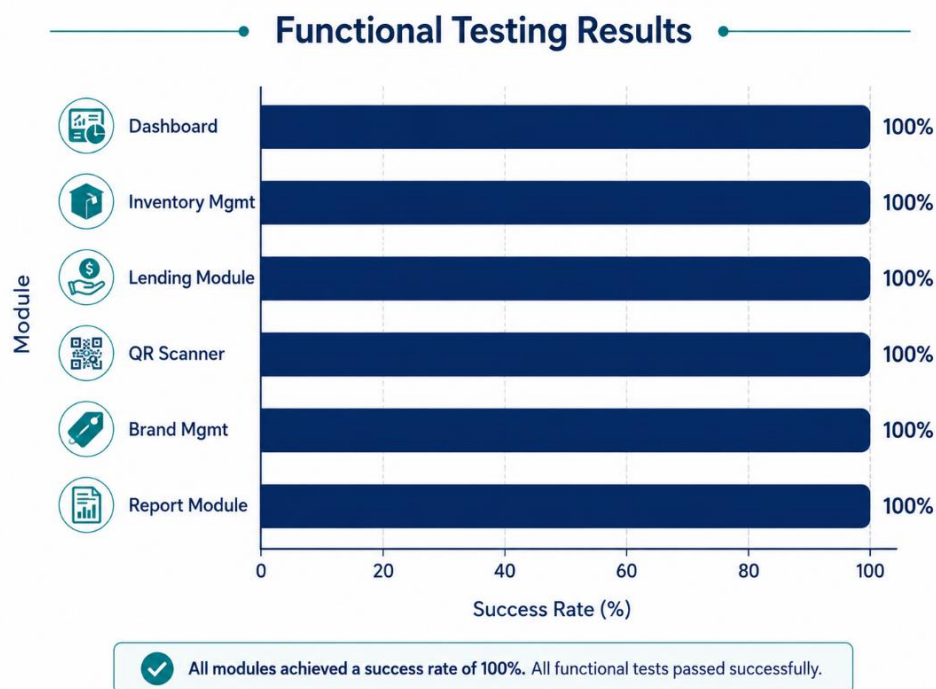


Figure 11. Functional Testing Result

3.3.2 Structural Validation (White-Box Testing)

White-box testing was applied using basis path testing supported by cyclomatic complexity analysis to measure internal logical structure (in **Tabel 2**).

Cyclomatic complexity is calculated using:

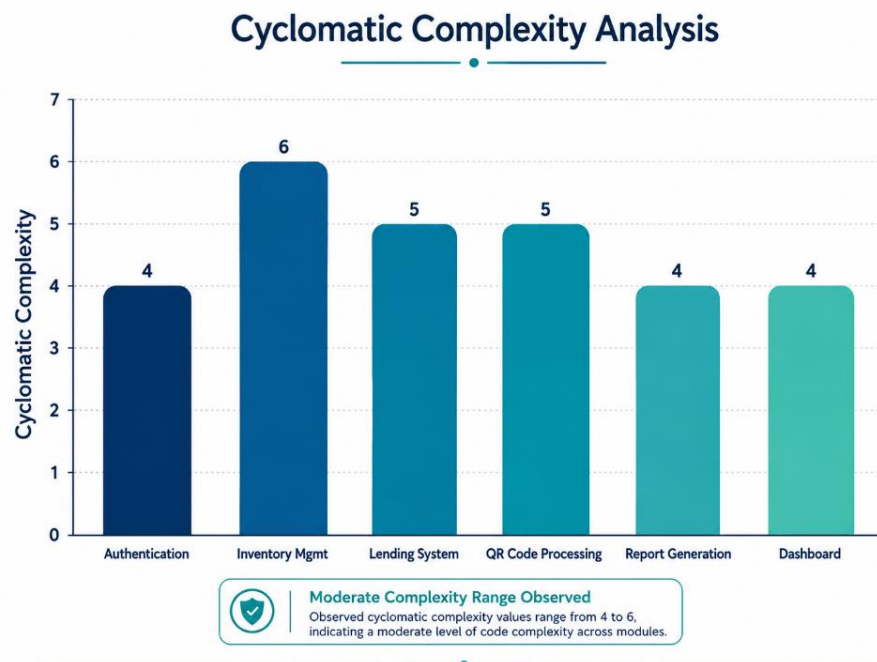
$$V(G) = P + 1$$

where P represents decision nodes in the program.

