**ĐẠI HỌC PHENIKAA**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN PHENIKAA**

**A logo for a university

AI-generated content may be incorrect.**

**MÔN HỌC: PHÂN TÍCH VÀ THIẾT KẾ PHẦN MỀM**

**ĐỀ TÀI: “HỆ THỐNG QUẢN LÝ CỬA HÀNG THIẾT BỊ ĐIỆN TỬ TRỰC TUYẾN”**

**Thành viên nhóm**

|  |  |  |
| --- | --- | --- |
| **Nguyễn Văn Thăng** | **23010572** | **K17-KHMT(AI&KHDL)\_1** |
| **Phạm Văn Sự** | **23010523** | **K17-KHMT(AI&KHDL)\_1** |
| **Đặng Anh Tuyền** | **23010912** | **K17-KHMT(AI&KHDL)\_1** |
| **Trần Đình Dũng** | **23010596** | **K17-KHMT(AI&KHDL)\_1** |
| **Nguyễn Huy Toàn** | **23017052** | **K17-KHMT(AI&KHDL)\_1** |

**NHÓM 16**

**GIÁO VIÊN GIẢNG DẠY:**

**ThS VŨ QUANG DŨNG & ThS NGUYỄN XUÂN QUẾ**

**ĐẠI HỌC PHENIKAA**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN PHENIKAA**

A logo for a university

AI-generated content may be incorrect.

**MÔN HỌC: PHÂN TÍCH VÀ THIẾT KẾ PHẦN MỀM**

**ĐỀ TÀI: “HỆ THỐNG QUẢN LÝ CỬA HÀNG THIẾT BỊ ĐIỆN TỬ TRỰC TUYẾN”**

**Thành viên nhóm**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Họ và tên** | **Mã sinh viên** | **Điểm bằng số** | **Điểm bằng chữ** | **Đóng góp** |
| **Nguyễn Văn Thăng** | **23010572** |  |  |  |
| **Phạm Văn Sự** | **23010523** |  |  |  |
| **Đặng Anh Tuyền** | **23010912** |  |  |  |
| **Trần Đình Dũng** | **23010596** |  |  |  |
| **Nguyễn Huy Toàn** | **23017052** |  |  |  |

|  |  |
| --- | --- |
| **Chữ ký giám thị 1** | **Chữ ký giám thị 2** |
|  |  |

Mục lục

[**I: GIỚI THIỆU VÀ LẬP KẾ HOẠCH** 2](#_Toc210239968)

[**1.1. Giới thiệu** 2](#_Toc210239969)

[**1.1.1. Giới thiệu đề tài** 2](#_Toc210239970)

[**1.1.2. Mục Tiêu dự án** 2](#_Toc210239971)

[**1.1.3. Phạm vi triển khai** 3](#_Toc210239972)

[**1.2. Mô hình quy trình** 4](#_Toc210239973)

[**II: PHÂN TÍCH YÊU CẦU** 5](#_Toc210239974)

[**2.1. Yêu cầu chức năng** 5](#_Toc210239975)

[**2.2.1. Khách hàng** 5](#_Toc210239976)

[**2.2.2. Quản trị viên** 6](#_Toc210239977)

[**2.2.3. Cổng thanh toán** 7](#_Toc210239978)

[**2.2. Yêu cầu phi chức năng** 8](#_Toc210239979)

[**2.2.1. Hiệu suất** 8](#_Toc210239980)

[**2.2.2. Khả năng mở rộng** 8](#_Toc210239981)

[**2.2.3.** **Tương thích & Di động** 9](#_Toc210239982)

[**2.2.4. Độ tin cậy & Khả dụng** 9](#_Toc210239983)

[**2.2.5. Bảo trì** 10](#_Toc210239984)

[**2.2.6. Bảo mật & Tuân thủ** 11](#_Toc210239985)

[**2.2.7. Khả năng sử dụng** 11](#_Toc210239986)

[**2.2.8. Quan sát & Vận hành** 12](#_Toc210239987)

[**2.3. Mô hình hóa** 14](#_Toc210239988)

[**2.4. Kịch bản** 14](#_Toc210239989)

[**2.4.1. Trường hợp sử dụng: Đặt hàng** 14](#_Toc210239990)

[**2.4.1.1. Kịch bản phụ: Hết hàng đột ngột ngay thời điểm xác nhận** 15](#_Toc210239991)

[**2.4.2. Trường hợp sử dụng: Đặt hàng** 15](#_Toc210239992)

[**2.4.2.1. Kịch bản phụ: Thẻ bị từ chối do không đủ số dư** 16](#_Toc210239993)

[**2.4.3. Trường hợp sử dụng: Hủy đơn hàng & Hoàn tiền** 16](#_Toc210239994)

[**2.4.3.1. Kịch bản phụ 3.1 — Chặn hủy khi đã xuất kho** 16](#_Toc210239995)

[**III: THIẾT KẾ HỆ THỐNG** 17](#_Toc210239996)

[**3.1. Thiết kế kiến trúc tổng thể** 17](#_Toc210239997)

[**3.1.1. Lựa chọn mô hình kiến trúc** 17](#_Toc210239998)

[**3.2. Biểu đồ Gói** 18](#_Toc210239999)

[**3.2.1. Các tầng và mục đích** 18](#_Toc210240000)

[**3.2.2. Lớp giao diện** 19](#_Toc210240001)

[**3.2.3. Lớp nghiệp vụ** 19](#_Toc210240002)

[**3.2.4. Lớp truy cập dữ liệu** 20](#_Toc210240003)

[**3.2.5. Lớp đối tượng nghiệp vụ** 20](#_Toc210240004)

[**3.2.6. Luồng phụ thuộc chính** 20](#_Toc210240005)

[**3.2.7. Luồng ánh xạ dữ liệu** 20](#_Toc210240006)

[**3.3. Biểu đồ dữ liệu** 22](#_Toc210240007)

[**3.4. Mô hình luồng dữ liệu** 22](#_Toc210240008)

[**3.4.1. Cấp 0:** 22](#_Toc210240009)

[**3.4.2. Cấp 1:** 22](#_Toc210240010)

[**3.2. Thiết kế chi tiết (UML)** 24](#_Toc210240011)

[**3.2.1. Sơ đồ lớp (Class Diagram)** 24](#_Toc210240012)

[**3.2.1.1. Nhóm Người dùng & Module quản trị** 24](#_Toc210240013)

[**3.2.1.2. Nhóm Catalog (Product, Category) & Inventory** 25](#_Toc210240014)

[**3.2.1.3. Nhóm Cart** 25](#_Toc210240015)

[**3.2.1.4. Nhóm Order & Discount** 25](#_Toc210240016)

[**3.2.1.5. Nhóm Payment** 26](#_Toc210240017)

[**3.2.1.6. Nhóm CSKH, Review & Notification** 26](#_Toc210240018)

[**3.2.3. Sơ đồ trình tự (Sequence Diagram)** 27](#_Toc210240019)

[**3.2.4. Biểu đồ hoạt động** 27](#_Toc210240020)

[**3.2.5. Biểu đồ trạng thái** 27](#_Toc210240021)

[**3.2.5.1. Biểu đồ trạng thái – Đơn hàng** 27](#_Toc210240022)

[**3.2.5.2. Biểu đồ trạng thái – Thanh toán** 29](#_Toc210240023)

[**3.2.5.3. Biểu đồ trạng thái – Tồn kho/Sản phẩm** 30](#_Toc210240024)

[**3.2.5.4. Biểu đồ trạng thái – Tài khoản người dùng** 31](#_Toc210240025)

[**IV: TRIỂN KHAI & KIỂM THỬ** 32](#_Toc210240026)

[**4.1. Công nghệ sử dụng** 32](#_Toc210240027)

[**4.2. Kiểm thử** 32](#_Toc210240028)

[**V: TỔNG KẾT VÀ HƯỚNG PHÁT TRIỂN** 32](#_Toc210240029)

[**5.1. Đánh giá** 32](#_Toc210240030)

[**5.2. Hướng phát triển** 32](#_Toc210240031)

**Lời mở đầu**

Sự phát triển mạnh mẽ của thương mại điện tử đang định hình lại cách doanh nghiệp tiếp cận khách hàng và vận hành bán lẻ, đặc biệt trong ngành thiết bị điện tử với danh mục rộng, nhiều biến thể sản phẩm (màu sắc, dung lượng, cấu hình), chu kỳ đổi mẫu ngắn và nhu cầu hậu mãi cao. Từ thực tế đó, nhóm thực hiện đề tài “Hệ thống Quản lý Cửa hàng Thiết Bị Điện Tử Trực Tuyến” nhằm xây dựng một nền tảng hỗ trợ trọn vẹn quy trình bán hàng: quản lý sản phẩm và tồn kho, giỏ hàng và đơn hàng, thanh toán trực tuyến, khuyến mãi và voucher, theo dõi vận chuyển và chăm sóc khách hàng, đồng thời đáp ứng các yêu cầu về hiệu năng, bảo mật và trải nghiệm người dùng.

Trong khuôn khổ học phần Phân tích & Thiết kế Phần mềm, dự án là cơ hội để nhóm em áp dụng một quy trình phát triển hoàn chỉnh: bắt đầu từ khai thác và làm rõ yêu cầu, tiếp tục mô hình hóa nghiệp vụ, tiến hành thiết kế kiến trúc và dữ liệu, triển khai hiện thực các chức năng và kiểm thử để bảo đảm chất lượng trước khi bàn giao. Báo cáo trình bày mục tiêu, phạm vi và kiến trúc hệ thống, các mô-đun chức năng trọng tâm, những tiêu chí phi chức năng (hiệu năng, an toàn thông tin, khả dụng), cùng thách thức kỹ thuật và định hướng mở rộng như tối ưu tìm kiếm, gợi ý sản phẩm và báo cáo phân tích. Kỳ vọng của nhóm em là sản phẩm vừa giải quyết bài toán quản trị vận hành cho cửa hàng điện tử trực tuyến, vừa mang lại trải nghiệm mua sắm tin cậy, thuận tiện và nhất quán cho người dùng.

