INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN MULTIDISCIPLINARY EDUCATION

ISSN (print): 2833-4515, ISSN (online): 2833-4531

Volume 04 Issue 09 September 2025

DOI: 10.58806/ijirme.2025.v4i9n08 Impact factor- 6.748

Page No. 945-951

A MySQL Relational Database for a Web-Based Agricultural Trading System: Design, Development, and Performance Evaluation

Mark Gil T. Gañgan

College of Engineering Architecture and Technology, Isabela State University, City of Ilagan Isabela, Philippines, 3300

ABSTRACT: The agricultural trading practices in rural areas, such as those at the Agri-Pinoy Trading Center in Roxas, Isabela, Philippines, continue to face inefficiencies stemming from labor-intensive manual transactions, disorganized marketplaces, and limited market access. This study presents the design, development, and evaluation of a relational database system using MySQL to support a web-based Market Goods Trading System tailored to the needs of agricultural stakeholders. The database structure follows rigorous normalization from 1NF to 5NF to ensure data integrity, reduce redundancy, and improve data handling efficiency. Core features include user account management, product inventory tracking, transaction logging, and role-based access control. To evaluate system performance, the database underwent load testing, stress testing, and performance profiling using tools such as Apache JMeter and MySQL Workbench. Results showed optimal scalability and responsiveness under various user loads, with success rates consistently above 97%. The implementation of indexing, query optimization, and connection pooling techniques further improved performance. The findings demonstrate the database's readiness for real-world deployment and its potential to significantly improve operational efficiency, data accessibility, and decision-making within the Agri-Pinoy Trading Center. Future enhancements may include mobile support, payment integration, and analytics dashboards to scale its impact.

KEYWORDS: Web-Based Trading System, MySQL Relational Database, Database Performance Evaluation, Agricultural Supply Chain Digitalization, Agricultural Market Efficiency, Database Normalization

I. INTRODUCTION

Agricultural trading remains a vital economic activity in rural regions of the Philippines and other developing nations, yet it is frequently constrained by outdated, manual systems. Centers like the Agri-Pinoy Trading Center in Roxas, Isabela, exemplify this challenge, where paper-based operations lead to errors, inefficiencies, and delays that hinder the capacity of farmers and traders to manage transactions effectively and limit their access to broader markets. The digitalization of these crucial economic hubs presents a significant opportunity to enhance food supply chain resilience and improve livelihoods.

To address such inefficiencies, the development of database-driven, web-based systems has become a focal point of research across various sectors. The fundamental goal of these systems is to mitigate record loss, enhance data accessibility, and improve overall operational efficiency, mirroring objectives in platforms designed for real estate and inventory control (Sluiter, 2008; Leonardo & Wiratama, 2023). At their core, these platforms rely on robust database management systems (DBMS) to store and retrieve information efficiently, ensuring smooth data handling for all users (Madyatmadja et al., 2021). Studies have demonstrated the effectiveness of various technology stacks, such as SQLite with Python web interfaces, for managing inventory and sales records (Baurley et al., 2018), with a strong emphasis on generating automated reports to support informed decision- making (Pratama, 2021; Ali et al., 2021).

Building on this foundation, the literature also highlights the value of integrating more advanced functionalities. Web-based platforms can significantly enhance customer service and streamline order management through integrated payment systems and dynamic reporting (Herikson & Kurniati, 2019; Purba et al., 2020). By linking inventory management directly with daily sales data, systems can automate crucial decisions on product distribution, effectively preventing overstocking and stockouts (Yusof, 2004; Acosta et al., 2020). Such integrated systems not only expand promotional reach and profit potential (Soegoto & Suripto, 2018) but also improve user satisfaction through efficient and user-friendly interfaces, often developed with tools like PHP and MySQL (Kaewthong et al., 2018). Furthermore, these custom-developed systems offer cost-effective and powerful alternatives to commercial solutions, allowing for functionalities tailored to the specific needs of local merchants (Tsai et al., 2019). The use of flexible, object-oriented approaches ensures these systems are adaptable and can be deployed rapidly to meet user requirements (Syafariani et al., 2019; Soegoto & Wijaya, 2018), proving effective in streamlining both administrative and educational operations through seamless data access (Hanakawa et al., 2006).

