International Journal of Computing Sciences and Technology Studies
https://journals.hu.edu.so/index.php/ijcsts

A Smart Warehouse Management System for Inventory Control and Operational Efficiency: Design, Implementation and Functional Evaluation

Mustaf Abdi Ibrahim, Bashir Hassan Mohamed, Abduqadir Abdixamid Abukar, Ahmed Hussein Elmi*
Department of Computer Science, Faculty of Computer Science and Information Technology, Hormuud University, Mogadishu, Somalia

*Corresponding author's email: ahmedhussein56@gmail.com

Abstract
Warehouse operations that depend on paper records, spreadsheets, and delayed reconciliation are subject to inaccurate inventory records, limited visibility of stock movements, slow reporting, and weak accountability. These weaknesses motivate the adoption of integrated web-based warehouse management systems, particularly in settings where capital for barcode or RFID infrastructure is unavailable. This study designs, implements, and functionally evaluates a Smart Warehouse Management System (SWMS) that centralizes inventory and warehouse operations on commodity hardware, and provides assisted information retrieval through a natural-language query interface. A descriptive, multi-method study was used to characterize existing warehouse practice and derive system requirements. Direct observation, a thirteen-item structured interview schedule, and review of warehouse documentation informed requirements analysis. The system was implemented as a three-layer web application over a document-oriented data model of twenty-four collections and evaluated through functional verification and qualitative usability, performance, and security reviews. The implemented system supports role-based access control, product and category management, vendor and customer records, zone, rack, and bin location management, stock receiving and dispatch, adjustment and audit workflows with variance detection, threshold-based low-stock alerting, on-demand reporting, and natural-language inventory query in English and Somali. Five of seven specified functional test cases were executed and passed; the two covering audit variance computation and permission refusal were not executed and are reported as gaps. Qualitative usability, performance, and security reviews indicated acceptable operation under the tested conditions, but were not instrumented to yield quantitative measures. The study demonstrates the feasibility of integrating conventional warehouse functions with web-based access and assisted retrieval on a platform deployed and evaluated without any barcode, RFID, or scanning equipment. The evaluation establishes functional correctness rather than operational improvement; quantified field evaluation against the incumbent process remains necessary and is specified as the next step.

Keywords: Warehouse management system, Inventory control, Warehouse digitalization, Role-based access control, Stock audit, Bilingual query interface, Low-resource deployment, Somalia

1. Introduction

Warehouse management is an operational function comprising the receipt, storage, tracking, movement, and dispatch of goods. Effective management of these activities requires accurate and timely information about products, quantities, storage positions, suppliers, customers, and stock movements [1], [2]. In many warehouses these activities are still managed with paper records and basic spreadsheets, which becomes progressively harder to control as product volume and transaction complexity increase.

Semi-manual processes create a structural gap between physical and recorded stock. Transactions are captured on paper at the point of handling and transcribed later, so the record lags reality irrespective of transcription accuracy. Field observation conducted for this study identified delayed inventory updates, difficulty tracking stock movements, dependence on paper documents and spreadsheets, and limited accountability for individual actions as the principal operational weaknesses.

Warehouse Management Systems address these limitations by centralizing inventory records, mediating stock movement, and generating operational reports [3], [4]. The literature describes a progression from manual systems through computerized, web-based, and cloud-supported solutions to intelligent warehouse platforms [5], [6]. Much of this work, however, presupposes barcode or RFID data capture, and therefore capital expenditure on tags and readers that small and medium warehouses in low-income settings cannot finance [7], [8], [9]. The design target adopted here is accordingly a system that recovers as much of the benefit as possible in a configuration deployed without auxiliary capture hardware, relying on manually maintained bin-level position data instead.

This study develops a Smart Warehouse Management System combining core warehouse functions with assisted information retrieval. The intended contribution is not a claim of warehouse autonomy but a practical, web-accessible platform in which inventory information, operational records, reconciliation, reporting, and assisted lookup are integrated on commodity hardware. The work addresses four questions: how current warehouse practices operate; what requirements a web-based system should satisfy; how such a system can be designed; and how it can be implemented and verified.

2. Related Work

2.1 Evolution of warehouse management systems

The literature shows a progression from manual record keeping to digital and web-based warehouse management [5], [6]. Manual systems remain workable for very small operations but become difficult to control as transaction volume rises. Digital systems automate record keeping and inventory updates; web-based systems add centralized access from multiple locations and consistent data across users [4], [10]. Reported benefits are convergent in direction but vary in magnitude with warehouse scale, inventory heterogeneity, and how formalized the incumbent process was before implementation [3], [6], which suggests outcomes are contingent on the baseline rather than intrinsic to the technology.

2.2 Automated identification and the capital assumption

A substantial strand of work treats barcode, RFID, and IoT sensing as the enabling layer for real-time inventory visibility, and reports accuracy gains attributable to it [7], [8]. Each approach presupposes expenditure on tags, readers, and supporting infrastructure. This literature therefore establishes an upper bound on achievable accuracy rather than a feasible design target for warehouses that cannot finance the capture layer, and it leaves open how much of the gain is recoverable through process and interface design alone.

