Indonesian Journal of Multidisciplinary on Social and Technology
Homepage: https://journal.ilmudata.co.id/index.php/ijmst
Vol. 5 No. 1 (2026) pp: 433-443
P-ISSN: 2986-6790, e-ISSN: 2986-6782

Development of a Web-Based Inventory Damage Reporting System Using the Prototype Model

Hidayah Herliyani, Mutamassikin
Prodi Sistem Informasi, Fakultas, Universitas Islam Negeri Sultan Thaha Saifuddin Jambi
hidayah.herliyani10@gmail.com, mutamassikin@uinjambi.ac.id

Abstract

This study focuses on the design of a web-based Inventory Damage Reporting Information System at the Faculty of Science and Technology, UIN Sulthan Thaha Saifuddin Jambi. This research is motivated by the existing damage reporting process, which is still conducted verbally, through private messaging, WhatsApp groups, and manual data logging, thereby potentially causing delays in information delivery, difficulties in documentation, and sub-optimal monitoring and follow-up processes. This study aims to design and build an information system that can assist in the reporting, documentation, verification, and monitoring of inventory damage in a more structured and documented manner. This research utilizes the Prototype system development model, which enables iterative interaction and evaluation between developers and users. Data collection was conducted through observation, interviews, documentation studies, and literature reviews. The system was developed as a web-based application using the Laravel Framework with a MySQL database. System testing was conducted using Black Box Testing to ensure system functions run according to requirements, along with structured evaluative interviews at each prototype release stage to evaluate feature suitability based on user feedback. The designed system provides damage reporting features with information on item type, location, description, damage category, and photographic evidence, as well as features for report status verification and updates, inventory data management, report monitoring, and centralized damage history storage. This system is expected to help increase effectiveness, efficiency, and orderliness in the reporting process and management of inventory damage at the Faculty of Science and Technology, UIN Sulthan Thaha Saifuddin Jambi.

Keywords: Information System, Damage Reporting, Inventory, Web-Based, Prototype, Laravel, Black Box Testing.

1. Introduction

Inventory assets support teaching, learning, administration, and other academic activities in higher education institutions. Their usefulness depends not only on procurement and recording but also on the timely identification and handling of damage. Web-based inventory information systems have been used to improve the accuracy and accessibility of asset information in educational and institutional settings [1][2].When damage reports are managed manually, information may be incomplete, delayed, or difficult to retrieve, which can affect the follow-up of damaged assets.

At the Faculty of Science and Technology, UIN Sulthan Thaha Saifuddin Jambi, damage reports were commonly conveyed verbally, through personal messages, WhatsApp class groups, or manual records. These channels were not integrated into a centralized reporting record. As a result, reports could be mixed with other conversations, become difficult to trace, and lack consistent information about the damaged item, its location, and the progress of its handling. The operational process also involved different parties: the person who first noticed the damage, the inventory and facilities administrator, and the faculty official responsible for monitoring the reports.

Previous studies have addressed web-based inventory management and prototyping in several settings.[3] developed a prototype for state-owned inventory at Universitas Jambi, while Cipto [4] applied the Prototype method to inventory management in a vocational school. [5] also discussed the use of a prototype model for an inventory information system. Other studies have applied web-based asset management or inventory systems using different methods and contexts [6][7][8] These studies provide relevant approaches to inventory digitization, but the present study addresses a more specific operational issue: reporting damaged inventory items, attaching photographic evidence, verifying reports, and monitoring repair status in a university faculty environment.

The references reviewed in the thesis also include studies of web-based inventory implementation and alternative inventory-management approaches [9][10] , as well as work on software testing and web application development [11][12][13] Other web application studies cited in the thesis address digital medical records and e-catalog applications These works provide broader technical context, while the present research remains focused on the faculty inventory-damage reporting process.

The distinction of this study lies in combining a damage-reporting workflow with role-based access for students, the facilities administrator, and the Head of Subdivision (Kasubag), together with iterative user evaluation. The system was designed to record reports in a structured form and maintain their status and history. The Prototype model was selected because user requirements were not fully documented at the beginning and therefore needed to be refined through direct feedback. The research question was how a web-based information system could be designed and developed to support the reporting, verification, documentation, and monitoring of damaged inventory at the Faculty of Science and Technology. Accordingly, this study aimed to develop the system and evaluate its functional operation and user acceptance.