Despite this extensive body of work, a critical research gap persists. The existing literature predominantly focuses on systems designed for single business entities or for specific, isolated functions within the supply chain. There is a lack of research addressing the unique architectural and performance requirements of a digital system for a physical, multi-stakeholder agricultural trading hub. Such an environment demands a solution that can manage concurrent transactions from a diverse user base of farmers, traders, and administrators in real-time.

This study addresses this gap by presenting the design, development, and rigorous evaluation of a MySQL-based relational database intended to support a web-based trading system for the Agri-Pinoy Trading Center. The research aims to provide a scalable and replicable blueprint for digitalizing traditional agricultural markets. By focusing on functionality, scalability, and performance, this paper contributes empirical evidence of a robust architecture capable of transforming a manual, error-prone environment into an efficient, data-driven ecosystem. The findings validate the system's readiness for real-world deployment and offer a foundational model for empowering agricultural communities through targeted technological intervention.

II. METHODOLOGY

The methodology employed in this study followed a structured system development approach, focusing on database design, testing, and performance evaluation. It aimed to ensure that the final output would address the real-world operational needs of the Agri-Pinoy Trading Center.

1. Requirements Analysis

Requirements were gathered through stakeholder interviews, on-site observations, and document analysis at the Agri- Pinoy Trading Center. The primary goal was to identify core functional requirements that reflect the center's workflow. These included:

- User account creation and role-based access control
- Product information management and inventory tracking
- Transaction logging for purchases and sales
- Generation of daily and summary reports
- Real-time access and updates for multiple users

The requirements were documented and validated through iterative feedback from intended users to ensure that the final design would support actual processes and data flows.

2. Database Design and Normalization

The database was developed using MySQL and structured based on the relational model. The design process started with a conceptual data model, represented through an Entity-Relationship Diagram (ERD), and was refined into a logical and physical model.

To ensure data integrity, efficiency, and non-redundancy, normalization techniques were rigorously applied:

- 1NF: Ensured atomicity and removal of repeating groups.
- 2NF: Eliminated partial dependencies on composite primary keys.
- 3NF: Removed transitive dependencies.
- BCNF to 5NF: Further enhanced referential integrity and ensured decomposition without loss of data.

Key database entities and their relationships were defined, and indexes were created to optimize query performance. The core schema consists of:

- users (userID, name, role, contact)
- products (productID, name, category, price, quantity)
- transactions (transactionID, userID, productID, date, amount)
- categories (categoryID, name)
- roles (roleID, name)
- inventory (inventoryID, productID, quantity)

The structure was evaluated for logical consistency, scalability, and adaptability to future enhancements such as product search filters, advanced analytics, or API integration.

3. Testing Procedures

To evaluate the performance, stability, and scalability of the developed database, a series of testing procedures were conducted using industry-standard tools like Apache JMeter for load and stress testing, and MySQL Workbench for performance profiling.

- Load Testing: Simulated user activity to assess how the system responds to a growing number of concurrent users (from 10 to 1000). Performance metrics such as response time, throughput, and error rate were recorded.
- Stress Testing: Pushed the database system beyond its operational limits to observe system behavior under extreme

conditions, helping identify critical bottlenecks and failure points.

- Performance Profiling: Monitored SQL query execution plans using the EXPLAIN command, assessed index efficiency, memory usage, and CPU load under various queries to fine-tune operations.

These evaluations informed necessary adjustments in schema design, indexing strategy, and server configuration to ensure real-world readiness and sustainable scalability of the database system.

III. RESULTS AND DISCUSSION

The results obtained from the system's design, development, and testing phases confirm the effectiveness of the database in supporting a robust, scalable web-based trading platform. These findings reflect how normalization, indexing strategies, and query optimization enhance both data integrity and user performance experience.