# **I: GIỚI THIỆU VÀ LẬP KẾ HOẠCH**

## **1.1. Giới thiệu**

### **1.1.1. Giới thiệu đề tài**

Đề tài xây dựng Hệ thống Quản lý Cửa hàng Thiết bị Điện tử trực tuyến, giúp khách hàng tìm kiếm, đặt mua, thanh toán an toàn và theo dõi đơn hàng mạch lạc. Ở phía vận hành, hệ thống cung cấp nền tảng quản trị tập trung để quản lý sản phẩm, tồn kho, đơn hàng, khuyến mãi và chăm sóc khách hàng. Ba tác nhân chính gồm Khách hàng, Quản trị viên và Cổng thanh toán tích hợp, phối hợp để bảo đảm trải nghiệm liền mạch, vận hành minh bạch và khả năng mở rộng ổn định.

**1.1.2. Mục Tiêu dự án**

Trải nghiệm khách hàng mượt mà và rõ ràng. Đăng ký hoặc đăng nhập => tìm kiếm, lọc sản phẩm => xem chi tiết với hình ảnh, mô tả, tồn kho, đánh giá => quản lý giỏ hàng => đặt mua => thanh toán => theo dõi trạng thái đơn => xem lịch sử giao dịch. Sau mua => để lại đánh giá => nhận hỗ trợ trực tuyến trong tài khoản.

Phía vận hành dành cho quản trị viên. Quản lý người dùng, danh mục, sản phẩm, tồn kho. Điều phối quy trình đơn từ xác nhận đến giao nhận. Thực hiện thanh toán và hoàn tiền. Thiết lập khuyến mãi và triển khai chiến dịch marketing. Theo dõi báo cáo và thống kê theo thời gian thực để ra quyết định nhanh và chính xác.

Thanh toán an toàn với tích hợp cổng thanh toán, xác thực giao dịch bằng OTP và 3DS, mã hóa dữ liệu khi truyền và khi lưu trữ, hỗ trợ hoàn tiền và đối soát minh bạch, giảm rủi ro gian lận, bảo vệ thông tin nhạy cảm của khách hàng.

Mục tiêu phi chức năng được lượng hóa rõ ràng. Thời gian tải trang < 5 giây khi số người dùng đồng thời < 1000. Thông lượng xử lý ≥ 500 đơn/phút. Bảo mật đáp ứng TLS ≥ 1.2 và mã hóa dữ liệu khi lưu trữ. Hạ tầng mở rộng để tăng gấp đôi quy mô trong 6 tháng. Người dùng mới hoàn tất đặt hàng trong < 2 phút. Những chuẩn này giúp hệ thống nhanh, an toàn, dễ mở rộng.

### **1.1.3. Phạm vi triển khai**

- **Đối tượng sử dụng**: Khách hàng (end-user), Quản trị viên/nhân sự vận hành, và Cổng thanh toán bên thứ ba.

**- Chức năng trong phạm vi**:

+ Khách hàng: đăng ký/đăng nhập, tìm–lọc, xem chi tiết, giỏ hàng, đặt hàng, thanh toán, theo dõi & lịch sử đơn, đánh giá, yêu cầu hỗ trợ, nhận thông báo.

+ Quản trị viên: quản lý tài khoản, sản phẩm & tồn kho (thêm/sửa/xóa, cảnh báo hết hàng), quy trình đơn (xác nhận–đóng gói–giao–hủy/hoàn), CSKH (xử lý ticket/đánh giá), thanh toán/hoàn tiền, khuyến mãi/voucher, báo cáo–thống kê doanh thu/tồn kho/tỷ lệ đơn.

+ Cổng thanh toán: nhận yêu cầu, xác thực giao dịch, trả kết quả, hỗ trợ hoàn tiền, ghi log & báo cáo.

**- Giới hạn/ngoài phạm vi**: Tập trung mô hình **một cửa hàng trực tuyến**; chưa bao gồm quản lý đa chi nhánh/ERP phức tạp (có thể mở rộng về sau)

**Đối tượng sử dụng**: Khách hàng (end user). Quản trị viên và nhân sự vận hành. Cổng thanh toán bên thứ ba.

**Chức năng trong phạm vi**:

* **Khách hàng**: Đăng ký, đăng nhập. Cập nhật hồ sơ, địa chỉ, phương thức liên hệ. Tìm và lọc theo từ khóa, danh mục, thương hiệu, khoảng giá, còn hàng. Xem chi tiết với ảnh, mô tả, thông số, tồn kho theo kho, thời gian giao dự kiến. Quản lý giỏ hàng với thêm, bớt, đổi số lượng, áp mã giảm giá hợp lệ. Đặt hàng với địa chỉ, vận chuyển, phí hiển thị trước thanh toán. Thanh toán qua cổng, xác thực OTP hoặc 3DS, nhận kết quả. Theo dõi trạng thái đơn theo thời gian thực, xem lịch sử đơn. Đánh giá sản phẩm sau Delivered, một đơn một đánh giá cho mỗi sản phẩm. Yêu cầu hỗ trợ, tạo ticket, theo dõi phản hồi. Nhận thông báo qua email, sms, đẩy ứng dụng.  
  Luồng điển hình: Đăng ký hoặc đăng nhập => tìm, lọc => xem chi tiết => thêm vào giỏ => đặt hàng => thanh toán => theo dõi => đánh giá.
* **Quản trị viên**: Quản lý tài khoản người dùng với khóa, mở, đặt lại mật khẩu. Quản lý sản phẩm và tồn kho với thêm, sửa, xóa, ảnh, thuộc tính, cảnh báo sắp hết hàng, giữ hàng khi có đơn. Quy trình đơn với tiếp nhận, xác nhận, đóng gói, bàn giao vận chuyển, theo dõi giao thành công, hủy hoặc hoàn khi cần. CSKH với xử lý ticket, phân loại, gán người phụ trách, quản lý phản hồi và đánh giá. Thanh toán và hoàn tiền với tra cứu giao dịch, tạo yêu cầu hoàn, theo dõi trạng thái hoàn. Khuyến mãi và voucher với tạo, lên lịch, điều kiện áp dụng, giới hạn theo nhóm khách. Báo cáo và thống kê doanh thu, tồn kho, tỷ lệ đơn, tỷ lệ hủy, hiệu năng thanh toán. Phân quyền theo vai trò kho, chăm sóc, kế toán, quản trị hệ thống.
* **Cổng thanh toán**: Nhận yêu cầu tạo giao dịch, xác thực OTP hoặc 3DS, trả kết quả thành công hoặc thất bại, gọi webhook về hệ thống. Hỗ trợ hoàn tiền toàn phần hoặc một phần. Ghi log chi tiết, cung cấp báo cáo đối soát định kỳ.

**Giới hạn hoặc ngoài phạm vi**: Tập trung một cửa hàng trực tuyến. Chưa bao gồm đa chi nhánh, đồng bộ kho đa điểm, tích hợp ERP phức tạp. Chưa bao gồm chương trình thành viên nâng cao như tier, tích điểm liên kết đối tác. Có thể mở rộng về sau theo nhu cầu.

**Tiêu chí và ràng buộc chính**: Thời gian tải trang chủ và danh mục < 5 giây với người dùng đồng thời < 1000. Thông lượng xử lý ≥ 500 đơn/phút trong giờ cao điểm. Bảo mật với HTTPS TLS ≥ 1.2, khuyến nghị 1.3. Mã hóa dữ liệu nhạy cảm khi lưu trữ. Quy trình đặt hàng cho người dùng mới < 2 phút. Độ trễ cập nhật trạng thái đơn từ webhook vận chuyển < 5 giây. Tỷ lệ thất bại thanh toán trên tổng giao dịch < 2 phần trăm.

## **1.2. Mô hình quy trình**

Nhóm lựa chọn Agile – Scrum làm mô hình phát triển vì bảo đảm lặp–tăng trưởng (iterative & incremental), bàn giao sớm, và thích ứng nhanh với thay đổi.

Tóm lược cách vận hành: Dự án chia thành Sprint 1–2 tuần; cuối mỗi Sprint tạo Increment chạy được để demo/nhận phản hồi. Công việc quản lý qua Product Backlog → Sprint Backlog, theo dõi tiến độ bằng Daily Scrum và biểu đồ burn-down; chất lượng kiểm soát bằng Definition of Ready (DoR) và Definition of Done (DoD).

Lý do lựa chọn (phù hợp đặc thù dự án & nhóm 3 thành viên):

* Tối ưu cho nhóm nhỏ: Scrum khuyến khích giao tiếp trực tiếp; Daily ≤15’ giúp mọi người nắm rõ tiến độ, vướng mắc và phối hợp nhanh.
* Thích ứng thay đổi yêu cầu: Bài tập lớn thường biến động theo phản hồi giảng viên; Scrum cho phép ưu tiên lại sau mỗi Sprint, giảm rủi ro “đi sai hướng”.
* Minh bạch & dễ quản trị: Tất cả chức năng tập trung trong Product Backlog; mỗi Sprint cam kết phạm vi cụ thể (Sprint Goal), phân rã thành Task có ước lượng, dễ theo dõi trạng thái/velocity.
* Bàn giao liên tục, cải tiến liên tục: Mỗi Sprint có Increment sẵn sàng demo, nhận phản hồi sớm để tinh chỉnh; Retrospective chốt 1–2 hành động cải tiến quy trình/kỹ thuật cho Sprint sau.
* Bám mục tiêu chất lượng: DoD yêu cầu test pass, đáp ứng AC/NFR (hiệu năng, bảo mật, khả dụng), tài liệu & log được cập nhật, qua đó nâng chất lượng bản phát hành.

Kết luận: Scrum cung cấp khung làm việc gọn, rõ và hiệu quả cho nhóm nhỏ, giúp dự án bám mục tiêu, giảm rủi ro, nâng chất lượng và tăng tốc độ bàn giao trong suốt học kỳ.