2.3 Assisted retrieval and predictive functions

Recent work extends warehouse management with predictive analytics and conversational interfaces [4], [5]. Conversational interfaces are reported to shorten the path between a user question and an operational answer, which is relevant where staff have limited familiarity with data-model navigation. Predictive replenishment methods forecast demand over the replenishment lead time and derive safety stock from demand and lead-time variability [11], [12]; simpler reorder-point policies, in which a per-item threshold triggers a purchase signal, remain widespread in small-firm settings because they require no demand history.

2.4 Research and system gap

The reviewed literature indicates limited attention to integrated warehouse platforms designed specifically for low-resource environments where organizations face cost, skills, and connectivity constraints [9], [13]. The system reported here addresses that gap by combining inventory control, stock movement, reconciliation with variance detection, bin-level location tracking, reporting, role-based access, and assisted retrieval within one web-based platform, in a configuration that uses no scanning hardware. Table 1 positions the system against the comparator classes on the capability dimensions addressed in this study.

Table 1. Comparison of warehouse management approaches against the capability dimensions addressed in this study.

Capability	Manual / paper	Digital WMS	Web-based WMS	Capture-based WMS	This study
Centralised inventory record	No	Yes	Yes	Yes	Yes
Multi-location access	No	Limited	Yes	Yes	Yes
Bin-level position tracking	Informal	Varies	Varies	Yes	Yes
Automated data capture	No	No	No	Yes	Not used in evaluation
Audit with variance detection	Manual	Varies	Varies	Yes	Yes
On-demand reporting	No	Yes	Yes	Yes	Yes
Attributed action log	No	Varies	Varies	Yes	Yes
Assisted natural-language retrieval	No	No	No	Varies	Yes
Capital requirement	None	Low	Low–moderate	High	None in evaluated config.

Note. Entries for comparator classes summarize capabilities reported in the reviewed literature, not a single named product. "Varies" indicates the capability is present in some reported systems and absent in others.

3. Materials and Methods

3.1 Research design

The study used a descriptive, multi-method design combining direct observation, structured interviews, and document review. Its purpose was to characterise existing warehouse processes, identify operational weaknesses, derive system requirements, and use those requirements to guide design, development, and verification of the SWMS. Fig. 1 summarises the stages of that process and the data sources feeding each.

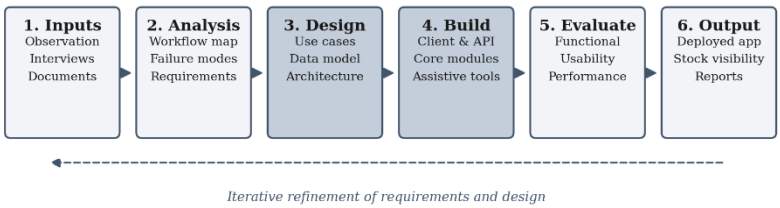


Fig. 1. Operational framework linking data collection, analysis, design, construction, evaluation, and outputs. Shaded stages denote artifact construction.

3.2 Data collection

Three methods were used. Direct observation documented how inventory was received, stored, recorded, issued, and dispatched, and identified manual record keeping, delayed updates, difficulty tracking stock movement, and limited real-time visibility. Structured interviews using a thirteen-item schedule were conducted with participants drawn from warehouse managers, storekeepers, and inventory-control personnel. Document review examined inventory records, stock-movement forms, supplier invoices, and warehouse reports to corroborate the interview accounts.

3.3 Requirements analysis

Requirements were derived from the observed workflow and reported user needs, and are grouped in Table 2 by concern. Each group is traceable to an operational weakness identified during observation and to a module in the implemented system.

Table 2. Functional and non-functional requirements derived from field analysis.

ID	Requirement group	Principal functions	Type
R1	Access and accountability	Authentication, role assignment, permission scoping, activity logging	Functional / NF
R2	Product master data	Register, update, categorise, search and filter products; unit and expiry handling	Functional
R3	Inventory control	Stock-in, stock-out, stock-level monitoring, real-time quantity updates	Functional
R4	Location management	Warehouse, zone, rack and bin definition; assignment of stock to bin positions	Functional
R5	Reconciliation	Adjustments with typed reason; physical count sessions with variance detection	Functional
R6	Business records	Vendor and customer records, purchases, sales orders, balances	Functional

ID	Requirement group	Principal functions	Type
R7	Reporting	Inventory, stock movement and operational reports with export and filtering	Functional
R8	Assisted retrieval	Threshold-based low-stock alerting; natural-language query in English and Somali	Functional
R9	Operability constraint	Runs on commodity workstation; no barcode, RFID or scanning hardware required	Non-functional
R10	Multi-branch separation	Warehouse-scoped data access across users, stock, transactions and reports	Non-functional

3.4 Development and evaluation process

No formal software development life-cycle model was prescribed at the outset. The work nevertheless followed a staged sequence of requirements analysis, system design, implementation, testing, and deployment, with iteration between analysis and design as requirements were refined.