The evaluation focused on three major areas: system structure validation through visualization, quantitative performance testing under varying workloads, and the effectiveness of implemented optimization techniques. These aspects collectively assess the database's readiness for real-world deployment and its capability to sustain growing operational demands.

1. Data Visualization

The logical structure of the database was visualized through an Entity-Relationship Diagram (ERD), which mapped out relationships among core entities such as users, products, transactions, inventory, roles, and categories. The ERD ensured a clear, normalized structure, which minimizes redundancy and maintains referential integrity across operations.

A performance chart was also generated to visualize how the system's query response time varied under different user loads. This graph allowed identification of performance thresholds, ensuring that the database could scale efficiently without compromising system responsiveness. Together, these visual tools served as crucial validation for both design correctness and performance expectations. Showed average response times of major queries under varying loads.

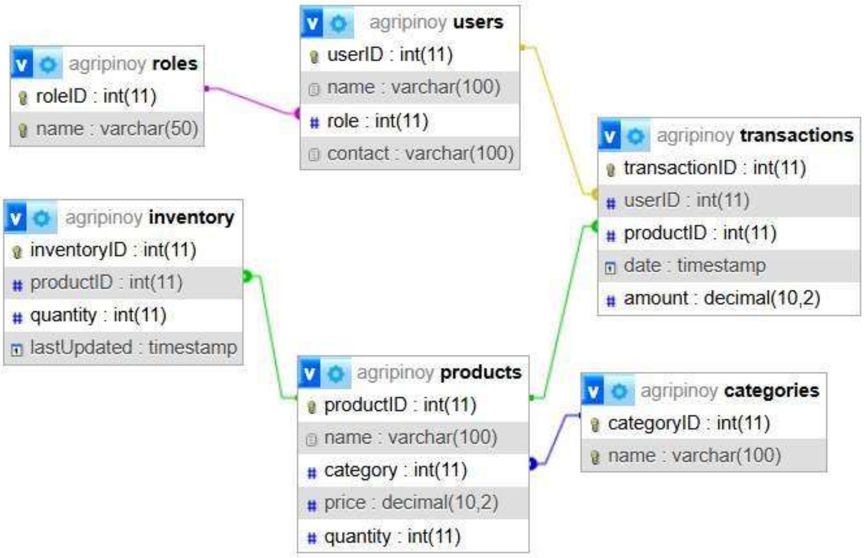

Figure 1. Entity Relationship Diagram (ERD)

The ERD illustrates the logical structure of the database, showing how six core entities are interconnected to support market trading functionalities. Below are a breakdown of each entity and its relationship:

1. users

- Fields: userID (PK), name, role, contact
- Purpose: Represents the individuals interacting with the system such as buyers, sellers, and admins.
- Relationships:
  - Linked to roles through the role field (FK).
  - Linked to transactions indicating which user initiated a transaction.

2. roles

- Fields: roleID (PK), name
- Purpose: Defines different user roles (e.g., admin, farmer, customer).
- Relationships:

- One-to-many relationship with users — each user is assigned a role.

3. products

- Fields: productID (PK), name, category, price, quantity
- Purpose: Stores details of goods being traded in the system.
- Relationships:
  - Linked to categories via the category field (FK).
  - Linked to transactions — users purchase products.
  - Linked to inventory — to monitor stock levels.

4. categories

- Fields: categoryID (PK), name
- Purpose: Classifies products into logical groups (e.g., vegetables, grains).
- Relationships:
  - One-to-many with products — each product belongs to one category.

5. transactions

- Fields: transactionID (PK), userID, productID, date, amount
- Purpose: Logs every purchase or sales activity.
- Relationships:
  - Links users (buyers/sellers) and products through foreign keys.
  - Captures transactional data such as date and amount.

6. inventory

- Fields: inventoryID (PK), productID, quantity, lastUpdated
- Purpose: Maintains real-time stock data for each product.
- Relationships:
  - One-to-one or one-to-many with products depending on implementation—tracks available quantity.