2. Research Methods

2.1. Research Location and Period

The research was conducted at the Faculty of Science and Technology, UIN Sulthan Thaha Saifuddin Jambi. The faculty was selected as the research object because its rooms and laboratories use various inventory assets that require reporting and handling when damage occurs. The research activities took place from January to June 2026, covering data collection, literature study, system design, prototype development, system testing, and report preparation.

2.2. Data Collection

Data were collected through observation, interviews, documentation studies, and literature reviews. Observation was used to examine the ongoing inventory-management and damage-reporting process, including how lecturers and students communicated complaints, to whom reports were submitted, which communication channels were used, and how follow-up was conducted. Interviews were conducted with inventory-management staff, lecturers, technicians at the faculty, and inventory-management staff at the university rectorate. The interviews explored operational difficulties, required features, and user expectations. Documentation and relevant literature were reviewed to support the analysis and system design.

2.3. System Design and Development

System design used Unified Modeling Language (UML), including use case diagrams to describe actors and their interactions, activity diagrams to represent the workflow from reporting to handling, and a class diagram to describe relationships among objects such as users, inventory, reports, and damage status. The application was implemented using PHP and Laravel 10.x, with MySQL as the relational database and phpMyAdmin for database management. Visual Studio Code was used as the code editor. The application structure followed the Model-View-Controller (MVC) pattern.

The supporting hardware listed in the thesis consisted of a laptop with 8 GB RAM and 256 GB storage, and an Android smartphone (Samsung A32) with 6 GB RAM and 128 GB storage. The software environment included Windows 10, Google Chrome, Microsoft Edge, Microsoft Office, Visual Studio Code, PHP using the Laravel framework, and MySQL. These tools supported development, database administration, and application access during the research.

2.4. Prototype Development Procedure

The Prototype model was selected to support iterative development and direct user involvement. The procedure began with initial requirements gathering through observation, interviews, and documentation. A quick design was then prepared to describe the interface, navigation, and main workflows, including damage reporting, report validation, and status monitoring. The initial prototype was built with multi-user login, a damage-report form with image upload, report-status management, and a history of asset complaints. Users then tried the prototype and

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

provided descriptive feedback through structured evaluative interviews. The prototype was refined according to this feedback and validated again before final system development.

The documented implementation shows an initial prototype and a subsequent iteration in which a facility-guidance feature was added. The first version presented the student dashboard before the additional guidance feature, while the second version included the facility-guidance menu. The iteration process also addressed interface and workflow feedback, including the need for photographic evidence, clearer report-status stages, and more accessible inventory information. This iterative procedure was intended to ensure that the final system reflected the operational needs identified by users.

2.5. System Testing and User Evaluation

Functional testing used Black Box Testing with the equivalence partitioning technique. The tests examined external system behavior without inspecting the internal code structure. Test scenarios covered the main functions for the administrator, student reporter, and Kasubag, including login, report submission, photo upload, report verification, status updates, report history, dashboard monitoring, and report export or printing. Each scenario compared the expected result with the observed result.

User acceptance was assessed using a questionnaire with a five-point Likert scale: strongly agree (5), agree (4), moderately agree (3), disagree (2), and strongly disagree (1). The questionnaire covered system functionality, system performance, interface experience, and efficiency and productivity. The respondents included the facilities administrator, students as reporters, and the Kasubag as the institutional supervisor and end user. The thesis reports ten responses in the calculation tables. The percentage for each item was obtained by dividing its mean score by the maximum Likert score and multiplying by 100%. The interpretation categories were 0–20% (very poor), 21–40% (poor), 41–60% (fair), 61–80% (good), and 81–100% (very good).

3. Results and Discussion

3.1. Analysis of the Existing Reporting Process

The initial analysis found that the reporting process was dispersed across verbal communication, personal chats, WhatsApp groups, and manual records. These channels did not provide a single structured repository for the report details and the subsequent handling process. The interviews also identified the need for a more orderly dashboard, clear report-status stages, and a mandatory photograph to support the assessment of damage. Users requested report information that could be checked without repeatedly opening separate detail pages, including the item category and room location.

The proposed system therefore centralized the reporting workflow. Students could submit reports and attach photographs, while the administrator could manage inventory and reports, verify the information, and update the handling status. The Kasubag could monitor reports, inspect their details, view report recapitulations, and print or export report data. The system also included facility guidance and review-related features, as described in the thesis implementation.