# **II: PHÂN TÍCH YÊU CẦU**

## **2.1. Yêu cầu chức năng**

**2.2.1. Khách hàng**

Đăng ký, đăng nhập, tìm kiếm, lọc, xem chi tiết:

Đăng ký hoặc đăng nhập nhanh bằng email hoặc số điện thoại, có thể xác thực OTP. Tìm theo từ khóa, nhận gợi ý tức thì, lọc theo danh mục, giá, thương hiệu, còn hàng, sắp xếp theo mới nhất hoặc bán chạy. Mở trang chi tiết để xem ảnh rõ nét, mô tả, thông số, điểm đánh giá và số lượt, tình trạng tồn kho, thời gian giao dự kiến theo địa chỉ đã lưu.  
Luồng: Đăng ký hoặc đăng nhập => tìm, lọc => xem chi tiết => chọn biến thể nếu có => thêm vào giỏ.

Giỏ hàng, đặt hàng, theo dõi và lịch sử:

Giỏ hàng cho phép đổi số lượng, xóa mục, lưu để mua sau, áp mã giảm giá và thấy lý do nếu không hợp lệ. Tại bước đặt hàng, chọn địa chỉ, ghi chú, chọn hãng vận chuyển, phí và thời gian giao hiển thị tức thì. Chọn phương thức thanh toán, thẻ hoặc ví hoặc COD, xác thực OTP hoặc 3DS khi cần, kết quả hiện rõ ràng, không tạo trùng đơn. Sau khi xác nhận, nhận mã đơn, theo dõi dòng thời gian giao hàng, xem lại lịch sử và tải hóa đơn bất cứ lúc nào.  
Luồng: Giỏ => địa chỉ => vận chuyển => thanh toán => xác nhận => theo dõi => xem lịch sử.

Hỗ trợ, chat, đánh giá, thông báo:

Khi cần trợ giúp, mở chat trong tài khoản, chọn chủ đề thanh toán hoặc vận chuyển hoặc đổi trả, đính kèm ảnh nếu cần, nhận phản hồi kịp thời. Sau khi nhận hàng, để lại đánh giá có sao và nhận xét, một đơn một đánh giá cho mỗi sản phẩm, hiển thị sau khi duyệt nhanh. Nhận thông báo về cập nhật đơn, hoàn tiền, mã giảm giá và ưu đãi mới, có thể bật tắt trong cài đặt.  
Luồng: Cần hỗ trợ => mở chat => nhận phản hồi. Sau nhận hàng => viết đánh giá => nhận thông báo.

**2.2.2. Quản trị viên**

Đăng nhập hệ thống quản trị: Quản trị viên có quyền cao nhất, đăng nhập bằng tài khoản bảo mật cao, bắt buộc 2FA khi khả dụng, phiên đăng nhập được ghi log, mọi thao tác đều được kiểm soát theo vai trò.

Quản lý người dùng và tài khoản: Xem danh sách người dùng theo bộ lọc, khóa hoặc mở tài khoản khi phát hiện rủi ro, đặt lại mật khẩu khi có yêu cầu hợp lệ, xem lịch sử đăng nhập và lịch sử thao tác để truy vết, gán vai trò và phạm vi quyền phù hợp.

Quản lý sản phẩm và kho: Tạo mới, chỉnh sửa, xóa sản phẩm, quản lý thuộc tính, ảnh, biến thể, cập nhật tồn kho theo từng kho, thực hiện nhập và xuất kho có chứng từ, đặt ngưỡng cảnh báo sắp hết hàng, theo dõi lịch sử biến động tồn.  
Luồng điển hình: phát hiện tồn thấp => tạo phiếu nhập => cập nhật tồn thành công => dỡ cảnh báo.

Quản lý đơn hàng: Xem danh sách theo trạng thái, tiếp nhận và xác nhận đơn hợp lệ, in phiếu lấy hàng và phiếu đóng gói, bàn giao vận chuyển, cập nhật giao thành công hoặc thất bại, xử lý hủy theo quy định, tiếp nhận yêu cầu trả hàng và đổi hàng, in hóa đơn và chứng từ liên quan.  
Luồng điển hình: đơn mới => xác nhận => đóng gói => giao hàng => giao thành công hoặc hủy hoặc trả hàng.

Chăm sóc khách hàng: Xem hồ sơ và lịch sử mua hàng, đọc và phản hồi yêu cầu hỗ trợ, ghi chú nội bộ để đồng bộ giữa các bộ phận, duyệt hoặc xóa đánh giá vi phạm tiêu chuẩn, đảm bảo phản hồi đúng thời gian cam kết.

Quản lý thanh toán: Theo dõi giao dịch theo thời gian thực, xác nhận giao dịch khi nhận được đối soát hợp lệ, xử lý hoàn tiền toàn phần hoặc một phần, kiểm tra kết nối với cổng thanh toán, lưu log yêu cầu và phản hồi để tra soát, bảo đảm không phát sinh giao dịch trùng.

Khuyến mãi và marketing: Tạo mã giảm theo sản phẩm, theo thương hiệu, theo giá trị giỏ, đặt điều kiện áp dụng và thời gian hiệu lực, giới hạn cho nhóm khách hàng cụ thể, theo dõi hiệu quả chiến dịch, dừng hoặc gia hạn khi cần.

Báo cáo và thống kê: Theo dõi doanh thu theo ngày và theo kênh, xem tồn kho hiện tại và vòng quay tồn, theo dõi tỷ lệ đơn theo trạng thái từ xác nhận đến giao thành công, quan sát tỷ lệ hủy và tỷ lệ trả hàng, xuất báo cáo định kỳ để phục vụ lập kế hoạch vận hành và nhập hàng

### **2.2.3. Cổng thanh toán**

Xử lý thanh toán: Nhận yêu cầu từ hệ thống, xác thực thông tin người trả và đơn hàng, kiểm tra hạn mức và tính hợp lệ, tạo mã giao dịch và trả kết quả thành công hoặc thất bại, gửi webhook để đồng bộ trạng thái.

Luồng tối thiểu: Tạo yêu cầu => xác thực => ủy quyền hoặc từ chối => trả kết quả kèm transactionId.

Bảo mật giao dịch: Hỗ trợ OTP hoặc 3DS hoặc sinh trắc khi khả dụng, truyền dữ liệu qua SSL hoặc TLS ≥ 1.2, mã hóa thông tin nhạy cảm khi lưu trữ, tuân thủ PCI DSS, áp dụng cơ chế chống gian lận theo rủi ro, ghi nhận thiết bị và địa chỉ IP để phát hiện bất thường.

Kết nối và hậu kiểm: Kết nối ngân hàng và ví điện tử và chuyển khoản, hỗ trợ hoàn tiền toàn phần hoặc một phần, ghi log đầy đủ yêu cầu và phản hồi, cung cấp báo cáo đối soát định kỳ, cho phép tra soát theo mã giao dịch ngoài và trong hệ thống.

## **2.2. Yêu cầu phi chức năng**

### **2.2.1. Hiệu suất**

Trang và ứng dụng: TTFB ≤ 800 ms p95. Thời gian tải đầy đủ < 5 s với < 1000 người dùng đồng thời. Mục tiêu giữ ổn định p99 ≤ 1200 ms cho TTFB.

API nghiệp vụ: Danh mục, sản phẩm, tìm kiếm có p95 < 300 ms. Các bước thanh toán có p95 < 500 ms. Truy vấn nặng có p99 < 800 ms. Tất cả endpoint đều có timeout hợp lý và trả lỗi có thể phục hồi.

Thông lượng: Xử lý ≥ 500 đơn/phút ổn định. Tìm kiếm ≥ 5000 truy vấn/phút. Callback thanh toán ≥ 1000 lượt/phút. Duy trì headroom ≥ 30 phần trăm để hấp thụ dao động tải.

Flash sale và pre-order: Chịu burst 5× trong 10 phút. Tỷ lệ lỗi < 0,5 phần trăm trong cửa sổ này. Ngân sách độ trễ không tính phần chờ của bên thứ ba. Thanh toán có retry có kiểm soát và idempotency để tránh trùng giao dịch.

Đo lường và phạm vi: Ghi rõ kịch bản đo, thiết bị và kết nối. Ví dụ, di động 4G và wifi gia dụng. Báo cáo cả p95 và p99. Tách bạch thời gian render phía client và thời gian xử lý phía server. Nêu cấu hình dữ liệu mẫu và kích thước phản hồi khi test.

Chấp nhận và giám sát: Chỉ tiêu đạt khi giữ vững 30 phút liên tục dưới tải mục tiêu. Theo dõi real-time, cảnh báo khi p95 vượt ngưỡng 10 phút liên tiếp. Lưu log truy vết và nhãn giao dịch để phân tích sau sự cố.

**2.2.2. Khả năng mở rộng**

Mục tiêu: Có thể ×2 số SKU và ×2 MAU trong ≤ 6 tháng, vẫn giữ SLA hiệu suất ở 3.1.

Chiến lược: Hỗ trợ scale ngang bằng cách thêm node cho web, API, search, cache. Hỗ trợ scale dọc bằng cách tăng CPU và RAM cho các nút nóng. Định lượng năng lực bằng số user đồng thời, tần suất giao dịch, dung lượng dữ liệu và tốc độ tăng trưởng theo tháng.

Thiết kế: Tách lớp trình bày, dịch vụ, dữ liệu, dùng cache phân tán cho đọc nóng, tách đọc và ghi ở cơ sở dữ liệu, hàng đợi cho tác vụ nền, idempotency cho thanh toán và đơn hàng.

Năng lực và headroom: Duy trì headroom ≥ 30 phần trăm ở giờ cao điểm, kiểm thử tải định kỳ theo kịch bản ×2 và ×3 để phát hiện nút thắt cổ chai sớm.

Tiêu chí chấp nhận: Khi SKU và MAU tăng gấp đôi, p95 của các API chính vẫn trong ngưỡng 3.1, tỷ lệ lỗi 5xx ổn định dưới ngưỡng cảnh báo, chi phí hạ tầng tăng tuyến tính hoặc dưới tuyến tính theo tải.

