

Database Schema Analysis: Design Decisions & Assumptions

Executive Summary

The provided e-commerce database schema demonstrates a well-structured relational design that effectively balances normalization principles with practical performance considerations. The schema is designed to achieve primarily 3NF (Third Normal Form) to optimize query performance for common e-commerce operations.

Detailed Design Decisions and Justifications

1. User Management Architecture:

- Design Decision: Inheritance Hierarchy (*User* → *Customer/Admin*)
 - *User* (base table) → *Customer/Admin* (specialized tables)
- Justification:
 - Flexibility: Allows different user types with specific attributes
 - Extensibility: Easy to add new user types (Vendor, Staff, etc.)
 - Data Integrity: Common user attributes centralized
 - Performance: Avoids NULL values for type-specific fields
 - Alternative Considered: Single User table with nullable specialized columns.
 - Why Rejected: Would violate normalization and create sparse tables.

2. Product Catalogue Design:

- Design Decision: Product-Variant Relationship
 - *Product* (1) → (M) *ProductVariant*
- Justification:
 - Flexibility: Handles size, color, capacity variations efficiently
 - Inventory Management: Separate stock tracking for variants
 - Pricing Strategy: Supports variant-specific pricing (*additional_price*)
 - SKU Management: Each variant has unique identifier

- Key Design Features:
 - Base product contains common attributes
 - Variants store differential information
 - Optional variants (some products may not have variants)
- Category Assumptions:
 - Design Decision: Category Hierarchy (Single Level)
 - Category (simple table without *parent_id*)
 - Assumption:
 - Single-level categorization is sufficient
 - Justification:
 - Simplifies queries and maintenance
 - Limitation: Cannot handle subcategories
 - (Electronics → Mobile Phones → Smartphones)
 - Future Enhancement: Could add *parent_category_id* for hierarchical structure

3. Order Management System:

- Design Decision: Comprehensive Order Lifecycle
 - *Order* → *OrderItem* → *Payment* → *Shipment* → *Returns*
- Justification:
 - Auditability: Complete transaction history
 - State Management: Clear order status progression
 - Financial Tracking: Separate payment processing
 - Logistics: Independent shipping management
- Design Decision: Address Handling Strategy
- Approach: Dual reference system
 - Foreign keys to Address table (*shipping_address_id*, *billing_address_id*)
 - Denormalized address fields in *Order* table
 - Justification:
 - Data Integrity: Maintains current address references
 - Historical Accuracy: Preserves address at time of order
 - Performance: Reduces joins for order display
 - Legal Compliance: Important for shipping and tax purposes

4. Shopping Cart Architecture:

- Design Decision: Persistent Cart with Session Management
 - *Cart (1) → (M) CartItem*
- Justification:
 - User Experience: Cart persists across sessions
 - Multi-device Access: Cart accessible from different devices
 - Analytics: Track cart abandonment patterns
 - Performance: Batch operations on cart items
 - Key Features:
 - Timestamp tracking (*created_at, updated_at*)
 - Supports product variants in cart
 - Customer-specific carts

5. Supplier and Inventory Management:

- Design Decision: Separate Supplier Entity with Junction Table
 - *Supplier (M) ↔ (M) Product* via *product_supplier*
- Justification:
 - Multi-sourcing: Products can have multiple suppliers
 - Vendor Management: Centralized supplier information
 - Cost Tracking: Supplier-specific pricing
 - Supply Chain: Lead time and relationship management
 - Business Benefits:
 - Enables supplier comparison
 - Supports drop-shipping models
 - Facilitates vendor performance tracking

6. Customer Engagement Features:

- Design Decision: Separate Wishlist and Review Entities
 - *Customer (1) → (M) Wishlist*
 - *Customer (1) → (M) Review*
- Justification:
 - User Engagement: Encourages return visits
 - Social Proof: Reviews influence purchasing decisions
 - Analytics: Track customer preferences and satisfaction
 - Marketing: Wishlist abandonment campaigns

Assumptions Made in Schema Design

1. Business Logic Assumptions:

- Currency Assumptions:
 - Single Currency: All prices in Pakistani Rupees (PKR)
 - Implication: No multi-currency support
 - Future Enhancement: Add currency table and conversion rates
- Tax Handling:
 - Simplified tax structure
 - Current: Single *tax_amount* field in orders
 - Assumption: Uniform tax rates across products/regions
 - Limitation: Cannot handle complex tax scenarios
- Inventory Model:
 - Simple stock quantity tracking
 - Assumption: First-in-first-out inventory management
 - Limitation: No lot tracking or expiration date management
 - Geographic Scope: Pakistan-focused
 - Evidence: Pakistani phone number formats, addresses
 - Assumption: Domestic market only
 - Implication: Address and shipping models may need adjustment for international expansion

2. Strengths of the Current Design:

I. Data Integrity:

- Comprehensive foreign key relationships
- Proper entity separation
- Consistent naming conventions

II. Business Logic Support:

- Complete e-commerce workflow coverage
- Flexible product variant system
- Comprehensive order lifecycle management
- Extensibility
- Modular design allows easy feature additions
- User type inheritance supports new roles

- Discount system can handle various promotion types

III. Performance Considerations:

- Strategic denormalization where beneficial
- Efficient query patterns for common operations
- Proper indexing opportunities (primary/foreign keys)

IV. Expectations for Potential Enhancement

- Advanced Features:
 - Multi-level Categories: Hierarchical product categorization
 - Inventory Reservations: Handle stock allocation during checkout process
 - Advanced Pricing: Time-based pricing, bulk discounts, customer-specific pricing
- International Expansion:
 - Multi-currency Support: Currency conversion and management
 - Localization: Multi-language product descriptions
 - Regional Tax Systems: Complex tax calculation support

Conclusion

The e-commerce database schema demonstrates a solid understanding of relational database design principles with practical business considerations. The design successfully balances normalization benefits with performance requirements, creating a robust foundation for e-commerce operations.

The schema effectively supports the core e-commerce workflows while providing flexibility for future growth and feature additions. The assumptions made are reasonable for the target market and scale, with clear paths for enhancement as business requirements evolve.