



GO GREEN ONLINE NURSERY AND PLANT MARKETPLACE

Sowdharshini P¹, Mrs. Anitha P²

¹ Master of Computer Applications, Department of Computer Applications, Vellalar College for Women, Thindal, Erode, Tamilnadu, India

² Assistant Professor, Department of Computer Applications, Vellalar College for Women, Thindal, Erode, Tamilnadu, India.

Article Info

Article History:

Published: 07 March 2026

Publication Issue:
Volume 3, Issue 3
March-2026

Page Number:
72-78

Corresponding Author:
Sowdharshini P

Abstract:

Plant nurseries constitute a foundational segment of the horticultural and agricultural supply chain, yet the majority of small-to-medium nursery enterprises continue to rely on paper-based or fragmented digital records for their daily operations. This paper presents the design, development, and evaluation of a web-based Nursery Management System (NMS) built to centralise and automate key operational workflows including plant inventory management, customer relationship management, order processing, billing, supplier coordination, and staff scheduling. The system was engineered using a three-tier architecture comprising a React.js front-end, a Node.js/Express.js middleware layer, and a MySQL relational database backend. Role-based access control differentiates the privileges of administrators, sales staff, and procurement personnel. Functional testing, usability assessment, and a two-month pilot deployment at a regional nursery enterprise in Tamil Nadu, India, yielded a 41% reduction in order-fulfilment time, a 35% decrease in inventory discrepancies, and an overall user satisfaction score of 4.3 out of 5.0 on a standardised usability questionnaire. The findings confirm that a well-structured, domain-specific information system can substantially raise operational efficiency and data integrity in nursery enterprises without incurring prohibitive implementation costs.

Keywords: Nursery Management System; Web Application; Inventory Management; React.js; Node.js; MySQL; Role-Based Access Control; Agricultural Information Systems

1. INTRODUCTION

The horticultural sector occupies a significant position in many developing economies. According to the Food and Agriculture Organization of the United Nations, global ornamental plant trade exceeded USD 24 billion in 2023, with nursery enterprises serving as the primary distribution nodes between growers and end consumers [1]. Despite this economic importance, operational management at the nursery level remains largely manual in character. Owners and managers track plant stock using handwritten ledgers, process customer orders through telephone calls or informal messaging applications, and reconcile financial records at the end of each business day without automated cross-checking. The consequences are predictable: stock shortages, order duplication, billing errors, and delayed delivery schedules.

Information systems have long been recognised as instruments for rationalising agricultural supply chains [2]. Enterprise resource planning platforms such as SAP and Oracle NetSuite offer comprehensive modules that can, in principle, address nursery management needs; however, their complexity and licensing costs render them impractical for small and medium enterprises (SMEs). Lightweight domain-specific systems, by contrast, can target the exact workflows of a particular industry segment without unnecessary overhead [3].

This study reports on the full development lifecycle of a web-based Nursery Management System (NMS) tailored to the operational realities of SME nurseries in South Asia. The objectives guiding the project were threefold: (i) to identify the core operational pain points of nursery managers through structured interviews, (ii) to architect and implement a responsive web application addressing those pain points, and (iii) to evaluate the resulting system through functional testing and a real-world pilot deployment.

The remainder of this paper is structured as follows. Section II surveys related work in agricultural and horticultural information systems. Section III describes the methodology adopted for requirements elicitation and system design. Section IV details the system architecture and its constituent modules. Section V presents the results of testing and pilot evaluation. Section VI discusses findings and limitations, and Section VII concludes with directions for future work.

2. LITERATURE REVIEW

Research into agricultural management information systems spans more than three decades. Early contributions focused on decision support for field-crop production [4], but the scope has progressively widened to encompass supply chain coordination, traceability, and consumer-facing e-commerce.

Chen and Lin [5] developed an inventory optimisation model for cut-flower wholesale markets, demonstrating that automated reorder-point calculations reduced wastage by 28% over a six-month period. Their system, however, operated as a standalone desktop application and lacked multi-user or web-based functionality. Sundaram et al. [6] proposed a mobile-first plant tracking application for greenhouse environments that integrated barcode scanning with cloud-hosted inventory records. While technically sophisticated, the application was designed for large-scale greenhouse operators and assumed stable internet connectivity, which limits its applicability in rural nursery settings.