The interviews with the facilities-management representative clarified that the principal difficulty was not simply the use of informal communication, but the absence of systematic records. Reports sent through personal or group messages could be mixed with unrelated work conversations. This made it difficult to confirm whether a report had been received, to retrieve its details, and to monitor whether a repair had been completed. The initial requirements therefore emphasized a single reporting channel and a record that could be reviewed by the relevant institutional roles.

The feedback also shaped the information displayed in the application. The facilities administrator requested that the dashboard be arranged more compactly so that accumulated data would be easier to review. The administrator also requested clearer status terminology that followed the physical handling process. In addition, photographic evidence was requested as a required part of the report because it allowed the administrator to inspect the apparent condition of an item before going to the location. The inventory table was also expected to show the room location and item category directly.

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Tabel 1.Tabel User requirements and corresponding system responses.

Requirement identified during interviews	System response described in the thesis
Reports were scattered across verbal communication, private chats, and WhatsApp groups.	A dedicated web-based damage-report form and centralized report history.
Damage details were difficult to assess remotely.	Photographic evidence attached to the report and visible in report details.
The handling progress was not easy to follow.	Report-status management and monitoring by the administrator and Kasubag.
Item location and category required repeated navigation.	Room-location and item-category information in the inventory/report interface.
Users needed instructions on facility care and reporting.	Facility-guidance feature added in the subsequent prototype iteration.

3.2. System Features and Implementation

The application provides role-based access for three main actors. Students can register and log in, submit a damage report, upload photographic evidence, view their individual report history, and review facility-related information. The administrator manages user data and inventory-related information, monitors all reports, verifies report details, updates report status, manages facility guidance, and views review recapitulations. The Kasubag can view the dashboard and all reports, inspect report details and attached photographs, monitor the handling process, review completed reports, view recapitulations, and print or export reports.

The report form records the damaged item, its location, a description, the damage category, and photographic evidence. Report status is used to communicate the handling progress, with the workflow described in user feedback as progressing from a newly received report to a report under repair and finally to a completed repair. Centralized storage preserves the damage-report history and supports monitoring by the administrator and Kasubag.

Tabel 2. Main user roles and responsibilities

Actor	Main responsibilities in the system
Student / reporter	Register and log in; submit inventory damage reports; upload photographs; view individual report history; edit or cancel submitted reports as described in the system use cases; access facility guidance and review functions.
Facilities administrator	Manage user data; monitor and verify damage reports; inspect attached photographs; update report status; manage facility guidance; view review recapitulations and profile settings.
Head of Subdivision (Kasubag)	Monitor the dashboard and all reports; inspect report details; monitor handling progress; view completed-report history and recapitulations; print PDF reports and export spreadsheet data.



Figure 1. Landing page of the inventory damage reporting system.

Figure 1 presents the public landing page of the application. It introduces the facilities service and provides access to the damage-reporting and student-account functions. The landing page is the initial entry point before users access role-specific features.

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)



Figure 2. Administrator interface for monitoring inventory damage reports.



Figure 3. Report detail and verification interface with photographic evidence.

Figure 3 shows the report-detail interface used to examine the submitted damage information and the attached photograph. The administrator can inspect the item and reporter information, review the description, and update the report status. The integration of photographic evidence supports the verification process by allowing the damage to be viewed before follow-up.

3.3. Role-Based Interfaces and Workflow

The application separates the available functions according to the user role. The student-facing interface includes registration and login, a dashboard, a form for creating a damage report, individual report history, review history, facility guidance, and profile information. The report form is the primary entry point for submitting an incident. The information entered by the reporter is then available to the administrative role for checking and follow-up. The individual history page allows the reporter to revisit reports that have previously been submitted.

The administrator interface includes a dashboard, a page for monitoring all reports, report-detail and verification functions, user management, review recapitulation and graphs, facility-guidance management, and profile settings. The report list presents the reporter, item and location, and status information. The detail page provides access to the photograph and report description. The administrator can update the status after reviewing the report. The administrator also manages the guidance content that is displayed to students.