**2.2.3. Tương thích & Di động**

Thiết bị và trình duyệt: Hỗ trợ Chrome, Edge, Safari, Firefox các phiên bản trong 2 năm gần nhất, tối ưu mobile với viewport 360 đến 414 px, UI nhất quán trên desktop, tablet, mobile.

Hành vi và hiển thị: Responsive theo breakpoint, điều hướng bằng cảm ứng mượt mà, font hệ thống rõ ràng, kích thước nút chạm đạt chuẩn tiếp cận, trạng thái tải và lỗi hiển thị nhất quán.

Tích hợp: Tương thích các cổng thanh toán và đơn vị vận chuyển phổ biến, chuẩn trao đổi CSV và XLSX cho đối soát và nhập xuất dữ liệu, chống lỗi mã hóa khi xử lý tiếng Việt và ký tự đặc biệt.

Kiểm thử: Kiểm thử chéo thiết bị và trình duyệt trước phát hành, đo p95 render trên mobile thật, mô phỏng mạng 4G và wifi gia dụng, ghi rõ phạm vi đo và dữ liệu mẫu để lặp lại.

**2.2.4. Độ tin cậy & Khả dụng**

Khả dụng (SLA):

Mặt trước ≥ 99,9%/tháng, downtime ≤ 0,1% ≈ 0,72 giờ/tháng.

Luồng đơn hàng và thanh toán ≥ 99,95%/tháng, downtime ≤ 0,05% ≈ 0,36 giờ/tháng.

Thành phần. landing ≥ 99,95%/tháng ≈ ≤ 0,36 giờ/tháng, checkout ≥ 99,95%/tháng ≈ ≤ 0,36 giờ/tháng, dashboard ≥ 99,9%/tháng ≈ ≤ 0,72 giờ/tháng.

Đo lường và phạm vi: Đo RUM và synthetic từ góc nhìn người dùng, báo cáo p95, p99, tách client render và server time. Bảo trì có kế hoạch được loại trừ nếu thông báo trước và tổng thời lượng ≤ 2 giờ/tháng.

Giám sát và ngân sách lỗi:

Front: error budget 0,1%/tháng ≈ 0,72 giờ/tháng.

Checkout: error budget 0,05%/tháng ≈ 0,36 giờ/tháng. Cảnh báo khi tiêu thụ ≥ 50% ngân sách lỗi trong 7 ngày, theo dõi realtime tỷ lệ lỗi và p95 độ trễ.

Chịu lỗi: Retry có backoff và jitter, circuit breaker cho tích hợp ngoài, hàng đợi cho tác vụ nền, lưu tạm đơn, cho phép chuyển phương thức thanh toán khi cổng lỗi, hạ cấp tính năng không thiết yếu để giữ luồng mua hàng hoạt động.

**2.2.5. Bảo trì**

MTTR: Khôi phục Sev1 ≤ 60 phút, MTTD ≤ 5 phút nhờ cảnh báo tự động.

Kiến trúc mô đun: Biên dịch vụ rõ ràng, thay đổi cục bộ không làm gián đoạn toàn hệ thống.

Triển khai an toàn: Canary 5% => 25% => 100%, blue green khi cần, rollback ≤ 5 phút.

Lịch bảo trì: Cửa sổ 02:00–04:00 theo giờ địa phương, tổng thời lượng ≤ 2 giờ/tháng, thông báo trước ≥ 24 giờ.

Quản lý thay đổi: Review hai người, CI/CD tự động, checklist trước và sau phát hành, ghi rõ phạm vi ảnh hưởng.

Dữ liệu và schema: Migration tương thích ngược, feature flag bật tắt theo nhóm, backfill chạy nền, kiểm tra dữ liệu trước khi cắt cờ.

Thay thế nóng: Dịch vụ stateless scale ngang, thay pod không downtime, cơ sở dữ liệu có replica đọc, failover < 1 phút.

Sao lưu và phục hồi: RPO ≤ 5 phút với binlog hoặc CDC, snapshot hằng ngày, diễn tập khôi phục mỗi quý.

Runbook và nhật ký: Runbook cho sự cố phổ biến, SLO sửa lỗi Sev2 ≤ 2 ngày làm việc, lưu audit và changelog ≥ 180 ngày.

Vòng đời sản phẩm: Ưu tiên MVP cho ngắn hạn, kế hoạch refactor 3 đến 6 tháng cho dài hạn, lộ trình loại bỏ tính năng cũ với thông báo ≥ 30 ngày.

**2.2.6. Bảo mật & Tuân thủ**

Mã hóa và chuẩn: Toàn kênh dùng TLS ≥ 1.2, ưu tiên 1.3. Dữ liệu nhạy cảm mã hóa at rest bằng AES-256. Mật khẩu lưu bằng bcrypt hoặc argon2, kèm salt và tham số chi phí phù hợp. Không lưu PAN hoặc CSC, chỉ lưu token hóa từ cổng thanh toán.

Thanh toán an toàn: Tích hợp theo PCI DSS, áp dụng 3DS hoặc OTP khi khả dụng. Mọi callback phải xác minh chữ ký và thời hạn, đối chiếu số tiền và mã đơn. Ghi nhật ký đối soát đầy đủ, lưu cả request và response, dùng idempotency để chống giao dịch trùng.

Quyền riêng tư và pháp lý: Công bố quyền truy xuất và xóa dữ liệu của người dùng, hỗ trợ yêu cầu trong thời hạn cam kết. Hóa đơn điện tử đúng quy định. Chính sách bảo hành và đổi trả minh bạch, hiển thị ngay trong bước checkout.

Kiểm soát truy cập và phòng vệ: RBAC theo nguyên tắc ít quyền nhất, bắt buộc 2FA cho tài khoản quản trị. Audit log bất biến cho các thay đổi nhạy cảm như trạng thái đơn, hoàn tiền, điều chỉnh tồn. Áp dụng rate limit, danh tiếng IP, dấu vân tay thiết bị, phát hiện bất thường theo rủi ro.

Tiêu chí chấp nhận: SAST và DAST không còn lỗ hổng mức High hoặc Critical trước khi phát hành. Pentest định kỳ và sau các thay đổi lớn. Quy tắc bất biến, callback sai chữ ký hoặc sai số tiền hoặc sai mã đơn => không cập nhật PAID, mở tra soát, cảnh báo đội vận hành

**2.2.7. Khả năng sử dụng**

Thời gian hoàn tất: Người dùng mới từ tìm sản phẩm đến thêm vào giỏ đến thanh toán < 2 phút. Quy trình checkout ≤ 3 bước. Tự động gợi ý địa chỉ và phương thức vận chuyển. Tự động điền thông tin khi người dùng đã từng mua.

Luồng thao tác chuẩn. Trang chủ => tìm hoặc lọc => xem chi tiết => thêm vào giỏ. Giỏ hàng => địa chỉ => vận chuyển => thanh toán => xác nhận. Mỗi bước có trạng thái tải rõ ràng và thông báo lỗi dễ hiểu.

Đo lường theo NN Group, 5 thuộc tính:

Learnability: Người mới hoàn thành lần đầu < 2 phút. Tìm sản phẩm ≤ 3 click từ trang chủ.

Efficiency: Người đã quen hoàn tất mua lặp lại < 60 giây. Số trường nhập ở checkout ≤ 8.

Memorability: Sau 2 tuần không dùng, tỉ lệ quay lại thực hiện thành công ≥ 95 phần trăm mà không cần hướng dẫn.

Errors: Tỉ lệ lỗi nhập thẻ < 10 phần trăm. Mỗi lỗi có gợi ý sửa cụ thể. Không mất dữ liệu giỏ khi tải lại trang.

Satisfaction: Điểm hài lòng sau giao dịch ≥ 4 trên 5. Tỷ lệ bỏ giỏ do UI < 5 phần trăm.

Tiêu chí chấp nhận và giám sát: Ghi thời gian tác vụ end to end cho từng người dùng. Báo cáo p50, p95 theo thiết bị và mạng. Theo dõi tỷ lệ thành công mỗi bước của kênh checkout. Cảnh báo khi p95 thời gian hoàn tất > 2 phút hoặc lỗi nhập thẻ > 10% trong 1 giờ.

**2.2.8. Quan sát & Vận hành**

**- Giám sát và cảnh báo:**

Bao phủ quan sát: Áp dụng RED cho web service, request rate, error rate, duration. Áp dụng USE cho hạ tầng, utilization, saturation, errors trên CPU, RAM, I/O, network, cache, DB. Bật tracing phân tán end to end, gắn orderId và transactionId. Log tập trung dạng JSON, gắn requestId, lưu tối thiểu 30 ngày cho vận hành, 180 ngày cho bảo mật.

Bảng điều khiển thời gian thực: Dashboard p95, p99 cho API chính, tìm kiếm, thanh toán, cùng tỷ lệ lỗi 4xx, 5xx, hàng đợi nền, webhook, tỷ lệ thành công callback, độ trễ DB, cache hit ratio.

Cảnh báo và ngưỡng: Cảnh báo khi chỉ số lệch mục tiêu SLA > 10% trong cửa sổ 10 phút, ví dụ p95 thanh toán mục tiêu 500 ms, cảnh báo khi > 550 ms liên tục. Cảnh báo khi error rate tăng đột biến, khi tỷ lệ thành công checkout giảm, khi backlog hàng đợi vượt ngưỡng.

Khả năng tự phục hồi: Retry có backoff và jitter cho webhook, circuit breaker với tích hợp ngoài, autoscale theo CPU hoặc RPS, hạ cấp tính năng không thiết yếu khi tải tăng.

- Sao lưu và khôi phục thảm họa, DR:

Sao lưu định kỳ: Backup DB hằng ngày, giữ 30 ngày. Sao lưu binlog hoặc CDC để đạt RPO ≤ 15 phút. Mã hóa backup, kiểm tra tính toàn vẹn sau mỗi lần.