4. System Design and Implementation

4.1 Architectural design

The system follows a three-layer architecture. The presentation layer is a component-based browser client providing dashboards, data-entry forms, inventory views, and reports. The application layer exposes a REST interface and handles authentication, authorization, request validation, and transaction and stock logic. The persistence layer is a document-oriented store accessed through an object-document mapper that enforces schema structure at the application boundary. A warehouse identifier propagates through users, stock records, transactions, and reports, so that one deployment serves multiple branches with separated data. Fig. 2 shows the three layers and the request path between them.

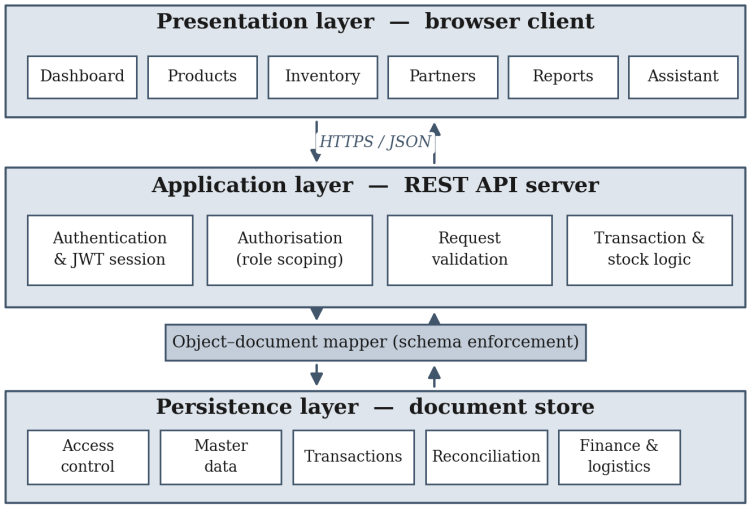


Fig. 2. Three-layer architecture of the Smart Warehouse Management System. The client runs on a commodity workstation, and the evaluated configuration used no scanning hardware; the warehouse identifier scopes all records for multi-branch operation.

4.2 User roles and access control

The system implements role-based access. Administrators manage users, roles, product and category master data, reports, and activity logs. Warehouse managers supervise stock movement, approve adjustments, run audits, and monitor low-stock alerts. Warehouse staff register products, maintain bin assignments and partner records, record receiving and dispatch transactions, and use the query interface for retrieval. Navigation is role-aware, restricting module visibility to the permission set attached to the role. Because SME staffing frequently assigns several functions to one person, permission sets are configurable and may overlap. Fig. 3 shows the three actors and the use cases available to each.

Fig. 3. Use-case diagram showing the three actors (administrator, warehouse manager, warehouse staff) and the use cases available to each.

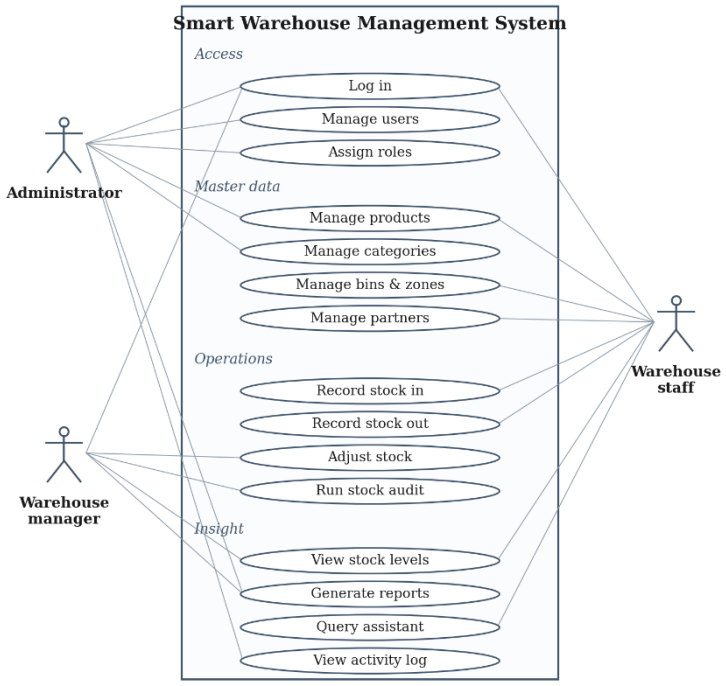


Fig. 3. Use-case model showing the three actors and the use cases available to each, grouped by concern.

4.3 Data model

A document-oriented store was selected because warehouse master data varies in structure across product classes and the schema was expected to evolve during development. The model comprises twenty-four collections spanning access control, warehouse topology, master data, trading partners, transactions, reconciliation, finance, personnel, and logistics. Fig. 4 groups these entities by concern, and Table 3 lists the principal collections with their key attributes. The full logical model is given in Appendix A.

