Name: Mohammed Abdul-Jalil

Role: Stackholder

Product: Chocolate

1. System Architecture Review

The system architecture aligns with best practices in database normalization, referential integrity, and modular design. The Entity-Relationship (ER) diagram effectively represents a marketplace system by structuring relationships between users, products, orders, payments, and reviews. Key considerations include:

Role-Based Access Control (RBAC): The User entity includes roles (customer, seller, admin), ensuring role-based functionality.

Transactional Integrity: The Order-Payment relationship supports payment tracking, ensuring financial accountability.

Scalability: The system can handle a large number of orders and users with minimal redundancy.

2. ER Diagram Review:

Strengths:

Entity Normalization: The schema follows third normal form (3NF), minimizing redundancy and optimizing storage.

Referential Integrity Enforced: Proper use of primary and foreign keys ensures consistency between related entities.

Comprehensive Order Management: The Order entity includes status tracking (pending, shipped, delivered, canceled), which aligns with e-commerce workflows.

Review System Enhances User Engagement: Customers can provide feedback on products, increasing credibility.

Areas for Improvement:

1. Missing Relationship Between OrderItem and User

The schema currently tracks the user at the order level, but individual OrderItem records should also have direct buyer tracking for analytics and accountability.

Solution: Add a UserId foreign key to OrderItem.

2. Review Table Needs a Unique Constraint

A single user should not be able to submit multiple reviews for the same product.

Solution (Database Consistency Rule): Enforce a unique constraint on (UserId, ProductId) to prevent duplicate reviews.

3. Payment-User Relationship Is Missing

Currently, payments are linked only to orders, not to users.

Solution (Financial Accountability Principle): Add a UserId foreign key in Payment to track who made the payment.

4. Order Status History Should Be Tracked

The Order entity overwrites status updates instead of keeping a history of changes.

Solution (Auditing & Compliance): Introduce an OrderStatusHistory table that logs status changes with timestamps.

3. Alignment with OPIT Business Goals Framework

According to the OPIT business and technology module, an effective database design must:

Support business processes efficiently ʼn The schema models key marketplace functions (orders, payments, reviews).

Ensure data integrity and security ʼn The use of foreign keys, constraints, and role-based access strengthens integrity.

Enhance user experience and engagement ʼn Review and order tracking improve customer trust and seller accountability.

Allow for scalability and future enhancements ʼn The modular structure supports future business expansion (e.g., adding promotions or customer service interactions).

Final Recommendation

The current ER diagram provides a solid foundation, but incorporating the suggested improvements will:

Enhance data tracking and analytics.

Improve business compliance with financial and review constraints.

Ensure scalability for future marketplace growth.