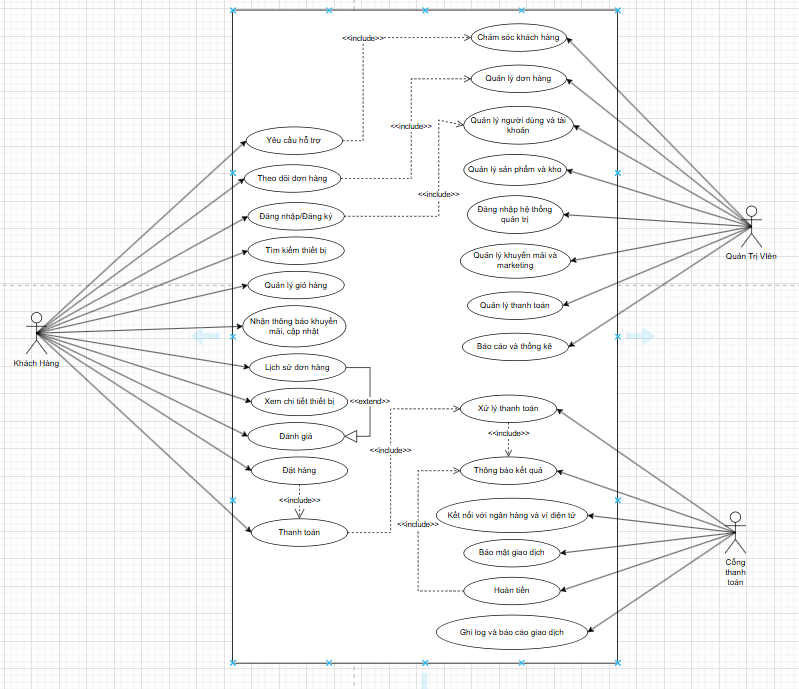
Mục tiêu khôi phục: RPO ≤ 15 phút, RTO ≤ 60 phút cho luồng quan trọng. Đặt SLA theo từng component, đặc biệt checkout và thanh toán.

Kiến trúc DR: Replica đọc ở vùng dự phòng, failover có kiểm soát, thử khôi phục trên môi trường tách biệt. Lưu cấu hình hạ tầng bằng IaC để phục dựng nhanh.

Diễn tập DR: Diễn tập hàng quý, kịch bản mất một vùng, hỏng DB, cổng thanh toán không phản hồi. Ghi nhận thời gian thực tế so với RPO và RTO, cập nhật runbook sau mỗi lần diễn tập.

Tiêu chí chấp nhận: Khôi phục bản sao gần nhất trong ≤ 60 phút, mất dữ liệu tối đa trong 15 phút gần nhất, không ghi đúp giao dịch sau khi phục hồi, đối soát sau DR khớp 100% cho các đơn và thanh toán đã “PAID”.

## **2.3. Mô hình hóa**



## **2.4. Kịch bản**

### **2.4.1. Trường hợp sử dụng: Đặt hàng**

Mục tiêu**:** Tạo một đơn gồm Laptop và Điện thoại, hoàn tất trơn tru.

Giao đoạn:

1. Khách hàng đã đăng nhập, gõ tìm hoặc vào danh mục, chọn một điện thoại đời mới và một laptop cấu hình cao.
2. Ở trang chi tiết, hệ thống hiển thị rõ “**Còn hàng**”, khách chọn biến thể, màu sắc, dung lượng đúng nhu cầu.
3. Khách bấm “**Thêm vào giỏ**”, cả hai sản phẩm vào giỏ ngay, hệ thống giữ chỗ tồn trong **~45 phút** để tránh mua trùng.
4. Ở giỏ, tổng tiền đã bao gồm VAT hiển thị minh bạch, không có phí ẩn. Khách bấm “**Tiến hành thanh toán**”.
5. Khách xác nhận địa chỉ, chọn vận chuyển “**Nhanh**”, phí hiển thị tức thì, không bất ngờ khi tới bước cuối.
6. Khách bấm “**Xác nhận đặt hàng**”.

Kết quả:

* Hệ thống tạo đơn **#DH9001**, trạng thái **“**Chờ thanh toán**”**.
* Dữ liệu đơn được “khóa” ở mức cần thiết để không bị sửa sai, tiếp tục chuyển sang quy trình Thanh toán.

#### **2.4.1.1. Kịch bản phụ: Hết hàng đột ngột ngay thời điểm xác nhận**

Bối cảnh: Máy tính X chỉ còn 1 chiếc. Một khách khác vừa thanh toán xong chiếc cuối cùng.

Giao đoạn:

1. Khách bấm “Xác nhận đặt hàng”.
2. Ở bước kiểm tra cuối, hệ thống đối chiếu tồn kho thực tế và thấy **= 0**.

Kết quả và thông báo gửi đến khách hàng:

* Đơn không được tạo với mặt hàng đã hết.
* Thông báo hiện rõ: “**Rất tiếc, [Máy tính X] vừa bán hết. Vui lòng xóa khỏi giỏ hoặc chọn mẫu thay thế**.”
* Khách có thể tiếp tục với các sản phẩm còn lại sau khi cập nhật giỏ.

### **2.4.2. Trường hợp sử dụng: Đặt hàng**

Mục tiêu: Thanh toán thành công cho đơn #DH9001.

Giao đoạn:

1. Khách chọn phương thức **Thẻ tín dụng/Ghi nợ**, nhập thông tin của thẻ.
2. Cổng thanh toán kích hoạt xác thực an toàn, ví dụ **OTP/3DS**, gửi yêu cầu đến ngân hàng phát hành.
3. Ngân hàng **chấp thuận**, cổng trả về kết quả **Thành công** kèm **transactionId**.

Kết quả:

Đơn cập nhật sang **“Đã thanh toán, Chờ xử lý”**.

Giữ hàng tạm thời được **hủy**, tồn kho được **trừ vĩnh viễn**.

Hóa đơn điện tử và thông báo xác nhận được gửi qua email hoặc SMS ngay sau đó.

#### **2.4.2.1. Kịch bản phụ: Thẻ bị từ chối do không đủ số dư**

**Bối cảnh:** Đơn điện thoại trị giá **30.000.000 VNĐ**.

**Cách diễn ra.**

1. Ngân hàng trả lỗi **“Không đủ tiền trong tài khoản”** cho cổng thanh toán.
2. Hệ thống hiển thị thông điệp rõ ràng, “Thanh toán không thành công. Thẻ không đủ số dư. Vui lòng kiểm tra lại hoặc chọn phương thức khác.”

**Kết quả và hướng xử lý.**

Khách chuyển sang COD.

Đơn cập nhật trạng thái “**Chờ xử lý, Thanh toán COD**”, quy trình tiếp tục như bình thường.

### **2.4.3. Trường hợp sử dụng: Hủy đơn hàng & Hoàn tiền**

**Mục tiêu:** Cho phép khách hàng hủy khi hàng chưa rời kho và tiền được hoàn tự động.

Giao đoạn:

1. Khách mở đơn #DH9004, trạng thái “**Đã thanh toán, chờ xử lý**”, bấm “**Hủy đơn hàng”.**
2. Hệ thống kiểm tra điều kiện, chưa bàn giao vận chuyển ⇒ cho phép hủy.
3. Khách chọn lý do, xác nhận. Hệ thống chuyển trạng thái “**Đã hủy**”, trả tồn kho ngay, đồng thời gửi yêu cầu hoàn tiền sang cổng thanh toán.

Kết quả: Email xác nhận gửi ngay: “**Yêu cầu hủy đã hoàn tất. Số tiền […] sẽ hoàn về thẻ/tài khoản trong 5–7 ngày làm việc**.”

#### **2.4.3.1. Kịch bản phụ 3.1 — Chặn hủy khi đã xuất kho**

Bối cảnh: Đơn #DH9005 đã “**Đang giao hàng**”, có mã vận đơn.

Giao đoạn và kết quả:

* Khi khách mở đơn, nút “**Hủy đơn hàng**” được ẩn hoặc vô hiệu hóa.
* Thông báo giải thích ngắn gọn và thẳng thắn: **“Đơn đang được vận chuyển nên không thể hủy. Nếu không còn nhu cầu, vui lòng từ chối nhận khi giao tới, hoặc làm thủ tục trả hàng sau khi nhận theo quy định.”**
* Từ đây, khách có thể theo hướng dẫn từ chối nhận hoặc tạo yêu cầu trả hàng sau khi bưu tá giao đến.

# **III: THIẾT KẾ HỆ THỐNG**

Chương này mô tả chi tiết quy trình thiết kế hệ thống dành cho nền tảng quản lý cửa hàng thiết bị điện tử trực tuyến. Quy trình được phân chia thành hai phần chính: thiết kế kiến trúc tổng quát nhằm xác định cấu trúc tổng thể, và thiết kế cụ thể với các biểu đồ UML để minh họa các yếu tố cùng sự tương tác nội bộ.

## **3.1. Thiết kế kiến trú****c tổng thể**

Thiết kế kiến trúc đóng vai trò cốt lõi, định hình cấu trúc cấp cao của hệ thống, bao gồm cách tổ chức và kết nối các thành phần chính. Việc chọn mô hình kiến trúc đúng đắn là yếu tố quyết định, ảnh hưởng lớn đến các tiêu chí phi chức năng như khả năng bảo trì, mở rộng quy mô, hiệu suất vận hành và an ninh.

### **3.1.1. Lựa chọn mô hình kiến trúc**

Dựa vào quy mô dự án và các yêu cầu đã nêu, nhóm em đã đánh giá các mô hình kiến trúc phổ biến như Monolithic (kiến trúc nguyên khối), Microservices, và Three-Tier (kiến trúc ba tầng) để tìm ra lựa chọn phù hợp nhất:

* **Kiến trúc Monolithic**: Toàn bộ ứng dụng được phát triển dưới dạng một khối thống nhất. Mô hình này nổi bật với sự đơn giản trong phát triển và triển khai ban đầu. Tuy nhiên, khi hệ thống trở nên phức tạp, việc bảo trì, khắc phục lỗi hoặc cập nhật một phần nhỏ có thể làm ảnh hưởng toàn bộ, trái với yêu cầu về "Khả năng bảo trì" đã đặt ra.
* **Kiến trúc Three-Tier**: Mô hình này chia ứng dụng thành ba tầng logic riêng biệt: Trình bày, Nghiệp vụ và Dữ liệu. Nó mang lại sự cân bằng tốt giữa tính dễ dàng và khả năng sắp xếp, phù hợp với các ứng dụng web tiêu chuẩn và đáp ứng hiệu quả các yêu cầu dự án.
* **Kiến trúc Microservices**: Hệ thống được tách thành nhiều dịch vụ nhỏ, hoạt động độc lập. Ưu điểm là tính linh hoạt cao và khả năng mở rộng mạnh mẽ. Thế nhưng, việc xây dựng và quản lý Microservices đòi hỏi hạ tầng phức tạp cùng chi phí vận hành lớn, không lý tưởng cho dự án quy mô trung bình với đội ngũ 5 thành viên.

