**ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

A blue circle with white text

AI-generated content may be incorrect.

**ĐỒ ÁN**

**KIỂM THỬ PHẦN MỀM**

**WEBSITE BÁN MÁY TÍNH**

**Giảng viên hướng dẫn:** TS.Đỗ Như Tài

**Sinh viên thực hiện**

3122411168 – Nguyễn Minh Quân

3122411037 – Đào Cư Đạt

3122411074 – Ngô Gia Hưng

3122411178 – Nguyễn Ngọc Sơn

**Thành phố Hồ Chí Minh, Tháng 9 năm 2025**

**Bảng phân công công việc**

|  |  |  |
| --- | --- | --- |
| Họ và tên | Công việc | Mức độ hoàn thành |
| Đào Cư Đạt |  | 100% |
| Nguyễn Minh Quân |  | 100% |
| Ngô Gia Hưng |  | 100% |
| Nguyễn Ngọc Sơn |  | 100% |

Lời cam đoan

Em xin cam đoan rằng đồ án môn học với đề tài kiểm thử “Website thương mại điện tử SkyWind” là kết quả nghiên cứu và thực hiện của nhóm em, dưới sự hướng dẫn của giảng viên phụ trách môn học. Toàn bộ nội dung trong đồ án này được em tìm hiểu, tổng hợp, phân tích và triển khai dựa trên kiến thức đã học cùng với các tài liệu, nguồn tham khảo có độ tin cậy.

Các số liệu, kết quả và hình ảnh trình bày trong đồ án đều trung thực, được thực hiện trong quá trình làm bài và chưa từng được sử dụng hoặc công bố trong bất kỳ công trình nào khác. Nếu phát hiện có sự sao chép, gian lận hay vi phạm bản quyền, nhóm em hoàn toàn chịu trách nhiệm trước nhà trường và giảng viên hướng dẫn.

Đồ án này được thực hiện với mục tiêu vận dụng các kiến thức về kiểm thử phần mềm, phân tích yêu cầu và kỹ thuật kiểm thử chức năng để đánh giá chất lượng của website thương mại điện tử bán laptop. Qua quá trình thực hiện, nhóm em đã cố gắng tìm hiểu, thử nghiệm các công cụ kiểm thử, phân tích kết quả và rút ra nhận xét thực tế nhằm nâng cao hiểu biết về quy trình đảm bảo chất lượng phần mềm.

Em xin gửi lời cảm ơn chân thành đến giảng viên hướng dẫn TS. Đỗ Như Tài trong bộ môn đã tận tình giúp đỡ, tạo điều kiện thuận lợi để em hoàn thành đồ án này.

Mục lục

[Lời mở đầu 8](#_Toc217446608)

[Chương 1. Tổng quan đề tài 9](#_Toc217446609)

[**1.** **Mô tả dự án** 9](#_Toc217446610)

[**2. Khảo sát hiện trạng** 9](#_Toc217446611)

[**2.1. Các quy trình nghiệp vụ** 9](#_Toc217446612)

[**3.** **Công nghệ sử dụng** 11](#_Toc217446613)

[**4.** **Yêu cầu người dùng** 12](#_Toc217446614)

[**4.1. Yêu cầu chức năng** 12](#_Toc217446615)

[**4.2. Yêu cầu phi chức năng** 14](#_Toc217446616)

[**5.** **Kế hoạch triển khai** 15](#_Toc217446617)

[**5.1. Phạm vi** 15](#_Toc217446618)

[**5.2. Chiến lược kiểm thử** 15](#_Toc217446619)

[**6.** **Mục tiêu** 16](#_Toc217446620)

[**Chương 2. Thiết kế phần mềm và kiến trúc** 18](#_Toc217446621)

[**1. Thiết kế phần mềm** 18](#_Toc217446622)

[**1.1. Bối cảnh kinh doanh** 18](#_Toc217446623)

[**1.2. Sơ đồ quy trình nghiệp vụ** 18](#_Toc217446624)

[1.3. Các tác nhân trong hệ thống 22](#_Toc217446625)

[**1.4. Sơ đồ ngữ cảnh** 23](#_Toc217446626)