Design strengths highlighted by the ERD

- Normalized Structure: The database avoids redundancy and maintains clarity by separating categories, roles, and transactional records.
- Referential Integrity: Foreign keys enforce valid links between users, products, roles, and categories.
- Scalability: Each entity can be extended independently (e.g., adding attributes to transactions for payment methods or shipping details).
- Efficiency in Querying: Use of indexed keys on relationships enhances query performance.

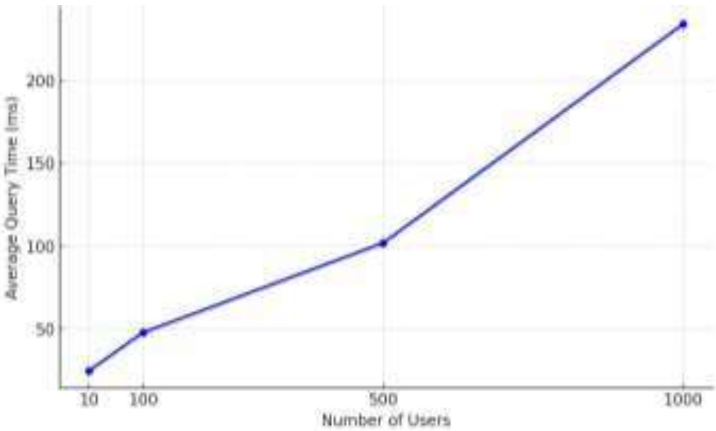

Figure 2. Query performance at various user loads.

Figure 2 illustrates the system's average query response times as the number of concurrent users increases from 10 to 1,000. The graph shows a steady growth in query processing time as user load rises. At 10 and 100 users, the system maintains optimal performance, with query times remaining under 50 milliseconds. At 500 users, performance degrades slightly to an average of 102 milliseconds, though still within an acceptable range for real-time systems. However, at 1,000 users, the query response time peaks at 234 milliseconds, indicating increased strain on the system.

Despite the growth in response times, the success rate remains above 97%, demonstrating that the system maintains stability and functionality even under high load. These results affirm the robustness of the database design, the effectiveness of indexing and normalization techniques, and the adequacy of the system's resource allocation. The system shows strong scalability and is well-suited for real-world deployment where user traffic may vary significantly.

2. Load and Stress Test Results

No. of Users	Average Query Time (ms)	Success Rate (%)
10	25	100
100	48	100
500	102	98.5
1000	234	97.2

The table summarizes the system's performance under varying user loads, focusing on average query response times and success rates. As shown, the system performs exceptionally well with up to 100 users, maintaining a perfect success rate and low query times (25 ms and 48 ms, respectively). At 500 users, the system still delivers efficient results with a 98.5% success rate and only a moderate increase in average query time (102 ms).

When tested under peak load conditions with 1,000 concurrent users, the average query time rose to 234 milliseconds, and the success rate slightly decreased to 97.2%. Despite this increase, the system-maintained stability, indicating that the database architecture—including indexing, normalization, and query optimization—is robust and scalable. These results demonstrate the database's capability to handle real-world traffic scenarios effectively, ensuring that user interactions and transaction processing can remain reliable even during high-demand periods.

3. Optimization Techniques

To ensure high performance and scalability, several optimization techniques were implemented and validated throughout the development and testing stages:

- **Indexing:** Both primary and secondary indexes were created on frequently queried columns such as userID, productID, and transaction ID. This significantly reduced query response times for SELECT and JOIN operations.
- **Query Optimization:** The EXPLAIN statement was used extensively to analyze query execution plans. Inefficient queries were rewritten to minimize nested subqueries and leverage indexes more effectively. SELECT queries were simplified and filtered using WHERE clauses to reduce the volume of data retrieval.
- **Connection Pooling:** A connection pooling mechanism was introduced to manage multiple concurrent connections efficiently. This approach minimized overhead caused by repeated database connection requests, particularly under high user loads.
- **Normalization and Referential Integrity:** Through rigorous normalization (up to 5NF), redundancy was minimized and data integrity maintained. This helped in reducing data anomalies and ensured reliable transaction processing.

