Bài tập ứng dụng

Website CoolStore

- Product Catalog

- Shopping Cart

- Payment Process

- Inventory

- Rating

- Access Control

Câu 1: Hãy liệt kê các yêu cầu chức năng và phi chức năng của hệ thống CoolStore dựa trên mô tả kịch bản nghiệp vụ

Yêu cầu chức năng:

- Product Catalog: Xem danh mục các sản phẩm, lọc và sắp xếp theo tên/giá, phân trang

- Product Detail: Xem tên, mô tả, số lượng tồn kho, thông tin kho, trạng thái “Sản phẩm nổi bật” và đánh giá

- Admin product management: SysAdmin CRUD sản phẩm, gán sản phẩm vào kho hiện có

- Shopping Cart:

+ Thêm sản phẩm từ trang danh mục sản phẩm hoặc chi tiết sản phẩm

+ Xem bảng tổng hợp với thông tin: Tổng chi phí giỏ hàng, tổng chi phí sau khuyến mãi, chi phí vận chuyển, khuyến mãi tiết kiệm, tổng giá trị đơn hàng. (in realtime)

+ Thêm/Xoá sản phẩm khỏi giỏ hàng

- Checkout: Hệ thống xác thực thông tin sản phẩm, xử lý thanh toán, và gửi email cho người mua

- Rating: Buyer có thể gửi đánh giá (1–5 sao) cho sản phẩm.

- User Auth: Đăng ký/Đăng nhập/Đăng xuất

- Inventory: Quản lý kho hàng

- Manage user’s carts: SysAdmin có thể xem bất kỳ giỏ hàng của bất kỳ người dùng nào, kích hoạt/vô hiệu hoá bất kỳ giỏ hàng nào không hợp lệ

Yêu cầu phi chức năng:

- Initial Setup: Khi khởi động cần tạo một tài khoản admin và 2 người mua(Buyer), sản phẩm mẫu, kho và đánh giá cho các sản phẩm khởi tạo

Câu 4: Mô tả chi tiết một Use Case "Mua sản phẩm từ trang chi tiết sản phẩm". Trình bày theo mẫu chuẩn gồm: Tên use case, Tác nhân chính, Mục tiêu, Tiền điều kiện, Luồng chính, Luồng thay thế, Hậu điều kiện, Ghi chú (nếu có).

|  |  |  |
| --- | --- | --- |
|  | | |
| Use Case Number | UC1 | |
| Use Case Name | Mua sản phẩm từ trang chi tiết sản phẩm | |
| Actors | Buyer | |
| Maturity | Focused | |
| Summary | Tính năng này cho phép người mua có thể mua sản phẩm từ trang chi tiết sản phẩm | |
| Basic Course of Events | Actor Action | System Response |
| 1. Thực hiện xác thực đăng nhập |  |
|  | 2. Hệ thống hiển thị danh mục các sản phẩm của CoolStore Actor |
| 3. Use case bắt đầu khi Actor chọn một sản phẩm bất kỳ của cửa hàng |  |
|  | 4. Hệ thống hiển thị chi tiết sản phẩm mà Actor chọn bao gồm tên, mô tả, số lượng tồn kho, thông tin kho, trạng thái “Sản phẩm nổi bật” và đánh giá |
| 5. Actor chọn số lượng cần mua và chọn thêm vào giỏ hàng. A1. E1 |  |
|  | 6. Hệ thống thêm sản phẩm vào giỏ hàng và tự động tính tổng số tiền của giỏ hàng đó bao gồm subtotal |
| Alternative Paths | **A1.** Kiểm tra xem sản phẩm có đủ để cho vào giỏ hàng không (Nếu quantity > stock on hand) thì hệ thống sẽ báo lỗi và thông báo cho người dùng đã hết hàng. Quay lại bước 4 của Basic Course of Events. | |
| Exception Paths | **E1**. Hệ thống kiểm tra xem số lượng mà người dùng nhập vào thêm vào phải là số dương hay không | |
| Extension Points | Không có | |
| Triggers | Nhân viên mua hàng thực hiện quản lý xe | |
| Assumption | Đã có database của sản phẩm và sản phẩm chi tiết.  Người dùng đã có giỏ hàng lúc tạo tài khoản | |
| Pre-conditions | Buyer đã đăng nhập vào hệ thống | |
| Post Conditions | Nếu thành công thì sẽ đưa người dùng đến trang checkout để tiến hành thanh toán giỏ hàng.  Nếu thất bại thì giỏ hàng sẽ xoá sản phẩm khỏi cart | |
| Reference: Bunsiness Rules | **Không có** | |
| Author(s) | Hiếu | |
| Date | 15-09-2025 | |

Câu 8: Đề xuất cách tổ chức kiến trúc phần mềm cho hệ thống CoolStore theo mô hình kiến trúc ba lớp (3-tierarchitecture).