The Kasubag interface is oriented toward monitoring and recapitulation. It includes a dedicated login, dashboard, report list, report details, review graphs, facility guidance, and profile. The dashboard summarizes incoming reports and their handling status. The report page allows the Kasubag to inspect the reports submitted by students, while the output functions support the printing of PDF recapitulations and the export of spreadsheet data. This role separation reflects the operational relationship between the reporter, the report manager, and the supervisor.

The application therefore covers the reporting process from the perspective of each participant. The reporter provides the initial information and evidence; the administrator checks the report and manages its progress; and the Kasubag monitors the accumulated information. This arrangement is consistent with the problem analysis in which the person who first notices damage is not necessarily the staff member responsible for repair coordination or institutional monitoring.

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

3.3.1. UML System Modelling

The system requirements were translated into models using the Unified Modeling Language (UML). The use case diagram summarizes the interactions between the administrator, student reporter, and Kasubag. It distinguishes reporting activities from administrative verification and supervisory monitoring, and shows the role-specific access provided by the application.

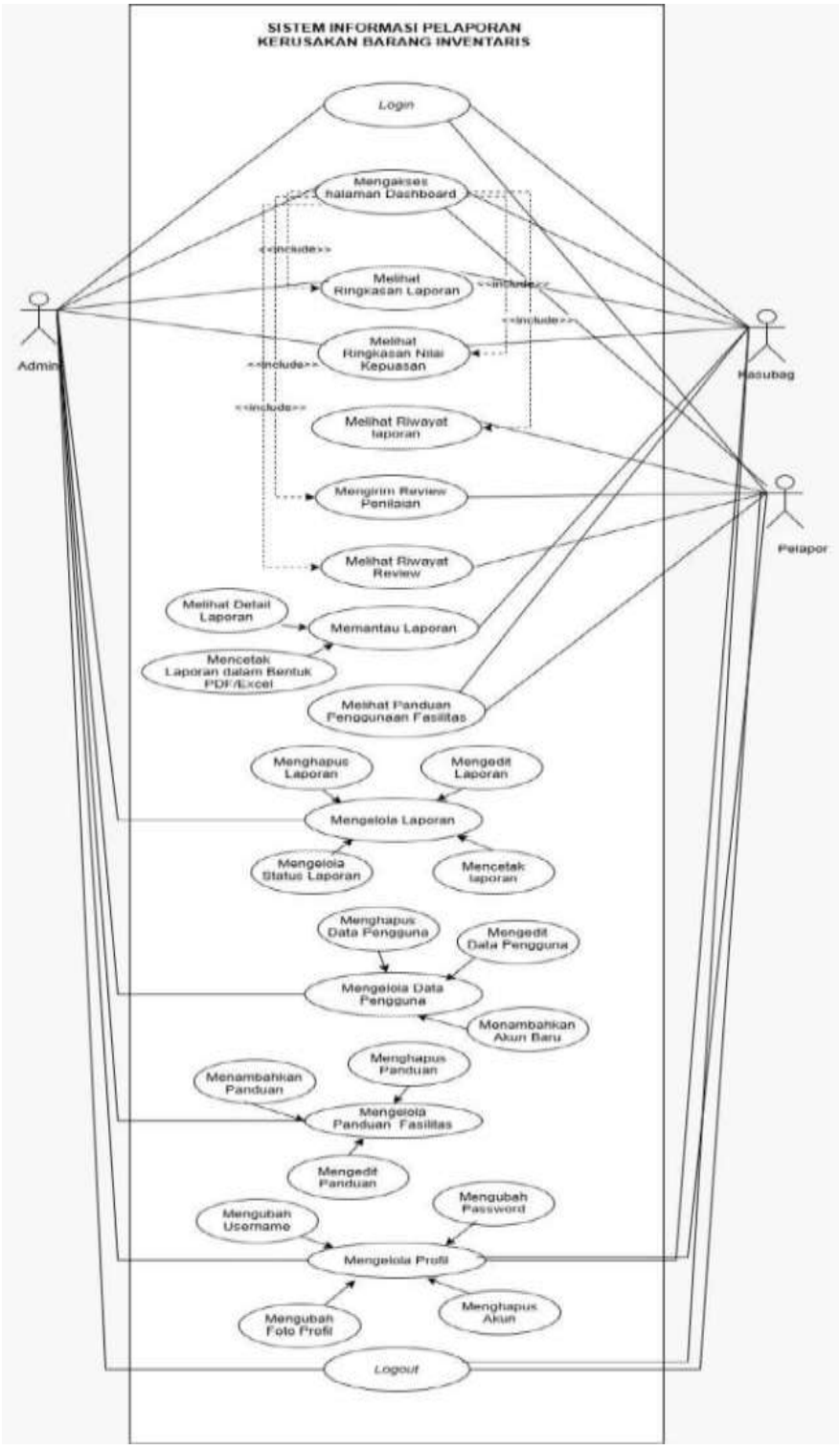


Figure 4. Use case diagram of the inventory-damage reporting system.