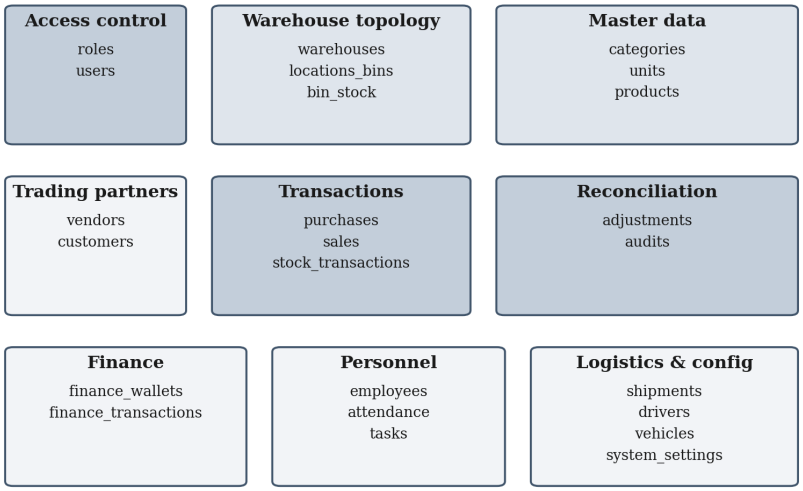


Fig. 4. Data model organized by concern. Shaded groups carry the operational core: stock transactions and the reconciliation workflow.

Table 3. Principal entity groups, their collections, and key attributes.

Group	Collections	Key attributes
Access control	roles, users	role name, permission set, access level (system / branch / warehouse), assigned warehouse, account status
Warehouse topology	warehouses, locations_bins, bin_stock	warehouse code and manager, zone, rack, bin code, bin capacity and status, quantity held per bin
Master data	categories, units, products	SKU, category, unit, purchase cost, selling price, reorder level, expiry-tracking flag
Trading partners	vendors, customers	name, contact details, payable or receivable balance, status
Transactions	purchases, sales, stock_transactions	order number, line items, total amount, payment status, order status, transaction type and date
Reconciliation	adjustments, audits	adjustment quantity, bin reference, typed reason, acting user; audit code, system quantity, physical quantity, computed variance
Finance	finance_wallets, finance_transactions	wallet type (cash, bank, mobile money), balance, currency, transaction type and amount
Personnel and logistics	employees, attendance, tasks, shipments, drivers, vehicles	staff and assignment records; shipment destination, driver, vehicle and status
Configuration	system_settings	company name, currency, timezone, low-stock threshold

Note. The audits collection stores system quantity, physical quantity, and computed variance per counted product, so inventory variance can be calculated from recorded system and physical-count quantities.

4.4 User interface

The client was implemented with a component-based JavaScript framework, a responsive component library, a client-side router, a context provider for authentication state, framework hooks for local view state, and a promise-based HTTP client for API communication. The dashboard (Fig. 5) presents an operational summary comprising total products, available stock, low-stock alerts, recent stock movements, and inventory statistics.

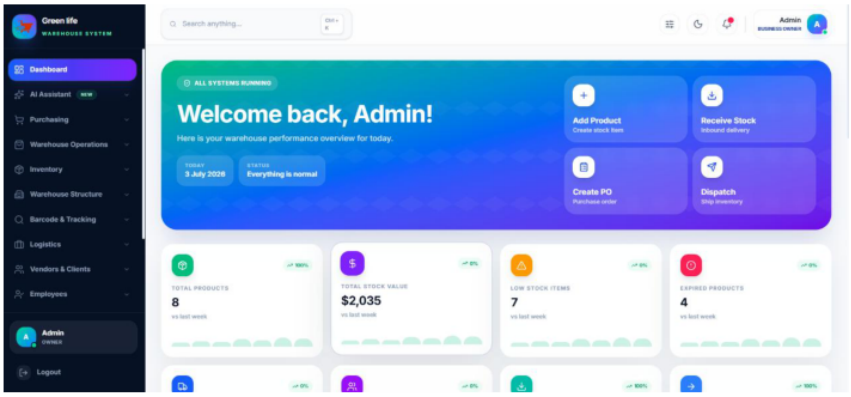


Fig. 5. Dashboard presenting the operational summary: product count, stock value, low-stock items, and expired-product count, with role-aware module navigation.

4.5 Assisted retrieval functions

The query interface provides a natural-language layer over selected inventory information, answering questions about low-stock items, recent inventory updates, and recorded quantities for a given product, with responses derived from current system state. The interface accepts queries in both English and Somali, which matters in a setting where warehouse staff may be more fluent operating in Somali than in the language of the underlying data model; bilingual retrieval is a direct expression of the accessibility requirement R8. Related views group stock analysis, expiry-risk analysis, and reorder suggestions alongside the conversational entry point (Fig. 6).