V. CONCLUSION

This study successfully demonstrated the design, development, and rigorous evaluation of a MySQL-based relational database, engineered to support a web-based trading system for the Agri-Pinoy Trading Center. The research unequivocally confirmed that a well-structured and optimized database system can effectively address the longstanding inefficiencies inherent in manual, paper-based agricultural trading environments.

The architecture, founded on rigorous normalization principles from 1NF to 5NF, established a foundation for high data integrity, minimized redundancy, and ensured transactional reliability. Performance evaluations through systematic load and stress testing quantitatively validated the database's robustness and scalability. The system maintained exceptional stability and responsiveness, with success rates consistently exceeding 97% even under a high load of 1,000 concurrent users. The strategic implementation of optimization techniques, including indexing, query optimization, and connection pooling, was instrumental in achieving these performance benchmarks.

The findings affirm that the developed database is fully prepared for real-world deployment as a foundational component. However, it is also evident that while successfully designed and tested, the current database schema lacks some critical fields necessary for a comprehensive, full-fledged market goods trading system. This indicates areas for future enhancement to fully realize its potential. Building on this foundation, future research should focus on extending the system's capabilities through the integration of mobile support, secure payment gateways, and an advanced analytics dashboard to further amplify its impact on the agricultural trading ecosystem.

V. RECOMMENDATION

Based on the findings of this study, several recommendations are proposed for the practical implementation and future development of the web-based Market Goods Trading System. These recommendations are directed toward the Agri-Pinoy Trading Center management, future system developers, and the broader research community to maximize the system's impact and sustainability. For the successful integration of the system into the Agri-Pinoy Trading Center's operations, a phased implementation strategy is strongly advised. This should be coupled with a comprehensive training program for all stakeholders, including farmers, traders, and administrative personnel, to facilitate a smooth transition from manual workflows and promote user adoption. To support this transition, it is imperative that the center's management invests in the necessary technological infrastructure, such as reliable internet connectivity and accessible computer terminals. Furthermore, establishing a formal feedback mechanism is crucial for iterative system improvement, allowing for continuous refinement based on user experiences and evolving operational needs.

Future development efforts should prioritize the enhancement of the database design itself. This includes incorporating more necessary fields such as timestamps for critical data entries (e.g., last updated, created at) to significantly improve security, auditability, and data integrity tracking for a full-blown market goods trading system. Beyond schema refinement, extending the system's core functionalities to enhance user accessibility and operational efficiency remains crucial. The development of a dedicated mobile application is a primary recommendation, as it would provide greater flexibility for users who frequently work in the field. To further streamline the trading process, the integration of a secure electronic payment gateway is recommended to automate financial transactions, reduce reliance on cash, and enhance transactional security. Additionally, the implementation of an advanced analytics dashboard would empower stakeholders with data-driven insights into market trends, sales performance, and inventory turnover, thereby supporting more strategic decision-making.

From a research perspective, this study serves as a foundational model for digitalizing agricultural trading in similar contexts. It is recommended that this system be considered a pilot for potential replication and scalability in other agricultural trading centers across the Philippines. A longitudinal study is also warranted to quantitatively assess the long-term socio-economic impact of the system on farmer income, market efficiency, and food supply chain resilience. Such research would provide empirical evidence of the value of technological intervention in rural agricultural economies and inform future policy and development initiatives.

ACKNOWLEDGMENT

The researcher extends sincere gratitude to all individuals and institutions who contributed to the successful completion of this study. Deep appreciation is given to the Local Government Unit of Roxas and the Agri-Pinoy Trading Center for their invaluable support, collaboration, and for providing the necessary data and insights that guided the development of this project. Their contributions were vital in addressing the real-world challenges of local agricultural trade and in aligning the system with the practical needs of both farmers and buyers.