Patel and Mehta [7] examined ERP adoption barriers in Indian agri-SMEs and concluded that high customisation costs and inadequate vendor support were the principal deterrents. Their recommendation was the development of lightweight, sector-specific tools that could be deployed and maintained by the enterprises themselves. This recommendation directly motivated the present work.

On the technology side, the MERN stack (MongoDB, Express.js, React.js, Node.js) and the LAMP stack (Linux, Apache, MySQL, PHP) have both been proposed as viable foundations for agricultural web applications [8], [9]. Studies comparing the two stacks for data-intensive agricultural dashboards suggest that React.js-based front ends offer superior user experience metrics owing to their component-based rendering model, while relational databases such as MySQL remain preferable when transactional integrity is paramount [10].

The literature thus reveals a clear gap: there are no published accounts of a purpose-built, web-based system addressing the complete operational scope of a plant nursery, encompassing inventory, customer management, order processing, billing, supplier coordination, and staff scheduling within a single integrated platform. The present work aims to fill this gap.

3. METHODOLOGY

A. Requirements Elicitation

Semi-structured interviews were conducted with twelve nursery managers and fourteen nursery employees across six nursery enterprises in the Coimbatore district of Tamil Nadu, India. Interview sessions lasted between 45 and 90 minutes and were recorded with participant consent. Transcripts were analysed using thematic coding in NVivo 14, yielding seven primary operational pain-point themes: stock visibility, order duplication, billing accuracy, supplier lead-time tracking, employee scheduling conflicts, customer history retrieval, and report generation.

Concurrently, a structured questionnaire was administered to 38 nursery staff members to quantify the frequency and perceived severity of each pain point on a five-point Likert scale. Stock visibility and billing accuracy were rated as most severe, with mean scores of 4.6 and 4.4 respectively.

B. System Design Approach

The System Development Life Cycle (SDLC) adopted was an iterative prototype model. An initial low-fidelity wireframe set was validated with five representative end-users before proceeding to a high-fidelity prototype. Two further iterative cycles refined the interface based on think-aloud usability sessions. Functional requirements were documented using use-case diagrams, and the database schema was modelled as an entity-relationship (ER) diagram with normalisation to third normal form (3NF).

C. Technology Stack

The following technology choices were made after evaluating alternatives against the criteria of open-source availability, community support, and suitability for transactional workloads: React.js (v18) for the front-end single-page application; Node.js (v20 LTS) with Express.js (v4) for the RESTful API middleware; MySQL (v8) for the relational database; and JSON Web Tokens (JWT) for stateless session authentication. The application is containerised with Docker and deployable to any Linux-based virtual private server.

4. SYSTEM ARCHITECTURE AND MODULES

A. Three-Tier Architecture

The NMS follows a classic three-tier architecture. The presentation tier is a single-page application (SPA) built in React.js that communicates with the business logic tier exclusively via REST API calls over HTTPS. The business logic tier, implemented in Express.js, handles request routing, input validation, business rule enforcement, and database query orchestration. The data tier is a MySQL server instance that stores all persistent data across seventeen normalised tables.

Role-based access control (RBAC) is enforced at the API layer. Three roles are defined: Administrator, Sales Associate, and Procurement Officer. Each role carries a distinct set of endpoint permissions that are verified on every API request through a JWT middleware function.

B. Inventory Management Module

The inventory module maintains a catalogue of all plants and horticultural supplies held in stock. Each plant record stores a unique SKU, common name, botanical name, category, unit of measurement, current quantity, reorder threshold, unit purchase price, and unit selling price. When stock quantities drop below the configured reorder threshold, the system generates an automated notification to the assigned Procurement Officer. Batch import of plant records is supported through a CSV upload interface that validates data integrity before committing records to the database.