[**1.5. Use case** 24](#_Toc217446627)

[**1.6. User stories** 43](#_Toc217446628)

[**1.7. Domain driven design** 45](#_Toc217446629)

[**1.8. Data model** 47](#_Toc217446630)

[**1.9. Wireflow** 50](#_Toc217446631)

[**1.10. Giao diện** 50](#_Toc217446632)

[2. Thiết kế kiến trúc 55](#_Toc217446633)

[**2.1. Sơ đồ khối** 55](#_Toc217446634)

[**2.2. Kiến trúc C4** 55](#_Toc217446635)

[**2.3. Sơ đồ luồng giao tiếp** 59](#_Toc217446636)

[**2.4. Sơ đồ triển khai** 60](#_Toc217446637)

[Chương 3. Kế hoach kiểm thử 61](#_Toc217446638)

[1. Giới thiệu 61](#_Toc217446639)

[**1.1. Mục đích** 61](#_Toc217446640)

[**1.2. Phạm vi** 61](#_Toc217446641)

[2. Hạng mục được kiểm thử 61](#_Toc217446642)

[**2.1. Chức năng** 61](#_Toc217446643)

[**2.2. Khả năng sử dụng** 63](#_Toc217446644)

[**2.3. Tương thích** 63](#_Toc217446645)

[**2.4. Giao diện** 63](#_Toc217446646)

[**2.5. Bảo mật** 63](#_Toc217446647)

[**2.6. Cơ sở dữ liệu** 63](#_Toc217446648)

[**2.7. Hồi quy** 64](#_Toc217446649)

[**2.8. API** 64](#_Toc217446650)

[**2.9. Kiểm thử xử lý lỗi và dữ liệu biên** 64](#_Toc217446651)

[**2.10. Kiểm thử hiệu năng (Performance Testing – mức cơ bản)** 65](#_Toc217446652)

[3. Hạng mục không được kiểm thử 65](#_Toc217446653)

[**3.1. Các chức năng thuộc hệ thống bên thứ ba** 65](#_Toc217446654)

[**3.2. Kiểm thử bảo mật nâng cao** 65](#_Toc217446655)

[**3.3. Kiểm thử khả năng sử dụng chuyên sâu** 65](#_Toc217446656)

[4. Tiêu chí kiểm thử chấp nhận 66](#_Toc217446657)

[**4.1. Độ bao phủ kiểm thử** 66](#_Toc217446658)

[**4.2. Tỷ lệ trường hợp kiểm thử đạt** 66](#_Toc217446659)

[**4.3. Số lượng trường hợp kiểm thử** 66](#_Toc217446660)

[**4.4. Số lượng lỗi** 66](#_Toc217446661)

[**4.5. Mật độ kiểm thử đơn vị** 66](#_Toc217446662)

[**4.6. Độ bao phủ mã nguồn** 66](#_Toc217446663)

[**4.7. Quy trình CI/CD** 67](#_Toc217446664)

[5. Chiến lược kiểm thử 67](#_Toc217446665)

[**5.1. Phương pháp kiểm thử** 67](#_Toc217446666)

[**5.2. Loại kiểm thử** 67](#_Toc217446667)

[**5.3. Cấp độ kiểm thử** 68](#_Toc217446668)

[6. Tài nguyên 69](#_Toc217446669)

[**6.1. Nguồn nhân lực** 69](#_Toc217446670)

[**6.2. Lịch trình kiểm thử** 70](#_Toc217446671)

[7. Môi trường 71](#_Toc217446672)

[**7.1. Phần cứng** 71](#_Toc217446673)

[**7.2. Phần mềm** 72](#_Toc217446674)

[**7.3. Hạ tầng** 72](#_Toc217446675)

[8. Tài liệu bàn giao 72](#_Toc217446676)

[Chương 4. Thiết kế kiểm thử 74](#_Toc217446677)

[**1. Giới thiệu** 74](#_Toc217446678)

[**2. Quy trình thiết kế kiểm thử theo V-model** 74](#_Toc217446679)

[**2.1. Phân tích yêu cầu – 1a** 74](#_Toc217446680)