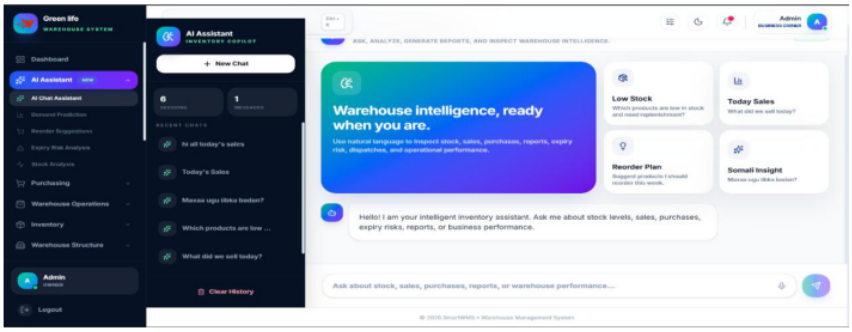


Fig. 6. Natural-language query interface, showing English and Somali query examples and the grouped analysis views. Content shown is demonstration data. Replenishment support is provided by a per-product reorder level: a configured minimum quantity below which the item is surfaced in the low-stock view. The threshold is set by an administrator and is static; it is not derived from demand history and does not incorporate supplier or import lead time. This is a deliberate simplification, since the incumbent process generates no reliable transaction history to learn from, and an alert that staff can interpret and act on has practical value independent of forecasting sophistication.

4.6 Security

Five controls protect warehouse and financial data. Credentials are stored as bcrypt hashes rather than in recoverable form. Upon successful authentication, the server issues a JSON Web Token (JWT), which is validated on subsequent protected requests. Role-based access control restricts each user to the permission set attached to their role and, where configured, to an assigned warehouse. API endpoints validate request payloads before processing. An activity log records state-changing operations against the acting user, supplying attribution that the incumbent paper process could not provide.

4.7 Technology stack

Table 4 lists the components used at each layer, together with the supporting development, testing, and version-control tools.

Table 4. Implementation stack and supporting tools.

Layer / concern	Component	Role in the system
Presentation	React.js	Component-based user interface
Presentation	Bootstrap, HTML5, CSS3	Responsive layout and styling
Presentation	React Router	Client-side route handling
Presentation	Context API, React Hooks	Authentication and local view state
Presentation	Axios	HTTP communication with the API
Application	Node.js	Server-side runtime
Application	Express.js	REST API framework
Application	Jsonwebtoken	Token-based session authentication
Application	Bcrypt	Password hashing
Persistence	MongoDB	Document-oriented data store
Persistence	Mongoose	Object-document mapping and schema enforcement
Environment	VS Code, Postman, Git, MongoDB Compass	Development, API testing, version control, database administration

5. Verification and Evaluation

5.1 Functional verification

Functional verification was performed against the principal system modules. Each test case exercised a defined precondition and expected system state, and was recorded as passed where the observed behavior matched the expectation. Table 5 reports the cases and their outcomes; TC-06 (audit variance) and TC-07 (permission refusal) were specified but not executed.

Table 5. Functional verification cases and outcomes.

ID	Module	Precondition	Expected behavior	Observed	Status
TC-01	Authentication	Registered account, valid credentials	Credentials verified against stored hash; session token issued; protected routes accessible	As expected	Pass
TC-02	Product management	Authenticated user with create permission	Product persisted with SKU, category, unit, cost, price and reorder level	As expected	Pass
TC-03	Inventory control	Product exists with known quantity	Recorded receipt or dispatch updates available quantity without manual recalculation	As expected	Pass
TC-04	Reporting	Transaction history present	Report generated on demand from transaction record and exported	As expected	Pass
TC-05	Query interface	Inventory data present	Query in English or Somali returns quantity consistent with stored state	As expected	Pass
TC-06	Stock audit	Counted quantity entered against system quantity	Variance computed and stored; adjustment attributed to acting user	Not executed	Pending
TC-07	Access control	User without the required permission	Action refused and refusal recorded in activity log	Not executed	Pending

5.2 Usability, performance, and security review

Usability review considered ease of navigation, interface clarity, speed of operation, and accessibility of features, with participants reporting that the interface was clear and straightforward to use. The review was not administered with a standardized instrument; the System Usability Scale [14], [15] and established technology-acceptance constructs [16] are identified in Table 6 as the measures that would convert these observations into comparable, reportable results. Performance review considered response time, data-retrieval speed, concurrent-user handling, and inventory-processing speed, with operation reported as efficient under normal warehouse conditions. Security review examined password hashing, authentication, prevention of unauthorized access, session handling, and input validation, with unauthorized access prevented in the cases tested.

Table 6. Evaluation status by dimension, and the measurement required to strengthen each.