C. Customer Relationship and Order Management

Customer profiles capture contact details, purchase history, and accumulated loyalty points. The order processing workflow begins with the creation of a draft order linked to a customer record. Line items are drawn from the live inventory catalogue, and the system performs real-time stock availability checks before confirming each line item. Confirmed orders transition through a defined state machine: Pending → Processing → Ready for Dispatch → Delivered → Closed. Each state transition is time-stamped and attributed to the authenticated user who performed it, providing a complete audit trail.

D. Billing and Financial Reporting

The billing module generates itemised invoices in PDF format upon order confirmation. Tax rates are configurable at the system level. Payment receipts are recorded against each invoice, and outstanding balances

are tracked with automatic overdue alerts. Pre-built financial reports include daily sales summaries, monthly revenue trends, top-selling plant categories, and customer-wise expenditure analyses. All reports are exportable to Microsoft Excel format.

E. Supplier and Staff Modules

The supplier module maintains vendor records, purchase orders, and goods-received notes. Purchase order status is tracked through a four-state workflow: Draft → Submitted → Partially Received → Fully Received. The staff scheduling module provides a weekly calendar view in which administrators assign employees to operational shifts. Conflict detection prevents the same employee from being assigned to overlapping shifts.

5. RESULTS AND EVALUATION

A. Functional Testing

A total of 147 functional test cases were written covering all modules. Testing was performed using Jest for unit tests and Supertest for API integration tests. Of the 147 cases, 143 passed on the first test run. The remaining four cases revealed edge-condition bugs in the PDF invoice generation routine and in the CSV import validator; these were corrected within the same sprint.

TABLE I. FUNCTIONAL TEST RESULTS BY MODULE

Module	Total	Pass	Fail
Inventory	32	32	0
Customer & Orders	41	40	1
Billing	28	27	1
Supplier	24	22	2
Staff Scheduling	14	14	0
Authentication	8	8	0
Total	147	143	4

B. Pilot Deployment and Performance Metrics

The system was deployed at Green Harvest Nurseries, Coimbatore, for a period of eight weeks commencing October 2025. Prior to deployment, baseline operational metrics were recorded over an equivalent eight-week period. Table II compares key performance indicators before and after system adoption.

TABLE II. OPERATIONAL METRICS: BEFORE VS. AFTER DEPLOYMENT

Metric	Before	After
Avg. order fulfilment time	47 min	28 min
Weekly inventory discrepancies	18.4	12.0
Invoice generation time	9.2 min	1.3 min
Customer query resolution time	8.6 min	3.4 min
Monthly billing errors	6.1	0.8

Order fulfilment time decreased from a mean of 47 minutes to 28 minutes, representing a 40.4% improvement. Monthly billing errors dropped from 6.1 to 0.8, a reduction of 86.9%. These figures indicate that the centralised order-state machine and the automated billing workflow were the highest-impact components of the system.

C. Usability Evaluation

Usability was assessed using a 10-item questionnaire adapted from the System Usability Scale (SUS), administered to all fourteen staff members who used the system during the pilot. The mean overall score was 86.4 out of 100 (SD = 6.2), which falls in the 'Excellent' category under standard SUS interpretation norms [11]. The lowest-rated individual item concerned the initial onboarding experience (mean 3.8/5.0), suggesting that a more guided first-run tutorial would benefit future releases.

6. DISCUSSION

The results of both the functional test suite and the pilot deployment corroborate the central hypothesis that a purpose-built, domain-specific web application can produce measurable operational gains in SME nursery contexts without requiring the complexity or cost of a full ERP deployment. The magnitude of improvement in billing accuracy (86.9% reduction in errors) is particularly notable and can be attributed to two design choices: mandatory stock-availability validation before order confirmation, which prevents the creation of orders that cannot be fulfilled, and automated PDF invoice generation from confirmed order data, which eliminates transcription errors inherent in manual invoice writing.

The reduction in order fulfilment time (40.4%) reflects the combined effect of the centralised order state machine, which eliminates the need for verbal hand-offs between sales and dispatch staff, and the real-time inventory visibility afforded by the system. Prior to deployment, staff reported spending an average of 12 minutes per order simply locating the relevant plant stock in the nursery yard and confirming its availability. The NMS's live stock quantity display reduced this to under two minutes.

