

E-Commerce System – Requirements

Project Overview

This web-based application will include a backend database for an e-commerce system that supports two user roles: Customers and Staff. Customers can browse products, place multi-item orders, and view order history. Staff can authenticate, add/edit products, manage stock levels, and monitor orders.

Users & Roles

1. Customer – view products, places orders with single/multiple items, can view past orders
2. Staff – should have authentication, adds/edits products, orders more product, can view orders

Data Requirements

1. Customer – customer_id (PK), name, email (unique), created_at
2. Staff – staff_id (PK), name, email (unique), role, created_at
3. Product – product_id (PK), name, price, stock_qty, active flag
4. Order – order_id (PK), customer_id (FK→Customer), order_date, status
5. OrderItem – (order_id, product_id) composite PK, quantity, unit_price, FKs to Order and Product

Use Cases

1. Staff log in and creates a new product with stock amount
2. Staff updates stock quantity for an existing product
3. Customer places an order containing single/multiple products
4. Customer views order history
5. Staff can view recent orders and their status

Constraints & Assumptions

1. Stock quantity cannot be negative
2. Order must check available stock
3. An order must have at least one item
4. Status values of orders should include pending, paid, shipped, canceled
5. Unique IDs for customers, staff, products, orders

Notes on Authentication (Optional for CLI Scope)

If implementing authentication in the CLI/business logic, store Staff credentials (e.g., hashed password) in the Staff table or a related auth table. For the minimum scope, you may simulate authentication in the CLI and still record `staff_id` on product changes for traceability