[**2.2. Thiết kế hệ thống – 2a** 74](#_Toc217446681)

[**2.3. Thiết kế kiến trúc – 3a** 75](#_Toc217446682)

[**2.4. Thiết kế module – 4a** 75](#_Toc217446683)

[**2.5. Kiểm thử đơn vị - 1b** 75](#_Toc217446684)

[**2.6. Kiểm thử tích hợp – 2b** 76](#_Toc217446685)

[**2.7. Kiểm thử hệ thống – 3b** 77](#_Toc217446686)

[**2.8. Kiểm thử chấp nhận – 4b** 77](#_Toc217446687)

[**3. Kỹ thuật thiết kế kiểm thử** 78](#_Toc217446688)

[**3.1. Kiểm thử hộp đen** 78](#_Toc217446689)

[**3.2. Kiểm thử hộp trắng** 78](#_Toc217446690)

[**4. Phương pháp thiết kế kiểm thử** 78](#_Toc217446691)

[**4.1. Kiểm thử thủ công** 78](#_Toc217446692)

[**4.2. Kiển thử tự động** 78](#_Toc217446693)

[**5. Áp dụng GenAI trong việc sinh Test Case** 79](#_Toc217446694)

[**5.1. Đầu vào GenAI** 79](#_Toc217446695)

[**5.2. Giai đoạn xử lý của GenAI** 79](#_Toc217446696)

[**5.3. Hoàn thiện test case** 79](#_Toc217446697)

[Chương 5. Báo cáo kiểm thử 80](#_Toc217446698)

[**1. Tổng quan về quá trình kiểm thử** 80](#_Toc217446699)

[**2. Báo cáo trường hợp kiểm thử** 80](#_Toc217446700)

[**2.1. Giới thiệu** 80](#_Toc217446701)

[**2.2. Phạm vi bao phủ** 80](#_Toc217446702)

[**2.3. Kết quả thực thi** 82](#_Toc217446703)

[**3. Báo cáo lỗi** 83](#_Toc217446704)

[**3.1. Phân loại lỗi theo mức độ nghiêm trọng** 83](#_Toc217446705)

[**3.2. Thống kê lỗi** 83](#_Toc217446706)

[**3.3. Quá trình xử lý lỗi** 84](#_Toc217446707)

# Lời mở đầu

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ như hiện nay, các hoạt động kinh doanh, buôn bán trực tuyến ngày càng trở nên phổ biến và đóng vai trò quan trọng trong đời sống hiện đại. Thương mại điện tử đã mang lại nhiều tiện ích vượt trội, giúp người tiêu dùng có thể dễ dàng tìm kiếm, mua sắm và thanh toán chỉ với vài thao tác đơn giản. Websit**e SkyWind** được xây dựng với mục tiêu trở thành một ứng dụng thương mại điện tử chuyên cung cấp và phân phối các dòng laptop uy tín, chất lượng trên thị trường. Hệ thống cho phép người dùng dễ dàng tìm kiếm, so sánh và lựa chọn sản phẩm phù hợp với nhu cầu học tập, làm việc và giải trí.

Tuy nhiên, để một website thương mại điện tử hoạt động ổn định, hiệu quả và mang lại trải nghiệm tốt cho người dùng, việc kiểm thử phần mềm đóng vai trò vô cùng quan trọng. Kiểm thử giúp phát hiện và khắc phục lỗi, đảm bảo các chức năng của hệ thống vận hành đúng yêu cầu, từ đó nâng cao chất lượng và độ tin cậy của sản phẩm phần mềm.

Xuất phát từ tầm quan trọng đó, em chọn thực hiện đề tài “Kiểm thử website thương mại điện tử SkyWind” với mục tiêu tìm hiểu, áp dụng các kỹ thuật và công cụ kiểm thử để đánh giá chất lượng website. Đề tài tập trung vào việc phân tích yêu cầu, xây dựng các trường hợp kiểm thử, thực hiện kiểm thử chức năng và ghi nhận kết quả kiểm thử.