Tabel 2. Cyclomatic Complexity Analysis

Module	Decision Nodes (P)	Complexity V(G)	Independent Paths	Result
Authentication	3	4	4	Valid
Inventory Management	5	6	6	Valid
Lending System	4	5	5	Valid
QR Code Processing	4	5	5	Valid
Report Generation	3	4	4	Valid
Dashboard	3	4	4	Valid

The results indicate that all modules fall within the acceptable complexity threshold ($V(G) \leq 6$), confirming that the system logic is well-structured and maintainable.

**Figure 12.** Cyclomatic Complexity Chart Analysis

The Cyclomatic Complexity Analysis (in **Figure 12**) shows that the system modules fall within a moderate complexity range ($V(G) = 4-6$). The Inventory Management module has the highest complexity value (6), followed by Lending System and QR Code Processing modules (5). This level of complexity is considered acceptable in software engineering practice, as it indicates a balance between functionality and maintainability. The higher complexity in Inventory Management is caused by multiple decision branches such as item validation, condition checking, and transaction verification. Meanwhile, other modules such as Authentication and Dashboard maintain lower complexity values, indicating simpler control flow structures. Overall, the results confirm that the system is structurally well-designed,

with manageable logic complexity that supports maintainability, scalability, and future system enhancement.

3.3.3 Code Coverage and Test Effectiveness

Code coverage analysis (in [Tabel 3](#) and [Figure 13](#)) was performed to evaluate the completeness of testing execution across system modules. The metrics used include: Statement Coverage, Branch Coverage, Path Coverage.

Tabel 3. Code Coverage Results

Module	Statement (%)	Branch (%)	Path (%)
Authentication	96	93	91
Inventory Management	95	92	90
Lending System	94	90	88
QR Code Module	93	89	87
Report Module	92	88	86
Dashboard	95	91	90

The system achieved an average coverage above 90%, indicating high test completeness and strong validation of execution paths.

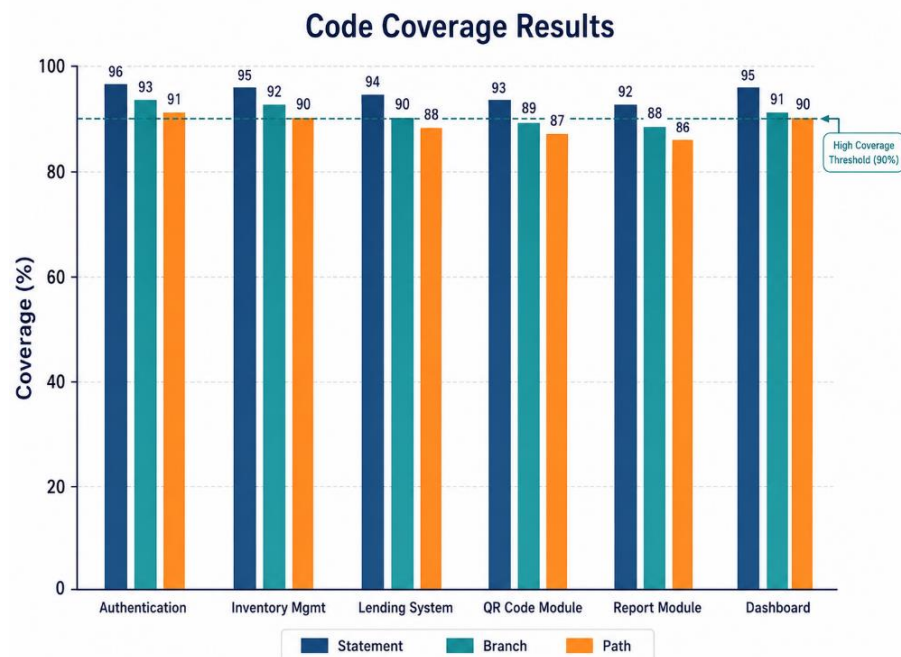


Figure 13. Code Coverage

The Code Coverage Results indicate a high level of test completeness across all modules, with statement coverage ranging from 92% to 96%, branch coverage from 88% to 93%, and path coverage from 86% to 91%. The highest coverage is observed in the Inventory Management and Authentication modules, indicating that most execution paths, including edge cases, have been successfully tested. The QR Code

Module shows slightly lower coverage, which is typical due to external dependency factors such as hardware scanning behavior and runtime environment variability. Overall, all modules exceed the commonly accepted threshold of 85%, demonstrating that the system has been thoroughly tested and possesses strong reliability and robustness in execution.