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

The class diagram represents the main data entities and their relationships. The thesis database design identifies users, reports, reviews, and guides as principal tables. User records support authentication and role assignment; report records store damage information and its handling status; review records support the review-related functions; and guide records store facility-care information. These entities provide the data structure required by the three-role workflow.

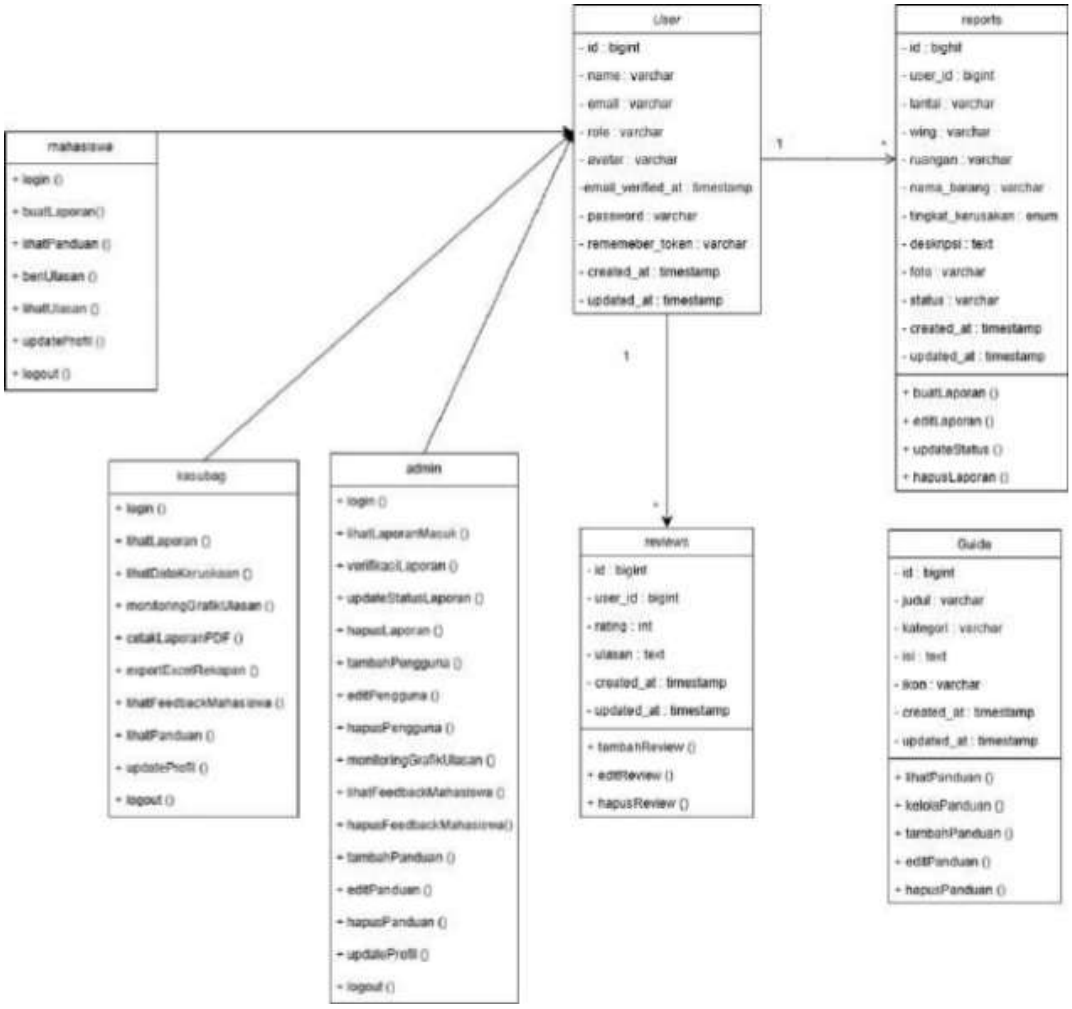


Figure 5. Class diagram showing the main system data entities.