**Lý do lựa chọn kiến trúc Three-Tier:**

* Phân tách rõ ràng Kiến trúc 3-Lớp chia hệ thống thành ba tầng logic độc lập: Tầng Trình bày, Tầng Nghiệp vụ và Tầng Dữ liệu. Sự phân chia này khớp hoàn hảo với yêu cầu "Mã nguồn có cấu trúc" và "Khả năng bảo trì". Nó hỗ trợ các thành viên nhóm làm việc song song trên các tầng khác nhau (ví dụ: một người chuyên về giao diện ở Tầng Trình bày, trong khi những người khác xử lý logic ở Tầng Nghiệp vụ), giảm thiểu tác động lẫn nhau và nâng cao hiệu quả nhóm.
* Khả năng mở rộng linh hoạt: Dù không uyển chuyển như Microservices, kiến trúc 3-Lớp vẫn cho phép mở rộng từng tầng một cách riêng lẻ khi cần. Chẳng hạn, nếu lượng truy cập tăng vọt, hệ thống có thể thêm máy chủ web ở Tầng Trình bày hoặc nhân rộng máy chủ ứng dụng ở Tầng Nghiệp vụ mà không làm gián đoạn các tầng còn lại.
* Tăng cường bảo mật: Việc cách ly Tầng Dữ liệu khỏi truy cập trực tiếp từ phía client (Tầng Trình bày) là nguyên tắc an ninh cơ bản. Mọi truy vấn từ người dùng đều phải qua Tầng Nghiệp vụ để xác thực và kiểm soát quyền hạn trước khi tiếp cận cơ sở dữ liệu. Điều này trực tiếp thỏa mãn yêu cầu "Phân quyền truy cập", giúp ngăn chặn các truy cập không được phép vào dữ liệu quan trọng.
* Tương thích với công nghệ đương đại: Kiến trúc này hòa hợp tốt với các framework web hiện đại. Tầng Trình bày có thể sử dụng ReactJS, còn Tầng Nghiệp vụ có thể áp dụng các công cụ mạnh như Node.js hoặc Laravel – những lựa chọn đã được đề cập trong kế hoạch phát triển.

**3.2. Biểu đồ Gói**

**A screenshot of a computer

AI-generated content may be incorrect.**

**3.2.1. Các tầng và mục đích**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tầng | |  | | --- | | Vai trò |  |  | | --- | |  | | | Lớp/Thành phần tiêu biểu | | --- |  |  | | --- | |  | |
| UI | Nhận yêu cầu từ Admin/Customer, điều hướng luồng nghiệp vụ qua controller | **AdminUI, CustomerUI, OrderController, ProductController, PaymentController, CartController** |
| service | Xử lý logic nghiệp vụ, kiểm tra điều kiện, phối hợp nhiều repository | **OrderService, ProductService, PaymentService, CartService, InventoryService** |
| repository | |  | | --- | | Truy vấn/ghi dữ liệu của các thực thể |  |  | | --- | |  | | **OrderRepository, ProductRepository, PaymentRepository, CartRepository, CustomerRepository** |
| domain | Mô hình hóa dữ liệu cốt lõi và quan hệ | **Order, Product (Laptop, Smartphone), Payment, Cart, Customer** |

**3.2.2. Lớp giao diện**

|  |  |  |
| --- | --- | --- |
| Lớp | Nhóm | Chức năng/Hoạt động chính |
| |  | | --- | | AdminUI |  |  | | --- | |  | | UI | quảnLýSảnPhẩm(), xemĐơnHàng() |
| CustomerUI | UI | |  | | --- | | đăngNhập(), xemSảnPhẩm(), xemChiTiet(), thanhToán() |  |  | | --- | |  | |
| OrderController | Controller | |  | | --- | | nhận yêu cầu tạo/xem đơn → gọi **OrderService** |  |  | | --- | |  | |
| ProductController | Controller | |  | | --- | | tìm kiếm/xem chi tiết SP → gọi **ProductService** |  |  | | --- | |  | |
| PaymentController | Controller | thao tác thanh toán → gọi **PaymentService** |
| CartController | Controller | thêm/xóa/xem giỏ → gọi **CartService** |

**3.2.3. Lớp nghiệp vụ**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nghiệp vụ | |  | | --- | | Mục tiêu |  |  | | --- | |  | | | Tương tác chính | | --- |  |  | | --- | |  | |
| OrderRepository | |  | | --- | | tạoĐơn(), xácNhậnĐơn() |  |  | | --- | |  | | dùng **OrderRepository**, tham chiếu **CartService**, kiểm tra tồn kho (I**nventoryService**) |
| |  | | --- | | ProductRepository |  |  | | --- | |  | | |  | | --- | | tìmKiếmSảnPhẩm(), xemChiTiet() |  |  | | --- | |  | | dùng **ProductRepository** |
| PaymentRepository | |  | | --- | | xửLýThanhToán() |  |  | | --- | |  | | dùng **PaymentRepository**, cập nhật trạng thái **Order** |
| CartRepository | thêmSP(), xóaSP(), tínhTổng() | dùng **CartRepository**, truy vấn **ProductRepository** để lấy giá/thuộc tính |
| CustomerRepository | cậpNhậtKho() | thường được **OrderService** gọi khi xác nhận đơn |

**3.2.4. Lớp truy cập dữ liệu**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | **Repository** |  |  | | --- | |  | | | **Các thao tác chính** | | --- |  |  | | --- | |  | |
| OrderRepository | save(), findById() |
| ProductRepository | |  | | --- | | findAll(), findByCategory(), save() |  |  | | --- | |  | |
| PaymentRepository | |  | | --- | | save() |  |  | | --- | |  | |
| CartRepository | |  | | --- | | save(), clear() |  |  | | --- | |  | |
| CustomerRepository | findByUsername(), save() |

**3.2.5. Lớp đối tượng nghiệp vụ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Thực thể | Thuộc tính chính | | Ghi chú/Quan hệ | | --- |  |  | | --- | |  | |
| Order | id:String, ngày:Date, tổngTiền:Decimal | liên hệ **Customer**, gồm nhiều **Product** qua **Cart** |
| Product | id:String, tên:String, giá:Decimal | có **phân loại con**: **Laptop, Smartphone** |
| Payment | |  | | --- | | id:String, phươngThức:String |  |  | | --- | |  | | gắn với một **Order** |
| Cart | |  | | --- | | items:List |  |  | | --- | |  | | chứa danh sách sản phẩm + số lượng |
| |  | | --- | | Customer |  |  | | --- | |  | | |  | | --- | | id:String, tên:String |  |  | | --- | |  | | xác thực từ **CustomerUI / CustomerRepository** |

**3.2.6. Luồng phụ thuộc chính**

|  |  |
| --- | --- |
| Nguồn → Đích | Ý nghĩa |
| |  | | --- | | **UI/Controller → Service** |  |  | | --- | |  | | Controller chuyển yêu cầu người dùng thành lệnh nghiệp vụ (create order, add to cart, pay…) |
| |  | | --- | | **Service → Repository** |  |  | | --- | |  | | |  | | --- | | Service đọc/ghi dữ liệu (sản phẩm, giỏ, đơn, thanh toán, khách hàng) |  |  | | --- | |  | |
| |  | | --- | | **Repository ↔ Domain** |  |  | | --- | |  | | |  | | --- | | Repository thao tác trực tiếp trên đối tượng miền (**Order, Product**, …) |  |  | | --- | |  | |
| |  | | --- | | **Service ↔ Service** |  |  | | --- | |  | | |  | | --- | | **OrderService** gọi **InventoryService**/**CartService** để xác nhận đơn và trừ kho |  |  | | --- | |  | |
| |  | | --- | | **UI ↔ UI** |  |  | | --- | |  | | **AdminUI** và **CustomerUI** là hai điểm vào khác nhau (quyền/luồng khác nhau) nhưng đều qua controller tương ứng |

**3.2.7. Luồng ánh xạ dữ liệu**

|  |  |  |
| --- | --- | --- |
| Trường hợp sử dụng | Bước chính | Lớp tham gia |
| Khách xem danh mục & chi tiết | CustomerUI → ProductController → ProductService → ProductRepository | **CustomerUI, ProductController, ProductService, ProductRepository** |
| |  | | --- | | Thêm vào giỏ |  |  | | --- | |  | | CustomerUI → CartController → CartService → (ProductRepository để lấy giá) → CartRepository | **CartService, CartRepository** |
| |  | | --- | | Tạo & xác nhận đơn |  |  | | --- | |  | | CustomerUI → OrderController → OrderService → (CartService, InventoryService) → OrderRepository | **OrderService, OrderRepository** |
| |  | | --- | | Thanh toán |  |  | | --- | |  | | CustomerUI → PaymentController → PaymentService → PaymentRepository → cập nhật **Order** | |  | | --- | | **PaymentService, PaymentRepository** |  |  | | --- | |  | |
| |  | | --- | | Admin quản lý sản phẩm |  |  | | --- | |  | | AdminUI → ProductController → ProductService → ProductRepository (save/findByCategory) | **AdminUI, ProductService, ProductRepository** |