REFERENCES

1) Acosta, D., Alquizar, M. L., Junio, C. A., Talara, D. C., & Buladaco, M. V. M. (2020). Analysis and Design of Sales and Inventory Management System for Yochang General Merchandise. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3643181
2) Ali, M., Yulianti, Y., & Putra, E. R. (2021). Information Systems Stock Inventory in CV Kencana Motor using Laravel Frame. TEPIAN, 2(4), 136. https://doi.org/10.51967/tepian.v2i4.182
3) Baker, G. (2006). Developing an Integrated School Information System. In Kluwer Academic Publishers eBooks (p. 67). Springer Science+Business Media. https://doi.org/10.1007/0-387-24045-4_7
4) Baurley, J. W., Budiarto, A., Kacamarga, M. F., & Pardamean, B. (2018). A Web Portal for Rice Crop Improvements. International Journal of Web Portals, 10(2), 15. https://doi.org/10.4018/ijwp.2018070102
5) Hanakawa, N., Akazawa, Y., Mori, A., Maeda, T., Inoue, S., & Tsutsui, S. (2006). A Web-based integrated education system for a seamless environment among teachers, students, and administrators. Systems and Computers in Japan, 37(5), 14. https://doi.org/10.1002/scj.20475
6) Herikson, R., & Kurniati, P. S. (2019). Web-Based Ordering Information System on Food Store. IOP Conference Series Materials Science and Engineering, 662(2), 22010. https://doi.org/10.1088/1757-899x/662/2/022010
7) Kaewthong, K., Phooma, J., Tongliemnak, K., Thepraksa, N., & Sukkhieo, P. (2018). Folk wisdom using for management and information system development for marketing demonstration center (community shop). AIP Conference Proceedings, 2016, 20065. https://doi.org/10.1063/1.5055467
8) Leonardo, P., & Wiratama, J. (2023). Empowering Efficiency: A Web-Based Inventory and Sales Information System for Drinking Water Distributors through Rapid Application Development. Journal of Information Systems and Informatics, 5(2), 742. https://doi.org/10.51519/journalisi.v5i2.498

9) Madyatmadja, E. D., Hakim, A. N., & Sembiring, D. J. M. (2021). Performance testing on Transparent Data Encryption for SQL Server's reliability and efficiency. Journal Of Big Data, 8(1). https://doi.org/10.1186/s40537-021-00520-z
10) Pratama, D. A. (2021). DESIGN OF FISH SALES INFORMATION SYSTEM IN PT XYZ USING LARAVEL FRAMEWORK. Intelmatics, 1(1). https://doi.org/10.25105/itm.v1i1.7808
11) Purba, J. T., Hery, H., & Widjaja, A. E. (2020). E-commerce implementation in supporting business services strategy (case study at petshop gifaro evidence). Journal of Physics Conference Series, 1563(1), 12012. https://doi.org/10.1088/1742- 6596/1563/1/012012
12) Sluiter, I. (2008). Chapter One. General Introduction. In BRILL eBooks (p. 1). Brill. https://doi.org/10.1163/ej.9789004166240.i-516.5
13) Soegoto, E. S., & Suripto, A. (2018). Design of E-commerce Information System on Web-based Online Shopping. IOP Conference Series Materials Science and Engineering, 407, 12008. https://doi.org/10.1088/1757-899x/407/1/012008
14) Soegoto, E. S., & Wijaya, R. W. (2018). Building a PlayStation Rental (PS) Information System Web-based. IOP Conference Series Materials Science and Engineering, 407, 12017. https://doi.org/10.1088/1757-899x/407/1/012017
15) Syafariani, R. F., Defranata, D., & Hayati, E. N. (2019). Web Based Information System of Sales and Purchases on Cafe. Jurnal Manajemen Informatika (JAMIKA), 9(2), 112. https://doi.org/10.34010/jamika.v9i2.2168
16) Tsai, H.-F., Chang, B. R., Huang, H.-C., & Huang, C. (2019). Implementation of Mobile Point-of-sale Cashier Management System. Sensors and Materials, 31(8), 2463. https://doi.org/10.18494/sam.2019.2427
17) Yusof, A. M. (2004). An on-line purchasing and decision support system for distributed retail chain stores. https://doi.org/10.1145/1052220.1052244