Dimension	Current basis	Yields	Measurement required
Functional correctness	Documented test cases	Largely established (5 of 7)	Execute TC-06 (audit variance) and TC-07 (permission refusal)
Usability	Informal user feedback	Not quantified	System Usability Scale, $n \geq 20$; report mean, SD and percentile band

Dimension	Current basis	Yields	Measurement required
Acceptance	Not assessed	Absent	Perceived usefulness, ease of use and behavioral intention scales; report reliability per construct
Performance	Qualitative observation	Not quantified	Load test at 10 / 50 / 100 concurrent users; report p50, p95, p99 latency, throughput, error rate
Security	Manual checks	Partially established	Document the test cases attempted and their outcomes; add dependency vulnerability scan
Operational improvement	Not assessed	Absent	Pre/post field measurement of inventory accuracy, report preparation time, receiving cycle time and dispatch error rate
Retrieval accuracy	Not assessed	Absent	Query test set of ≥ 100 items; report intent accuracy, answer correctness, latency, fallback rate

5.3 Deployment

The system was deployed to a hosting environment supporting the server runtime and database. Deployment involved uploading application files, configuring the database connection and environment variables, installing dependencies, and starting the application server. The platform is reachable through a browser wherever internet connectivity is available; the system has no offline mode, and under unstable connectivity users experience delays in access and in recording updates. Fig. 7 closes this section by tracing each observed operational weakness through the requirement it generated and the module that implements it to the verification case that exercises it.

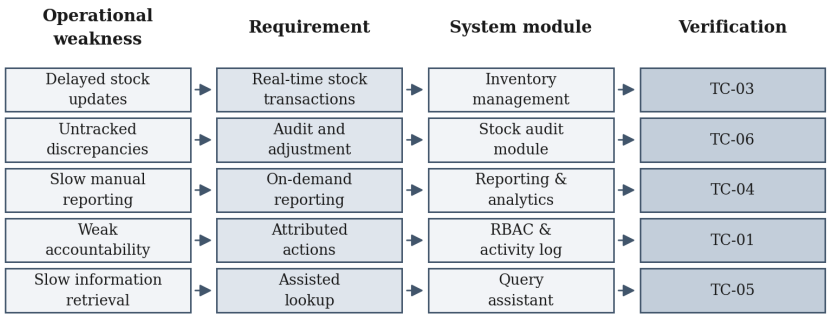


Fig. 7. Traceability from observed operational weakness through requirement and implementing module to the verification case that exercises it.

6. Discussion

The results indicate that consolidating core warehouse functions into a single web platform addresses several of the operational weaknesses identified during requirements analysis [3], [10]. Dispersed paper and spreadsheet records are replaced by a centralized store in which inventory quantities update as a consequence of recorded transactions rather than by later transcription. This reduces the lag associated with later transcription by updating inventory

records when transactions are entered into the system, although a residual interval remains between physical handling and record entry.

Two design choices merit comment because they depart from the assumptions of mainstream warehouse systems. First, the configuration deployed and evaluated in this study used no barcode, RFID, or scanning hardware, recovering locational benefit instead through bin-level position recorded as manually maintained master data [2], [17]. The interface reserves a barcode and tracking module for future capture-device integration, but no such device was used in the work reported here. The trade-off is explicit: position is only as accurate as the last recorded placement, which makes the audit workflow an important mechanism for maintaining location accuracy. Second, the audit collection stores counted quantity alongside system quantity and computes the variance, so reconciliation produces evidence about where and why accuracy is lost rather than only a corrected number. Reported deployment-related expenditure was USD 425, covering a workstation, connectivity, and printing, with no expenditure on capture hardware. This figure describes the evaluated deployment configuration; it should not be interpreted as total software development cost.

The evaluation must nevertheless be read with care. Functional verification demonstrates that selected modules operate as specified. It does not establish performance benchmarks, retrieval accuracy, or effectiveness across a population of warehouse users, and it provides no comparison against the incumbent process. Access to real warehouse environments was limited, testing scenarios and users were few, the system depends on stable connectivity, and the assistive functions are basic implementations.

Accordingly, the strongest claim the study supports is an implemented integrated platform and an implementation approach suitable for empirical evaluation, not a demonstrated operational improvement. Future work should measure the system in operational warehouses against a documented pre-deployment baseline, administer standardised usability and acceptance instruments, evaluate retrieval accuracy on a defined query set, and test performance under realistic concurrent load.

7. Limitations and Future Work

Table 7 sets out the limitations that bound the findings reported above, the consequence of each, and the work that would address it.

Table 7. Limitations and the corresponding future work.

Limitation	Consequence	Future work
No quantified field evaluation	Operational improvement is not demonstrated, only functional correctness	Pre/post measurement at multiple sites using the measures in Table 6
Requires stable connectivity; no offline mode	Access and recording degrade under intermittent connectivity	Local write buffering with deterministic reconciliation on restoration
Recording occurs at a workstation, not at point of handling	A residual interval remains between movement and record	Mobile client for warehouse-floor capture
Replenishment uses a static per-product threshold	Alerts do not adapt to demand variability or supplier lead time	Lead-time-aware reorder point evaluated against established forecasting baselines

Limitation	Consequence	Future work
Assistive retrieval is unspecified and unevaluated	Capability boundary and failure modes are unknown	Specify the mechanism; evaluate on a query set with a failure taxonomy
Limited test environments and users	Findings may not generalise across warehouse scale or sector	Multi-site study with documented site profiles

8. Conclusion

This study presented the design, implementation, and functional verification of a Smart Warehouse Management System for inventory control. Requirements were derived from observation, structured interviews, and document review, and the resulting platform was implemented as a three-layer web application over a document-oriented model of twenty-four collections. The system integrates role-based access control, product and inventory management, bin-level location tracking, stock receiving and dispatch, adjustment and audit workflows with variance detection, vendor and customer records, on-demand reporting, threshold-based alerting, and assisted natural-language retrieval, and it was deployed and evaluated on commodity hardware without scanning equipment.