Thông qua quá trình thực hiện đồ án, em mong muốn củng cố và mở rộng kiến thức về kiểm thử phần mềm, đồng thời rèn luyện kỹ năng làm việc thực tế, góp phần nâng cao năng lực chuyên môn trong lĩnh vực phát triển và đảm bảo chất lượng phần mềm.

Em xin chân thành cảm ơn thầy TS. Đỗ Như Tài đã tận tình hướng dẫn và tạo điều kiện giúp em hoàn thành đồ án này.

# Chương 1. Tổng quan đề tài

# **Mô tả dự án**

Dự án kiểm thử “Website thương mại điện tử SkyWind” được thực hiện nhằm đánh giá mức độ đáp ứng các yêu cầu chức năng của hệ thống và phát hiện các lỗi phát sinh trong quá trình phát triển. Hoạt động kiểm thử tập trung vào việc xác minh tính đúng đắn của dữ liệu, sự ổn định trong quá trình vận hành và tính chính xác của các chức năng nghiệp vụ chính của website.

Quá trình kiểm thử bao gồm việc phân tích yêu cầu, xây dựng và thực thi các trường hợp kiểm thử, thực hiện kiểm thử đơn vị, kiểm thử tích hợp, kiểm thử hệ thống và kiểm thử chấp nhận nhằm đảm bảo hệ thống hoạt động đúng theo đặc tả trước khi triển khai sử dụng.

# **2. Khảo sát hiện trạng**

## **2.1. Các quy trình nghiệp vụ**

**Quy trình xử lý giỏ hàng**

Sau khi đăng nhập và đã chọn sản phẩm mong muốn, khách hàng có thể thêm sách vào giỏ hàng. Tại giao diện giỏ hàng, hệ thống hiển thị danh sách các laptop đã thêm cùng hình ảnh, tiêu đề, giá bán, giảm giá, số lượng mua. Khách hàng có thể điều chỉnh số lượng hoặc xóa sách khỏi giỏ hàng bất kỳ lúc nào. Mỗi thay đổi sẽ được hệ thống tự động cập nhật lại tổng tiền, giúp người dùng dễ dàng kiểm soát chi phí trước khi tiến hành đặt hàng và thanh toán.

**Quy trình đặt hàng và thanh toán**

Sau khi khách hàng thêm sách vào giỏ hàng, hệ thống chuyển sang bước đặt hàng và yêu cầu cung cấp thông tin giao hàng (họ tên, số điện thoại, địa chỉ) và chọn phương thức thanh toán: COD hoặc Momo/Ngân hàng.

Nếu khách hàng chọn thanh toán qua Momo/Ngân hàng, hệ thống sẽ tạm thời lưu đơn hàng. Hệ thống sẽ tạo mã QR thanh toán và hiển thị cho khách hàng quét để hoàn tất thanh toán.

Khi hệ thống nhận được phản hồi từ Momo/Ngân hàng, nếu thanh toán thành công và số lượng laptop hiện có đủ, hệ thống sẽ tự động cập nhật trạng thái đơn hàng và giảm số lượng hiện có của các sản phẩm tương ứng, sau đó thông báo đặt hàng thành công.

Ngược lại, nếu thanh toán thành công nhưng số lượng hiện có không đủ, hệ thống sẽ tự động hoàn tiền cho khách hàng, hiển thị thông báo đặt hàng thất bại và nguyên nhân, đồng thời xóa đơn hàng để đảm bảo dữ liệu chính xác.

**Quy trình quản trị hệ thống**

Trang quản trị hệ thống được thiết kế dành cho quản trị viên hệ thống và nhân viên bán hàng, giúp theo dõi và điều hành hoạt động của website. Trong đó, quản trị viên hệ thống có toàn quyền thưc hiện tất cả chức năng và nhân viên bán hàng bị giới hạn quyền, không được thực hiện chức năng quản lý người dùng. Các chức năng chính bao gồm:

Quản lý laptop: thêm mới, xóa hoặc cập nhật thông tin sách (tên laptop, CPU, GPU, RAM, thông số laptop, mô tả, ảnh sản phẩm, số lượng hiện có) để khách hàng có thể xem sách mong muốn mua.