**Layered 3-tier: Presentation (UI) → Application (Services) → Data (DB)**

* **Presentation Layer (Client)**
  + React web app / mobile app.
  + Responsibilities: render catalog, cart UI, validate forms, manage session, call APIs.
* **Application Layer (Backend API)**
  + RESTful or gRPC microservices (or modular monolith):
    - Catalog Service (product search/filter, product details, ratings).
    - Cart Service (in-memory or persisted cart + merging, promotions).
    - Checkout/Order Service (orchestrates inventory + payment + order creation).
    - Inventory Service (manages stock per warehouse).
    - User/Identity Service (authz/authn).
    - Admin Service (product & warehouse management).
    - Notification Service (send email).
  + Use API Gateway for routing.
  + Business logic, transactions, integration with PaymentGateway & WarehouseSystem.
  + Use caching (Redis) for catalog hot data; use DB for persistent entities.
* **Data Layer**
  + Relational DB for orders, carts (persisted), product metadata (MySQL/Postgres).
  + NoSQL for product search/filters (Elasticsearch).
  + Cache (Redis) for sessions & cart partial storage.
  + Message queue (Kafka/RabbitMQ) for async tasks (send email, inventory updates).
* **Notes for consistency**
  + Inventory & checkout need strong consistency: use transactions or SAGA pattern with compensation.
  + For availability and scale: separate read models (CQRS) for catalog listing.

Câu 9: Viết ít nhất 3 ca kiểm thử (test cases) cho chức năng “Thêm sản phẩm vào giỏ hàng” – bao gồm dữ liệu đầu vào, bước thực hiện và kết quả mong đợi.

Mỗi test case gồm: ID, Mục tiêu, Dữ liệu đầu vào, Các bước, Kết quả mong đợi.

**TC1 – Thêm 1 sản phẩm có đủ tồn kho vào giỏ (happy path)**

* ID: TC\_ADD\_001
* Precondition: Product P123 có stock >= 3; user đã login.
* Test data: productId = P123; qty = 2; userId = U100
* Steps:
  1. User login U100.
  2. Truy cập product detail P123.
  3. Chọn qty = 2, click “Buy” (Add to Cart).
  4. Mở Cart.
* Expected:
  1. Hệ thống trả về HTTP 200 cho add action.
  2. Cart chứa item P123 với quantity = 2.
  3. Subtotal tăng tương ứng: unitPrice \* 2.
  4. No error messages; stock remains (availableStock decreased only at checkout or reserved depending design).

**TC2 – Thêm sản phẩm với quantity vượt quá stock**

* ID: TC\_ADD\_002
* Precondition: Product P200 có stock = 1; user not necessarily logged in (guest allowed).
* Test data: productId = P200; qty = 3
* Steps:
  1. Mở product P200 detail.
  2. Enter qty = 3, click “Buy”.
* Expected:
  1. Hệ thống trả về HTTP 4xx hoặc app-level error.
  2. Hiển thị message: “Requested quantity exceeds available stock (only 1 left).”
  3. Cart không chứa entry với qty > 1; either allow adding with max available qty =1 or reject add (depending business rule).
  4. No negative stock occurs.

**TC3 – Thêm cùng một sản phẩm 2 lần (increment quantity)**

* ID: TC\_ADD\_003
* Precondition: Product P300 stock >= 5; user logged in as U200; cart currently empty.
* Test data: productId = P300.
* Steps:
  1. From product detail, click “Buy” (qty default 1) → adds to cart.
  2. Navigate back to product detail P300, click “Buy” again (qty 1).
  3. Open Cart.
* Expected:
  1. After first add: cart contains P300 qty =1.
  2. After second add: cart contains P300 qty =2 (item merged, not duplicated entry).
  3. Subtotal updated correctly (2 \* unitPrice).