Five of the seven specified functional test cases were executed and passed, two remain unexecuted and are reported as gaps, and usability, performance, and security reviews reported satisfactory operation under the conditions tested. The study demonstrates the technical feasibility of combining conventional warehouse management with web-based accessibility and assisted retrieval with reported deployment-related expenditure of USD 425 in the evaluated configuration.

The study does not quantify operational improvement, retrieval accuracy, or large-scale deployment performance, and these limitations should guide subsequent work rather than be read as established outcomes. Priorities for further development are quantified field evaluation against the incumbent process, tolerance of intermittent connectivity, mobile capture at the point of handling, and lead-time-aware replenishment.

Declarations

Ethics approval

The study was conducted in accordance with the ethical requirements of the university.

Data availability

The data and source code are not deposited in a public repository; availability is subject to the authors' institutional and operational constraints.

Competing interests

The authors declare no competing interests.

Funding

This study received no external funding.

Author contributions

Mustaf Abdi Ibrahim, Bashir Hassan Mohamed, and Abduqadir Abdixamiid Abukar contributed to the system study and development; Ahmed Hussein Elmi supervised the work. The authors approved the final manuscript.

Figure provenance. Figures 1–4, 7, A1 and A2 were produced by the authors as original graphics. Figures 5 and 6 are screenshots of the authors' own system. No third-party figures are reproduced.

Acknowledgements
The authors thank the lecturers of the Faculty of Computer Science and Information Technology, and Hormuud University for academic support and resources.

References

[1] R. B. M. de Koster, A. L. Johnson, and D. Roy, "Warehouse design and management," International Journal of Production Research, vol. 55, no. 21, pp. 6327–6330, 2017.
[2] I. H. Mohamud, M. A. Kafi, S. A. Shahron, N. Zainuddin, and S. Musa, "The role of warehouse layout and operations in warehouse efficiency: A literature review," Journal Européen des Systèmes Automatisés, vol. 56, no. 1, 2023.
[3] K. Shanmugamani and F. Mohamad, "The implementation of warehouse management system (WMS) to improve warehouse performance in business to business (B2B)," International Journal of Industrial Management, vol. 17, no. 4, pp. 231–239, 2023, doi: 10.15282/ijim.17.4.2023.10091.
[4] Y. Zhang and F. Pan, "Design and implementation of a new intelligent warehouse management system based on MySQL database technology," Informatica, vol. 46, no. 3, 2022.
[5] L. Zhen and H. Li, "A literature review of smart warehouse operations management," Frontiers of Engineering Management, vol. 9, no. 1, pp. 31–55, 2022, doi: 10.1007/s42524-021-0178-9.
[6] J. Kembro and A. Norrman, "The transformation from manual to smart warehousing: An exploratory study with Swedish retailers," The International Journal of Logistics Management, vol. 33, no. 5, pp. 107–135, 2022, doi: 10.1108/IJLM-11-2021-0525.
[7] L. Custódio and R. Machado, "Flexible automated warehouse: A literature review and an innovative framework," The International Journal of Advanced Manufacturing Technology, vol. 106, pp. 533–558, 2020, doi: 10.1007/s00170-019-04588-z.
[8] A. Jarašūnienė, K. Čižiūnienė, and A. Čereška, "Research on impact of IoT on warehouse management," Sensors, vol. 23, no. 4, art. 2213, 2023, doi: 10.3390/s23042213.
[9] M. M. H. Shahadat, M. Nekmahmud, P. Ebrahimi, and M. Fekete-Farkas, "Digital technology adoption in SMEs: What technological, environmental and organizational factors influence in emerging countries?," Global Business Review, 2023, doi: 10.1177/09721509221137199.
[10] J. Kembro and A. Norrman, "Exploring trends, implications and challenges for logistics information systems in omni-channels: Swedish retailers perception," International Journal of Retail & Distribution Management, vol. 47, no. 4, pp. 384–411, 2019.
[11] J. D. Croston, "Forecasting and stock control for intermittent demands," Operational Research Quarterly, vol. 23, no. 3, pp. 289–303, 1972.
[12] A. A. Syntetos and J. E. Boylan, "The accuracy of intermittent demand estimates," International Journal of Forecasting, vol. 21, no. 2, pp. 303–314, 2005.
[13] R. Heeks, "Information systems and developing countries: Failure, success, and local improvisations," The Information Society, vol. 18, no. 2, pp. 101–112, 2002.
[14] J. Brooke, "SUS: A quick and dirty usability scale," in Usability Evaluation in Industry, P. W. Jordan et al., Eds. London, U.K.: Taylor & Francis, 1996, pp. 189–194.
[15] A. Bangor, P. T. Kortum, and J. T. Miller, "An empirical evaluation of the System Usability Scale," International Journal of Human–Computer Interaction, vol. 24, no. 6, pp. 574–594, 2008.