3.4. Prototype Iterations and User Feedback

The first prototype provided the main student-facing reporting functions. During evaluation, users asked for a clearer dashboard and a more structured presentation of inventory information. Feedback also emphasized the importance of an attached photograph for identifying the extent of damage and the need for status labels that reflect the actual repair process. The administrator-side evaluation further requested room-location and item-category information in the inventory table so that the item position could be identified more directly.

The next documented iteration included a facility-guidance feature. This addition provided users with guidance on facility use and reporting procedures. The user-facing guidance page shown in Figure 4 is part of the implemented system. The iterative process demonstrates how user comments were translated into interface and feature adjustments rather than treating the initial design as final.

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)



Figure 6. Student interface displaying facility-care guidance.

The facility-guidance page provides information for students about facility care and the reporting flow. In the thesis, this feature is documented as an addition in the second prototype iteration. The page complements the damage-reporting function by presenting guidance within the same application.

3.5. Black Box Testing Results

Black Box Testing was conducted for the administrator, student, and Kasubag roles. The administrator scenarios included valid and invalid login, dashboard access, report management, report-detail review, user management, review recapitulation, facility-guidance management, profile settings, and logout. Student scenarios included registration and login, dashboard access, report submission, photographic evidence, individual report history, review functions, facility guidance, and logout. Kasubag scenarios included login, dashboard monitoring, report review, report recapitulation in PDF, spreadsheet export, and logout[14][15].

The test tables in the thesis record the expected and observed outcomes for the tested scenarios as successful. The tested functions included validation of login credentials, role-based redirection, report data storage and display, status changes, report history, image attachment, report printing, and spreadsheet export. The thesis concludes that all tested main features operated according to their expected operational functions, with no reported functional failure in the test scenarios.

Tabel 3. Representative Black Box Testing areas and expected behavior.

Test area	Example of expected system behavior documented in the thesis
Authentication	Valid credentials open the dashboard assigned to the user role; invalid credentials are rejected with an error message.
Damage reporting	A submitted report is stored and displayed; incomplete input is validated; a photograph can be attached as evidence.
Report verification	The administrator can open report details, inspect the attached photograph, and save a status update.
Report history	Previously submitted reports can be displayed in the reporter history and the relevant administrative report list.
Monitoring and outputs	The Kasubag dashboard displays report counts; date-filtered PDF reports and Excel spreadsheet exports can be generated.

Tabel 4. Summary of Black Box Testing results based on the thesis test scenarios.

User role	Main functions tested	Reported test outcome
Administrator	Login, dashboard, report management and verification, user management, review recapitulation, facility guidance, profile, logout	All recorded scenarios: successful
Student reporter	Registration, login, damage-report submission, photo upload, report history, reviews, guidance, logout	All recorded scenarios: successful
Kasubag	Login, dashboard monitoring, report details, PDF report printing, Excel export, logout	All recorded scenarios: successful

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

3.6. User Acceptance Testing Results

User acceptance was measured across four evaluation variables. The detailed calculation tables in the thesis report an average of 89.6% for system functionality, 92% for system performance, 89.3% for interface experience, and 98% for efficiency and productivity. Each of these detailed variable scores falls within the 81–100% "very good" interpretation category defined in the thesis.

Tabel 5. User Acceptance Testing results from the detailed variable calculations.

Evaluation variable	Reported mean percentage	Interpretation
System functionality	89.6%	Very good
System performance	92%	Very good
Interface experience	89.3%	Very good
Efficiency and productivity	98%	Very good

The thesis states an overall UAT result of 92% and concludes that the system is very feasible for operational use. The positive results indicate that users assessed the main functions, system responsiveness, interface, and efficiency favorably. In particular, the questionnaire included statements on report submission, photographic uploads, dashboard accuracy, report-status processing, guidance access, report export, response speed, data storage, role-based access, interface clarity, and the reduction of time spent reporting damage.