3.3.4 Reliability and Maintainability Evaluation

To further strengthen software quality evaluation, this study maps testing outcomes into software quality attributes based on ISO/IEC 25010.

Key findings:

- Reliability: High, due to successful execution of all functional and structural test cases.
- Maintainability: Moderate to high, indicated by controlled cyclomatic complexity (4–6).
- Efficiency: High, as QR Code scanning reduces asset identification time significantly.
- Data Integrity: Strong, due to relational constraints and role-based access control.

3.3.5 Comparative Analysis of Testing Approaches

Table 4. Testing Comparison Summary

Testing Type	Objective	Result	Success Rate
Black-Box Testing	Functional validation	All modules passed	100%
White-Box Testing	Internal logic validation	All paths valid	100%
Coverage Testing	Execution completeness	Above 90% average	High
Quality Assessment	Software attributes	Reliable system	Strong

The consistency between black-box and white-box results indicates that the system is both functionally correct and structurally stable. This alignment strengthens the validity of the system for real-world deployment.

The Testing Comparison Summary (in [Table 4](#)) provides a holistic evaluation of system quality by integrating functional testing, structural testing, coverage analysis, and software quality assessment. Both Black-box and White-box testing achieved a 100% success rate, indicating full alignment between system functionality and internal logic correctness. Coverage Testing shows a high execution completeness level (>90%), while the Quality Assessment confirms that the system is reliable and stable.

This consistency across multiple testing dimensions indicates strong software integrity. The alignment between external behavior (black-box) and internal structure (white-box) further confirms that the system is free from major logical

inconsistencies and is suitable for deployment in institutional asset management environments, as shown in [Figure 14](#).

Testing Comparison Summary









 Testing Type	 Objective	 Result	 Success Rate
 Black-box Testing	Functional validation	All modules passed	100%
 White-box Testing	Internal logic validation	All paths valid	100%
 Coverage Testing	Execution completeness	Above 90% average	High
 Quality Assessment	Software attributes	Reliable system	Strong

Figure 14. Testing Comparison Result Summary

3.3.6 Discussion

The comprehensive testing framework confirms that the proposed inventory and lending system demonstrates strong software engineering quality. The combination of functional and structural testing ensures that both external behavior and internal logic are validated. Cyclomatic complexity results indicate that all modules remain within maintainable boundaries, preventing excessive control flow complexity. Furthermore, high code coverage values (>90%) confirm that most execution paths have been tested, reducing the likelihood of hidden runtime defects.

From a system engineering perspective, the implementation of QR Code technology significantly enhances operational efficiency by reducing asset identification time and eliminating manual recording errors. This contributes directly to improved data integrity and transparency in institutional asset management. Overall, the system demonstrates high readiness for deployment in higher education environments, particularly in asset-intensive administrative contexts.

4. Conclusion

Based on the research results and implementation, it can be concluded that the development of this web-based Inventory and Lending System successfully provides an effective digital solution for managing university assets in a transparent and organized manner. The integration of features such as item condition monitoring, brand management, and real-time lending status recording has proven capable of minimizing the risk of data loss and accelerating administrative reporting

processes. The results of the black-box testing confirm that all system functionalities, ranging from inventory data processing to transaction status validation, operate in accordance with user requirements. Consequently, this system is highly suitable for implementation to enhance operational efficiency and accountability in logistics management within higher education environments.

Acknowledgement

The authors would like to acknowledge the support from the Electrical Engineering Laboratory, Universitas Islam Malang, to run the experiments of the research.

Authors' Declaration

Authors' contributions and responsibilities - The authors made substantial contributions to the conception and design of the study. The authors took responsibility for data analysis, interpretation, and discussion of results. The authors read and approved the final manuscript.

Funding - No funding information from the authors.

Availability of data and materials - All data is available from the authors.

Competing interests - The authors declare no competing interest.

Additional information - No additional information from the authors.