[16] F. D. Davis, “Perceived usefulness, perceived ease of use, and user acceptance of information technology,” MIS Quarterly, vol. 13, no. 3, pp. 319–340, 1989.
[17] A. Fumi, L. Scarabotti, and M. M. Schiraldi, “The effect of slot-code optimization in warehouse order picking,” International Journal of Engineering Business Management, vol. 5, art. 20, pp. 1–10, 2013.

Appendix A. Logical Data Model

The logical data model is presented in two figures for legibility. Fig. A1 covers access control, warehouse topology, product master data, and the stock transaction and reconciliation collections that carry the operational core. Fig. A2 covers the commercial, financial, personnel, and logistics collections. Primary and foreign key markers denote identity and reference relationships in the logical model; in the implemented document store these are held as document identifiers rather than enforced constraints.

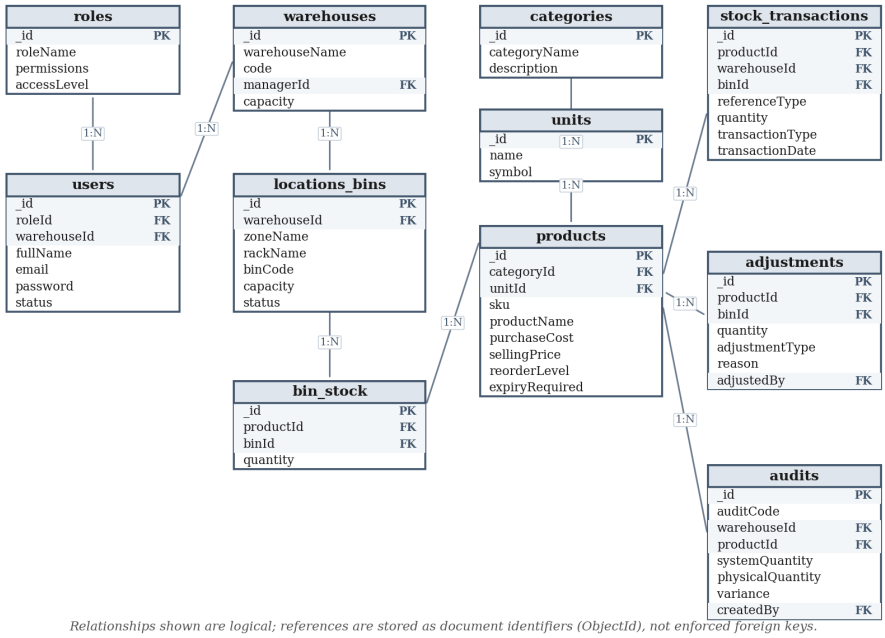
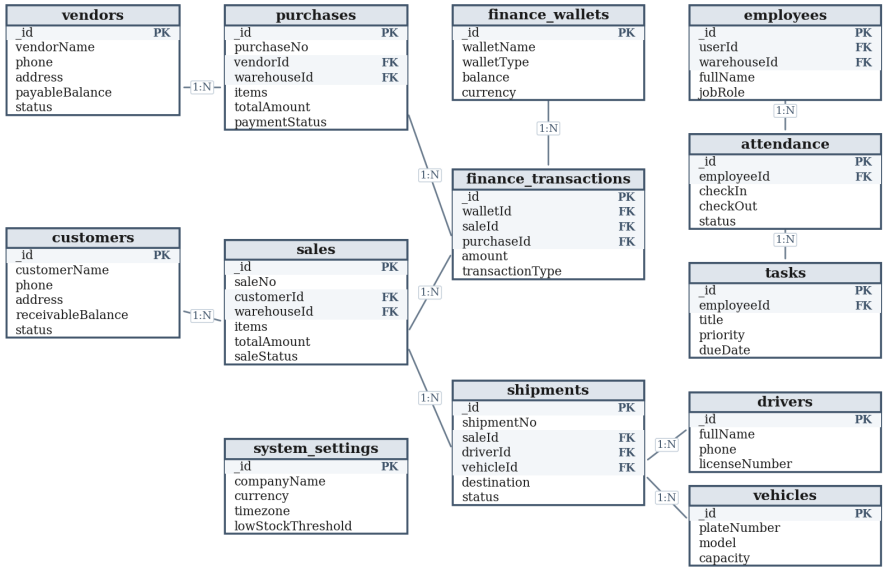


Fig. A1. Logical data model: access control, warehouse topology, product master data, stock transactions, and reconciliation.



Relationships shown are logical; references are stored as document identifiers (ObjectId), not enforced foreign keys.

Fig. A2. Logical data model: trading partners, purchasing and sales, finance, personnel, logistics, and system configuration.