3.7. User Acceptance Indicators

The questionnaire was organized around four dimensions. The functionality dimension examined whether users could submit damage reports, upload photographs, view accurate report data, follow status changes, access the guidance menu, manage facility guidance, use input validation, and obtain report outputs in Excel or PDF. The performance dimension examined availability when needed, response speed when saving a report, safe storage of photographs and report details, and the operation of role-based access. The interface dimension examined system stability, readability of status information, color composition, ease of understanding for new users, placement of action buttons, and dashboard layout. The efficiency and productivity dimension examined reporting time, status tracking, administrative workload in preparing monthly reports, review transparency, the risk of reports being overlooked, and the speed of decisions about facilities maintenance.

These indicators connect the acceptance assessment to the practical tasks performed by the three user groups. For reporters, the relevant experience concerned entering a report, attaching evidence, and following its history. For the administrator, the indicators concerned managing reports, status, guidance, and recapitulation. For the Kasubag, the focus included monitoring, report outputs, and the availability of information for evaluation. The five-point response scale provided a common basis for summarizing the responses across these different tasks.

Tabel 6. Summary of User Acceptance Testing indicators.

Dimension	Questionnaire focus recorded in the thesis
System functionality	Report entry, photo upload, dashboard data accuracy, status changes, guidance access and management, input validation, PDF/Excel output.
System performance	Availability, response speed, data storage, and separation of access rights for the three roles.
Interface experience	Stability, readability, visual composition, ease of use, action-button placement, and dashboard clarity.
Efficiency and productivity	Reporting time, status tracking, administrative workload, review transparency, prevention of overlooked reports, and decision-making.

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

3.8. Discussion

The developed system addresses the principal weakness identified during the initial analysis: damage reports were distributed across communication channels and were not maintained in a single structured record. By providing a dedicated report form and centralized history, the application creates a consistent record containing item information, location, description, damage category, and photographic evidence. This directly supports the reporting and documentation objectives of the research.

The three-role arrangement also reflects the reporting and follow-up responsibilities identified in the faculty environment. Students act as reporters, the administrator manages and verifies reports, and the Kasubag monitors the progress and recapitulation. The role-based dashboard and status updates make the report process more visible to the relevant users. The PDF and Excel output functions further support the preparation of report recapitulations, as confirmed by the Kasubag test scenarios.

The findings are consistent with the broader direction of previous web-based inventory studies, which have used digital systems to improve inventory recording and monitoring. The present study narrows the focus to damage reporting and its follow-up rather than general inventory administration alone. Its use of iterative prototyping is also in line with studies that have applied prototype development to inventory

The reported Black Box Testing outcomes support the conclusion that the tested functions operated as expected. The UAT findings also indicate favorable user assessments. However, the evaluation figures need to be read with attention to the calculation tables: the detailed variable percentages are 89.6%, 92%, 89.3%, and 98%, while the thesis separately reports a final overall value of 92%. The rounded mean of the four detailed variable percentages is approximately 92.2%, which is consistent with the reported overall value when rounded. The summary table numbered 4.30 in the thesis lists a different set of percentages (89%, 86%, 89%, and 91%); this discrepancy should be checked against the original questionnaire calculations before journal submission.

The application also incorporates functions beyond the submission form itself. The administrator can manage user accounts and facility-guidance content, while the Kasubag has access to a separate monitoring interface and report outputs. Students can view their report history and the facility-guidance content. These functions are documented in the system implementation chapter and indicate that the application was designed as a shared workflow rather than a single-purpose submission page.

The visual implementation supports the system workflow at several stages. The landing page introduces the service, the administrator interface provides an overview of reports, and the report-detail screen presents the photograph and report information for verification. The student guidance page provides information on caring for facilities and the sequence of handling a report. Together, these screens illustrate the connection between report submission, administrative verification, and user guidance.

4. Conclusion

This study successfully developed a web-based Inventory Damage Reporting Information System for the Faculty of Science and Technology, UIN Sulthan Thaha Saifuddin Jambi, using the Prototype model. The application provides structured damage reporting with item details, location, description, damage category, and photographic evidence. It also supports report verification, status updates, inventory-related management, report monitoring, facility guidance, and centralized storage of damage history. The system assigns functions to students as reporters, the facilities administrator as report manager, and the Kasubag as supervisor. The Black Box Testing scenarios recorded successful outcomes for the tested functions across the three user roles. The thesis reports an overall User Acceptance Testing score of 92%, interpreted as very good and very feasible. These results indicate that the developed system can support a more structured and documented damage-reporting process at the faculty. The thesis recommends future development toward mobile access, real-time notification integration, and further improvements to database scalability and digital-file security.

