Engineering an E-Commerce Database: Design, Implementation, and Optimization

Aravind Panchanathan
Information, Science & Technology
George Mason University
Email: apanchan@gmu.edu

***Abstract*—This paper presents the design and development of an E-Commerce Database that efficiently stores and manages online retail data. The database follows industry best practices in system design, data architecture, and governance. We discuss the engineering process, schema selection, and implementation details. Additionally, an ER diagram is included to visualize the structure. Finally, we acknowledge the assistance of ChatGPT in generating the dataset for testing and validation.**

***Index Terms*—E-Commerce, Database Engineering, System Design, Data Architecture, ER Diagram**

I. Introduction

The rapid growth of online commerce necessitates robust, scalable, and optimized databases. This paper details the engineering process behind an E-Commerce database designed to handle large-scale transactional data efficiently. Our approach ensures data integrity, performance optimization, and compliance with relational database best practices.

II. Engineering Process

The development of the E-Commerce database followed a structured approach:

1) **Requirements Analysis:** Identified core entities like users, products, orders, payments, promotions, and logistics.
2) **Schema Design:** Established a normalized schema to minimize redundancy and optimize query performance.
3) **Indexing Strategies:** Implemented indexes on frequently queried attributes to improve efficiency.
4) **Data Generation and Testing:** Used ChatGPT to generate a realistic dataset for validation.
5) **Optimization and Security:** Ensured ACID compliance, foreign key constraints, and indexing strategies.

III. Database Design and Schema

The database consists of multiple interrelated tables, structured for optimal performance and scalability. The key tables include:

- **Users:** Stores user information such as name, email, and addresses.
- **Products:** Contains product details, categories, and brand information.
- **Orders:** Manages transactions, linking users to purchased products.
- **Payments:** Tracks payment methods, statuses, and transaction dates.
- **Promotions:** Stores details of discount offers and coupons to enhance customer engagement.
- **Logistics:** Handles shipments, tracking, and delivery statuses.
- **Reviews and Analytics:** Stores user feedback, search history, and page views.

IV. Entity-Relationship Diagram

Figure 1 shows the ER diagram representing the relationships among the tables.

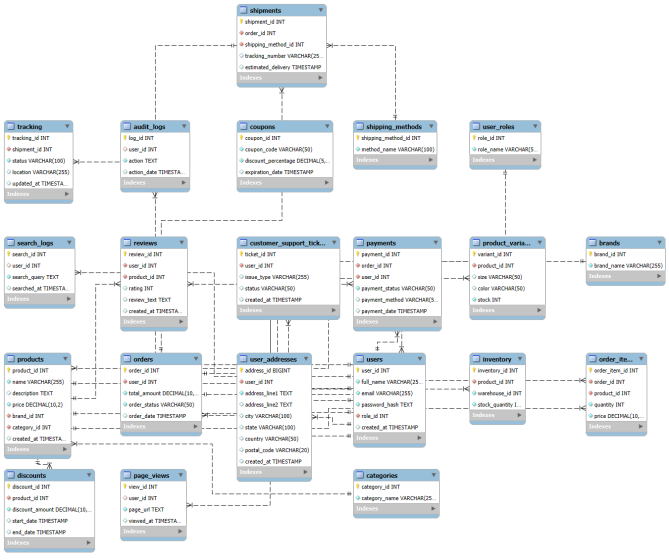

Fig. 1. Entity-Relationship (ER) Diagram of the E-Commerce Database.

V. Principles of Database Design and Optimization

The database schema was designed following essential **database design principles** to ensure efficiency and scalability:

- **Normalization:** The schema follows third normal form (3NF) to eliminate redundancy while maintaining referential integrity.
- **Indexing:** Frequently queried columns (e.g., user emails, product names, and order dates) are indexed to improve retrieval speed.

- **Foreign Key Constraints:** Enforced relationships between tables to maintain data consistency and prevent orphan records.
- **Partitioning:** Order data is partitioned by date range to optimize query performance for historical transactions.
- **ACID Compliance:** Ensured atomicity, consistency, isolation, and durability for transaction reliability.
- **Efficient Query Execution:** Optimized SQL queries using JOINs, indexing, and caching to handle large-scale queries.

VI. Conclusion and Future Work

This paper detailed the design and engineering of an E-Commerce database. Future work will involve performance benchmarking, query optimization, and integrating NoSQL elements for scalability.

Acknowledgment

We acknowledge the assistance of ChatGPT in generating realistic datasets for database validation and testing, which significantly streamlined the development process.

References