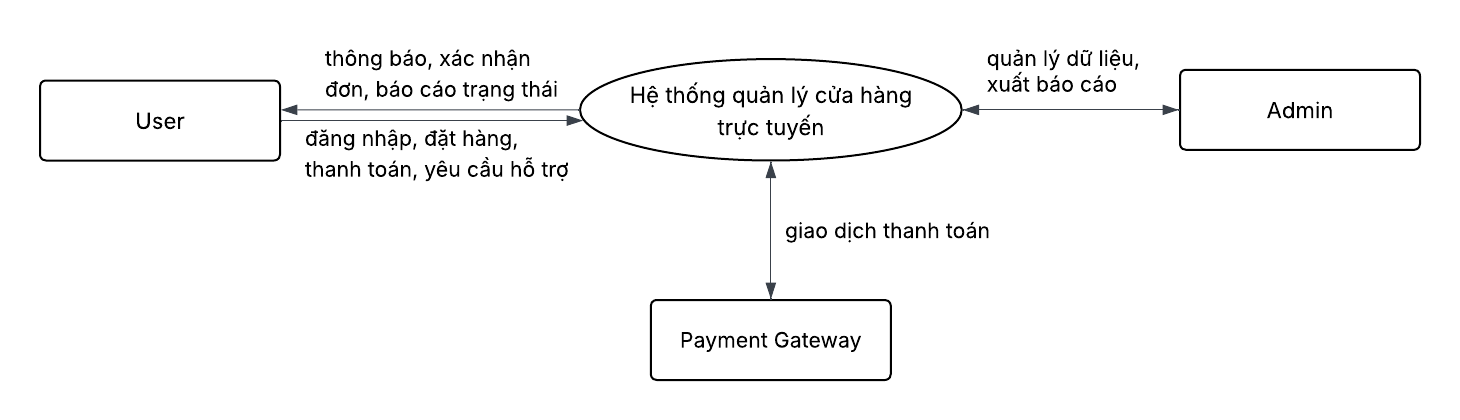
## **3.3. Biểu đồ dữ liệu**

**A computer screen shot of a diagram

AI-generated content may be incorrect.**

**3.4. Mô hình luồng dữ liệu**

**3.4.1. Cấp 0:**

****

**3.4.2. Cấp 1:**

**A diagram of a company

AI-generated content may be incorrect.**

## **3.2. Thiết kế chi tiết (UML)**

### **3.2.1. Sơ đồ lớp (Class Diagram)**

#### **3.2.1.1. Nhóm Người dùng & Module quản trị**

**A screenshot of a computer

AI-generated content may be incorrect.**

#### **3.2.1.2. Nhóm Catalog (Product, Category) & Inventory**

**A screenshot of a computer

AI-generated content may be incorrect.**

#### **3.2.1.3. Nhóm Cart**

**A screenshot of a computer

AI-generated content may be incorrect.**

#### **3.2.1.4. Nhóm Order & Discount**

**A screenshot of a computer

AI-generated content may be incorrect.**

#### **3.2.1.5. Nhóm Payment**

**A screenshot of a computer program

AI-generated content may be incorrect.**

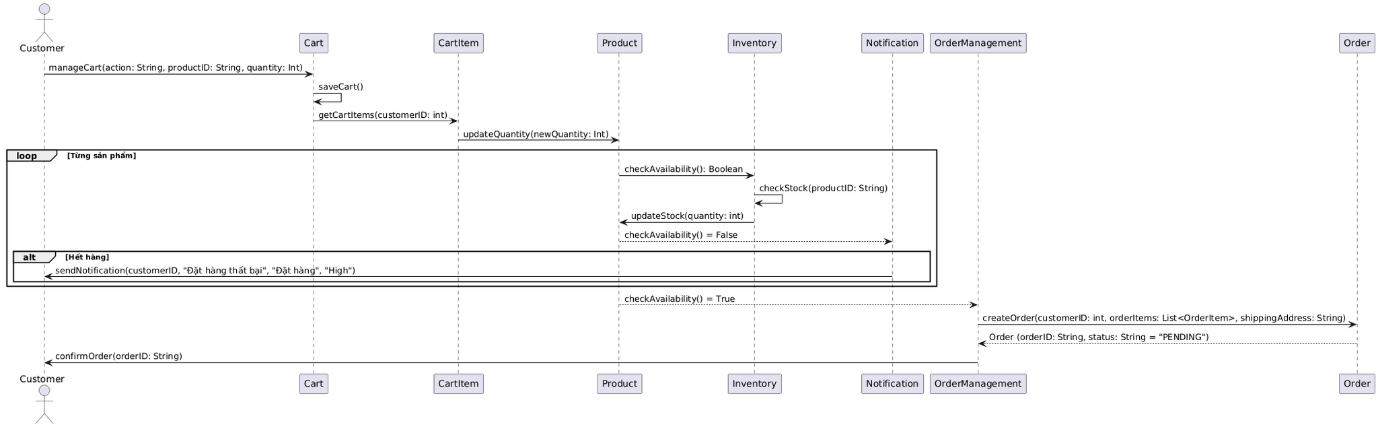
#### **3.2.1.6. Nhóm CSKH, Review & Notification**

**A screenshot of a computer

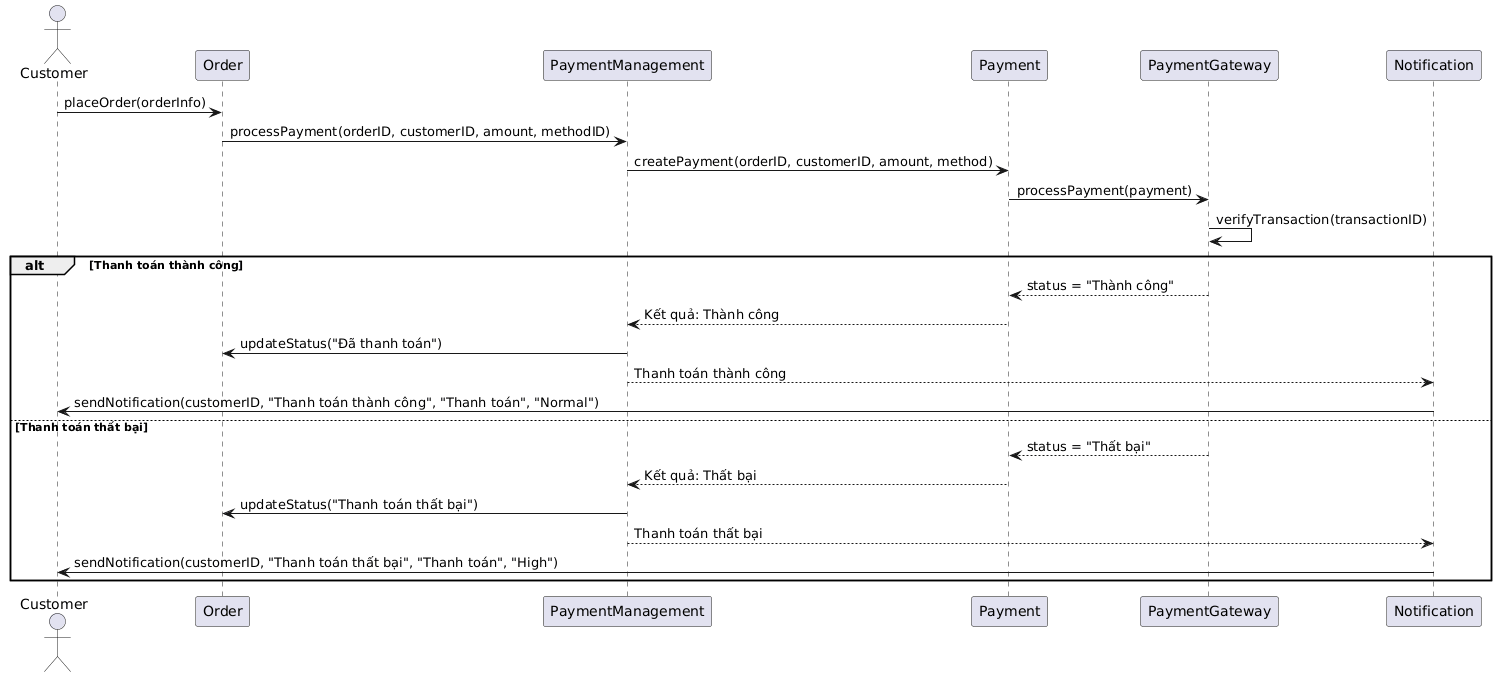
AI-generated content may be incorrect.**

### **3.2.3. Sơ đồ trình tự (Sequence Diagram)**

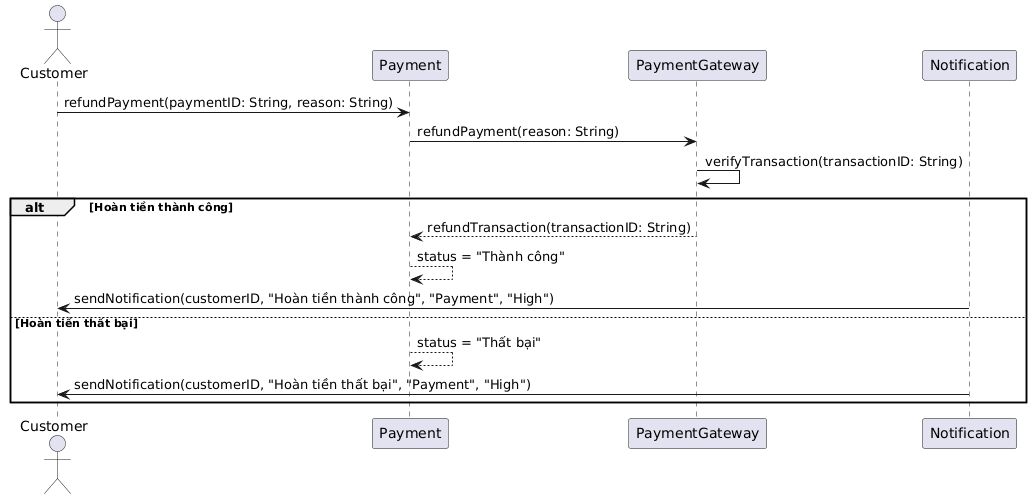
Đặt hàng:



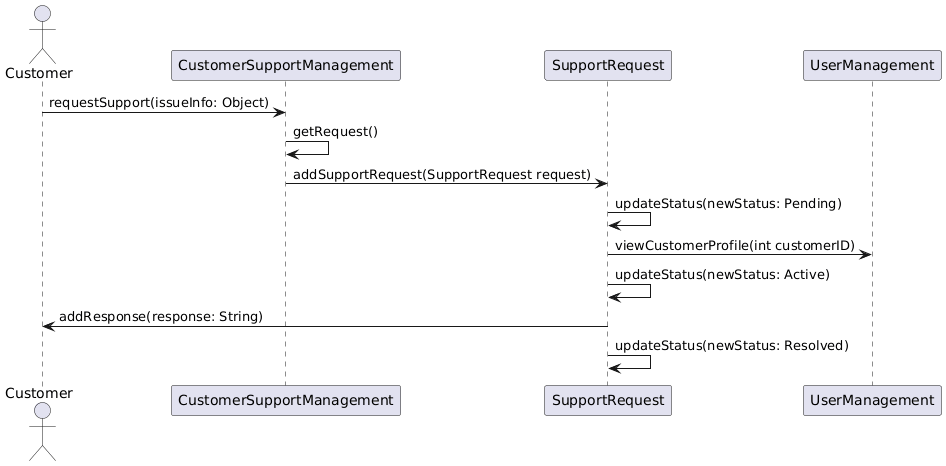
Thanh toán:



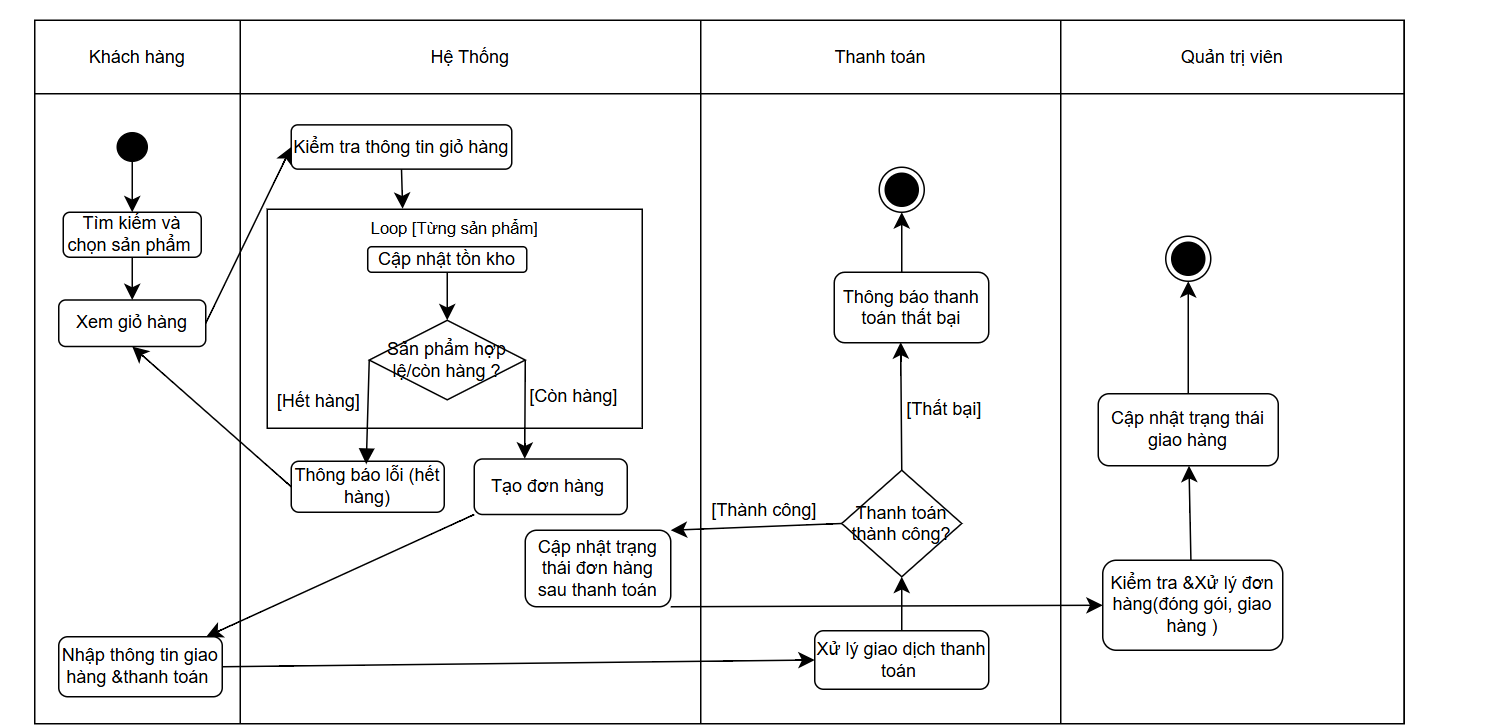
Hoàn tiền:



Hỗ trợ:



### **3.2.4. Biểu đồ hoạt động**

****

### **3.2.5. Biểu đồ trạng thái**

#### **3.2.5.1. Biểu đồ trạng thái – Đơn hàng**

A diagram of a flowchart

AI-generated content may be incorrect.

|  |  |
| --- | --- |
| Mục | Nội dung |
| Bắt đầu | |  | | --- | | Tạo đơn → **Chờ xử lý** |  |  | | --- | |  | |
| Trạng thái | |  | | --- | | Chờ xử lý → Đã xác nhận → Đang giao → (Đã hoàn thành | Đã hủy) |  |  | | --- | |  | |
| Sự kiện | |  | | --- | | Xác nhận đơn; Giao hàng; Giao thành công; Giao thất bại/Hủy giữa chừng; Hủy đơn |  |  | | --- | |  | |
| Kết thúc | |  | | --- | | **Đã hoàn thành** hoặc **Đã hủy** |  |  | | --- | |  | |
| Nguyên tắc then chốt | - Chỉ “Giao hàng” sau “Đã xác nhận”.  - Vào **Đã hoàn thành/Đã hủy** là kết thúc vòng đời. |

#### **3.2.5.2. Biểu đồ trạng thái – Thanh toán**

**A diagram of a diagram

AI-generated content may be incorrect.**

|  |  |
| --- | --- |
| Mục | Nội dung |
| Bắt đầu | |  |  |  | | --- | --- | --- | | |  | | --- | | Khi tạo đơn → **Chưa thanh toán** |  |  | | --- | |  | | |
| Trạng thái | |  |  |  | | --- | --- | --- | | |  | | --- | | Chưa thanh toán → Đang xử lý → (Đã thanh toán | Thanh toán thất bại | Hủy) |  |  | | --- | |  | |  |  | | --- | |  | |
| Sự kiện | |  |  |  | | --- | --- | --- | | |  | | --- | | Bắt đầu thanh toán; Webhook thành công; Webhook lỗi; Hủy thanh toán |  |  | | --- | |  | |  |  | | --- | |  | |
| Kết thúc | |  |  |  | | --- | --- | --- | | |  | | --- | | **Đã thanh toán** / **Thanh toán thất bại** / **Hủy** |  |  | | --- | |  | |  |  | | --- | |  | |
| Nguyên tắc then chốt | - Webhook **idempotent** theo **transactionId**.  - Chỉ set **Đã thanh toán** khi số tiền khớp đơn. |

#### **3.2.5.3. Biểu đồ trạng thái – Tồn kho/Sản phẩm**

**A diagram of a flowchart

AI-generated content may be incorrect.**

|  |  |
| --- | --- |
| Mục | Nội dung |
| Bắt đầu | |  |  |  | | --- | --- | --- | | |  | | --- | | Thêm mới → **Còn hàng** |  |  | | --- | |  | | |
| Trạng thái | |  |  |  | | --- | --- | --- | | |  | | --- | | Còn hàng ↔ Hết hàng → Ngừng kinh doanh |  |  | | --- | |  | |  |  | | --- | |  | |
| Sự kiện | |  |  |  | | --- | --- | --- | | |  | | --- | | Hết hàng; Nhập thêm hàng; Ngừng bán |  |  | | --- | |  | | |
| Kết thúc | |  |  |  | | --- | --- | --- | | |  | | --- | | **Ngừng kinh doanh** |  |  | | --- | |  | |  |  | | --- | |  | |
| Nguyên tắc then chốt | - Giảm kho bằng cập nhật **atomic/optimistic locking** để tránh oversell.  - “Ngừng kinh doanh” ⇒ ẩn khỏi catalog/khóa mua. |

#### **3.2.5.4. Biểu đồ trạng thái – Tài khoản người dùng**

**A diagram of a flowchart

AI-generated content may be incorrect.**

|  |  |
| --- | --- |
| Mục | Nội dung |
| Bắt đầu | |  |  |  |  | | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | Tạo tài khoản → **Đang hoạt động** |  |  | | --- | |  | | | |
| Trạng thái | |  |  |  |  |  | | --- | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | Đang hoạt động ↔ Khóa tạm thời → Vô hiệu hóa |  |  | | --- | |  | |  |  | | --- | |  | |  |  | | --- | |  | |
| Sự kiện | |  |  |  |  | | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | Bị khóa; Mở khóa; Vô hiệu hóa/Xóa; Xóa vĩnh viễn |  |  | | --- | |  | | | |
| Kết thúc | |  |  |  |  | | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | **Vô hiệu hóa** |  |  | | --- | |  | | |  |  | | --- | |  | |
| Nguyên tắc then chốt | **- Không ACTIVE** ⇒ chặn checkout/thanh toán. - “Khóa tạm thời” có thể mở lại. |

# **IV: TRIỂN KHAI & KIỂM THỬ**

## **4.1. Công nghệ sử dụng**

## **4.2. Kiểm thử**

# **V: TỔNG KẾT VÀ HƯỚNG PHÁT TRIỂN**

## **5.1. Đánh giá**

## **5.2. Hướng phát triển**