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Reference

1. Aramintha, F. C., Nurlaili, A. L., & Aditiawan, F. P. (2025). Design and implementation of digital medical record system using the Boyer–Moore algorithm. 8(2). https://doi.org/10.32877/bt.v8i2.2959
2. Aryani, R., Bintana, R. R., Arsa, D., & Abidin, Z. (2023). Prototype Sistem Inventarisasi Barang Milik Negara (SIBAMIRA) pada Universitas Jambi. 17(1), 61–72.
3. Assiddiq, M., Fathahillah, & Yasdin. (2025). Manajemen implementasi sistem informasi inventaris barang berbasis web. Variable Research Journal, 02(02).
4. Christian, A., & Suhartini, S. (2020). Implementasi model prototipe pada perancangan sistem informasi inventaris (studi kasus SMP Yayasan Bakti Prabumulih). Jurnal Sistem Informasi dan Sains Teknologi, 1(2), 1–10. https://doi.org/10.31326/sistek.v1i2.674
5. Cipto Gumono, E., Falgenti, K., & Baidawi, T. (2025). Rancang bangun sistem informasi inventaris di sekolah menengah kejuruan menggunakan metode prototype. 1(04), 373–379.
6. Elsen, R., & Muhamad, Y. (2025). Development of web-based hijab product e-catalog application integrated with WhatsApp chatbot. 32–44. https://doi.org/10.33364/sistematik/v.1-2.2536
7. Febriyanti, N. M. D., Oka, A. A. K., & Piarsa, I. N. (2021). Implementasi Black Box Testing pada sistem informasi manajemen dosen. 2(3).
8. Jannah, M., Suhartono, S., & Sidin, U. S. (2021). Sistem informasi aset (SIMaset) barang dan dokumen berbasis web. Teknosains: Media Informasi Sains dan Teknologi, 15(3), 367–380. https://doi.org/10.24252/teknosains.v15i3.23363
9. M. Zayyan Musoffa, Eri Sasmita Susanto, & Yudi Mulyanto. (2022). Sistem informasi manajemen aset berbasis web di Universitas Teknologi Sumbawa. Jurnal Informatika Teknologi dan Sains, 4(1), 42–51.
10. Marcellino. (2024). Analisa dan perancangan sistem informasi berbasis web pada inventaris barang menggunakan framework React.JS.
11. Nurkasih, P., & Suparman, P. (2022). Implementasi metode prototype pada sistem informasi persediaan barang berbasis website. Jurnal Sosial Teknologi, 2(7), 617–629. https://doi.org/10.36418/jurnalsostech.v2i7.375
12. Prayoga, M. R., Divanagara, M. A., & Nursetyo, A. (2025). Perancangan dan implementasi sistem informasi ujian berbasis website pada cabang olahraga Taekwondo (studi kasus: Pencab TI Kota Cirebon). 1(1), 39–48.
13. Putri, A., Ul Haq Azzahra, N., Adhi Samsara, T., & Hidayatullah Al Islami, dan. (2025). Implementasi sistem inventaris berbasis web untuk pengelolaan barang di TVRI Stasiun DKI Jakarta dengan metode prototype. Terapan Ilmu Multidisiplin, 1(3), 150–156. https://jurnal.portalpublikasi.id/index.php/JPTIM
14. Sibuea, F. P. J., Agustin, D., Ferdhinand, A., Widyatmoko, W., Nomensen, D., & Kusmawati, A. (2024). Rancang bangun sistem inventory barang berbasis web dengan metode prototyping di Program Studi Teknologi Rekayasa Otomotif Politeknik STMI Jakarta. ILKOMNIKA: Journal of Computer Science and Applied Informatics, 6(1), 91–101.
15. Wahyuni, N., Akmal, R., & Gunawan, A. (2020). Perancangan sistem informasi basis data inventaris barang berbasis web menggunakan model Waterfall. Jurnal Manajemen Industri dan Logistik, 4(2), 102–115. https://doi.org/10.30988/jmil.v4i2.434

DOI: https://doi.org/10.69693/ijmst.v5i1.14598
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)