References

- [1] S. Purnomoa dan F. A. Alijoyo, "Sistem Peminjaman Barang Menggunakan QR Code Berbasis Aplikasi Android," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 6, no. 2, hal. 322–328, 2024.
- [2] A. Pratiwi dan A. Mayansara, "Evaluasi Penerapan Sistem Informasi Manajemen Aset dan Tata Ruang di Rumah Sakit : Tinjauan Literatur," vol. 2, no. 5, hal. 244–253, 2025.
- [3] D. N. Arahman, Y. Sholva, M. Muthahhari, A. B. P. Negara, dan H. Anra, "Sistem Peminjaman dan Monitoring Ruangan dengan Fitur QR Code Berbasis Web dan Mobile Android," vol. 7, no. 1, hal. 271–285, 2025.
- [4] R. C. Mandala dan A. Susanto, "Pengembangan Sistem Inventaris Barang Berbasis QR Code pada Badan Kepegawaian Daerah Provinsi Bengkulu," *jurnal.pustakagalerimandiri*, vol. 3, no. 1, hal. 47–51, 2023.
- [5] A. Alfariy, "Literature Review Internet of Things (IOT): QR Code," *J. Jar. Komput. dan Keamanan*, vol. 04, no. 03, hal. 25–29, 2023.
- [6] R. F. Fajriyah, F. A. Tyas, dan A. Basir, "APLIKASI LABELING DAN TRACKING ASET MENGGUNAKAN QR CODE," *J. Tek. Inform. dan Sist. Inf.*, vol. 2, no. 1, hal. 34–40, 2022.
- [7] A. Budiman dan I. Ariyati, "IMPLEMENTATION OF A QR CODE-BASED WEB INVENTORY SYSTEM AT PT SELAHONJE JAYA ABADI BEKASI," *J. Sci. Res. Dev.*, vol. 8, no. 1, hal. 255–265, 2026.

- [8] Z. Hamzah dan Turahyo, "SISTEM INVENTORY TOOLS MENGGUNAKAN QR-CODE (Studi Kasus : PT. Pupuk Kaltim)," *J. Sains dan Sist. Teknol. Inf. (SANDI)*, vol. 6, no. 1, hal. 29–34, 2024.
- [9] M. D. H. Sandy, "Pengembangan Aplikasi Laundry Berbasis Android di Wilayah Kota Pekanbaru," vol. 2, no. 2, hal. 159–173, 2025.
- [10] B. Ocsis, S. Sauri, dan Y. P. Putra, "Ilmu dalam Tinjauan Filsafat yang Berkaitan Epistemologi," *J. Islam. Stud.*, vol. 7, no. 4, hal. 1261–1274, 2024, doi: 10.31943/afkarjournal.v7i4.1112.Science.
- [11] M. R. Kurniarullah, T. Nabila, A. Khalidy, V. J. Tan, dan H. Widiyani, "Tinjauan Kriminologi Terhadap Penyalahgunaan Artificial Intelligence: Deepfake Pornografi dan Pencurian Data Pribadi," *J. Ilm. Wahana Pendidik.*, vol. 10, no. 10, hal. 534–547, 2024.
- [12] B. A. Habsy, N. Mufidha, C. Shelomita, I. Rahayu, dan M. I. Muckorobin, "Filsafat Dasar dalam Konseling Psikoanalisis : Studi Literatur," vol. 7, no. 2, hal. 189–199, 2023, doi: 10.30653/001.202372.266.
- [13] Angelina, C. Yandhika, C. L. Hartanto, M. Graciela, dan A. Farisi, "Sebuah Tinjauan Literatur Sistematis Tentang Metode Pengembangan Perangkat Lunak Sistem Informasi Berbasis Web," vol. 5, no. 1, hal. 181–192, 2024.
- [14] T. M. Fawaati dan H. Sukri, "APPLICATION OF ARTIFICIAL INTELLIGENCE (AI) IN SEARCH ENGINE OPTIMIZATION (SEO)," *J. Alih Teknol. Sist. Inf.*, vol. 2, no. 1, 2022.
- [15] A. M. KURNIAWAN, "SISTEM INFORMASI PRAKTEK KERJA LAPANGAN BERBASIS WEBSITE PADA SMK NEGERI 5 SURAKARTA." 2022.
- [16] H. Kurniawan, W. Apriliah, I. Kurniawan, dan D. Firmansyah, "PENERAPAN METODE WATERFALL DALAM PERANCANGAN SISTEM INFORMASI PENGGAJIAN PADA SMK BINA KARYA KARAWANG," vol. 14, no. 4, hal. 13–23, 2021.
- [17] M. R. Wayahdi dan F. Ruziq, "Pemodelan Sistem Penerimaan Anggota Baru dengan Unified Modeling Language (UML) (Studi Kasus : Programmer Association of Battuta)," *J. Minfo Polgan*, vol. 12, no. 1, hal. 1514–1521, 2023.