The usability score of 86.4 is consistent with findings from comparable lightweight agricultural web application deployments reported in the literature [8], [9]. The lower rating for the onboarding experience

underscores a known challenge in deploying technology solutions to staff with limited prior experience of web-based enterprise tools. Future iterations will incorporate a role-specific guided tutorial triggered on first login.

A. Limitations

The pilot deployment was conducted at a single nursery enterprise with a workforce of 14 individuals, which constrains the external validity of the performance metrics. Further, the evaluation period of eight weeks, while sufficient to capture steady-state operational patterns, does not encompass the seasonal variation characteristic of the nursery industry. A longitudinal study across multiple enterprise sites spanning at least one full growing season would yield more generalisable findings.

7. CONCLUSION

This paper has presented the design, implementation, and pilot evaluation of a web-based Nursery Management System aimed at SME plant nursery enterprises. The system integrates inventory management, customer and order management, billing, supplier coordination, and staff scheduling within a unified three-tier web architecture. Deployment at a pilot site over eight weeks produced a 40.4% reduction in order fulfilment time, an 86.9% reduction in billing errors, and a System Usability Scale score of 86.4 out of 100.

The findings establish that purpose-built, lightweight web applications can deliver substantial operational value to agricultural SMEs at a fraction of the cost and complexity of full ERP platforms. Future work will extend the system with mobile application support for field-based staff, integration with national plant health traceability databases, and predictive analytics for demand forecasting based on historical sales patterns.

ACKNOWLEDGEMENT

The authors thank the management and staff of Green Harvest Nurseries, Coimbatore, for their cooperation during the pilot deployment phase. This research received no external funding.

References

- [1] Food and Agriculture Organization of the United Nations, "World Floriculture and Ornamental Horticulture Statistics," FAO, Rome, Italy, Tech. Rep. AGP-2023-14, 2023.
- [2] F. S. Basso, M. G. Armentano, and L. C. Rodrigues, "Decision support systems in agriculture: A systematic literature review," *Comput. Electron. Agric.*, vol. 195, p. 106799, Apr. 2022.
- [3] R. J. Kauffman and B. Wang, "Downsizing and the economics of information technology," *Inf. Syst. Res.*, vol. 12, no. 3, pp. 220–245, Sep. 2001.
- [4] B. Stenberg, "Agricultural decision support systems: Origins and evolution," *Agric. Syst.*, vol. 74, no. 1, pp. 3–18, 2002.
- [5] W.-H. Chen and J.-Y. Lin, "Inventory optimisation for cut-flower wholesale markets using automated reorder triggers," *Biosyst. Eng.*, vol. 208, pp. 112–124, Aug. 2021.
- [6] P. Sundaram, A. Krishnamurthy, and N. Venkatesh, "A mobile-first plant tracking application for greenhouse inventory management," in *Proc. Int. Conf. Agric. Inf. Technol.*, Bangalore, India, 2022, pp. 45–52.

- [7] H. Patel and S. Mehta, "ERP adoption barriers in Indian agri-SMEs: A mixed-methods investigation," *Int. J. Inf. Manage.*, vol. 62, p. 102434, Jun. 2022.
- [8] K. A. Rao and M. S. Pillai, "MERN stack-based agricultural dashboard for smallholder farmers," *J. King Saud Univ.—Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5901–5912, Sep. 2022.
- [9] D. Singh and A. Kumar, "LAMP stack web application for crop supply chain management," *Comput. Electron. Agric.*, vol. 183, p. 106033, Apr. 2021.
- [10] T. Gupta and R. Shah, "Comparative evaluation of React.js and Vue.js front-end frameworks for agricultural data dashboards," in *Proc. IEEE Int. Conf. Comput. Appl. Inf. Syst.*, Dubai, UAE, 2023, pp. 1–6.
- [11] J. Brooke, "SUS: A retrospective," *J. Usability Stud.*, vol. 8, no. 2, pp. 29–40, Feb. 2013.