Quản lý danh mục, thành phần của laptop: thêm, cập nhật hoặc xóa danh mục laptop, thành phần chính. Đảm bảo thông tin được tổ chức hợp lý, hỗ trợ khách hàng dễ dàng tìm kiếm sách.

Quản lý đơn hàng: quản trị viên hệ thống và nhân viên bán hàng có thể theo dõi danh sách đơn hàng, cập nhật trạng thái đơn hàng thủ công. Với đơn hàng được thanh toán bằng Momo/Ngân hàng, nếu trạng thái đơn hàng bị hủy hoặc trả hàng sẽ gọi đến Momo/Ngân hàng để hoàn tiền cho khách hàng.

Quản lý thanh toán: hiển thị danh sách các giao dịch thanh toán thành công và hoàn tiền của khách hàng qua cổng thanh toán Momo/Ngân hàng.

Quản lý người dùng: chỉ có quản trị viên hệ thống mới thêm, xóa, cập nhật hoặc khóa tài khoản quản trị viên và khách hàng.

# **Công nghệ sử dụng**

Website SkyWind được phát triển dựa trên mô hình Client–Server, ứng dụng các công nghệ hiện đại nhằm đảm bảo hiệu năng, bảo mật và khả năng mở rộng cao trong môi trường thương mại điện tử.

* 1. Frontend (Giao diện người dùng)
* Ngôn ngữ: HTML5, CSS3, JavaScript (ES6+).
* Framework: ReactJS – giúp xây dựng giao diện linh hoạt, tốc độ cao và dễ bảo trì.
* Thư viện hỗ trợ:
  + Axios – gọi API từ backend.
  + React Router DOM – điều hướng giữa các trang mà không cần tải lại toàn bộ website.
  + TailwindCSS / Bootstrap – hỗ trợ thiết kế giao diện hiện đại, tương thích đa nền tảng.
* Công cụ build: Vite / Webpack – tối ưu quá trình biên dịch và triển khai mã nguồn frontend.
  1. Backend (Xử lý nghiệp vụ)
* Ngôn ngữ lập trình: Python.
* Framework: FastAPI (hoặc Django REST Framework, nếu bạn đang dùng DRF).
  + Cung cấp API RESTful, tốc độ cao, dễ mở rộng và dễ tích hợp với frontend.
* Chức năng chính:
  + Quản lý người dùng, sản phẩm, giỏ hàng và đơn hàng.
  + Xử lý thanh toán, xác thực và phân quyền người dùng.
  + Gửi và nhận dữ liệu qua API an toàn, tuân thủ chuẩn HTTP/HTTPS.
  1. Database (Cơ sở dữ liệu)
* Hệ quản trị cơ sở dữ liệu:
  + PostgreSQL hoặc MySQL – dùng cho dữ liệu quan hệ.
  + Hoặc MongoDB – nếu cần lưu trữ dữ liệu linh hoạt dạng NoSQL.
* ORM:
  + SQLAlchemy (với FastAPI) hoặc Django ORM (với Django REST Framework) – hỗ trợ thao tác cơ sở dữ liệu hiệu quả và an toàn.
  1. Authentication & Security
* JWT (JSON Web Token) – xác thực người dùng và quản lý phiên đăng nhập.
* OAuth2 – hỗ trợ đăng nhập an toàn và tích hợp bên thứ ba.
* Passlib / BCrypt – mã hóa mật khẩu người dùng.
* CORS Middleware & HTTPS – bảo mật dữ liệu trao đổi giữa client và server.
  1. Tools & DevOps
* Git / GitHub – quản lý mã nguồn và làm việc nhóm.
* Postman – kiểm thử API trong quá trình phát triển.
* Docker (tuỳ chọn) – đóng gói ứng dụng để triển khai dễ dàng.
* Render / Vercel / Railway / AWS – triển khai backend và frontend trực tuyến.
  1. Testing & Monitoring
* Pytest / Unittest – kiểm thử chức năng backend.
* Google Analytics / Firebase – theo dõi hành vi và trải nghiệm người dùng.

# **Yêu cầu người dùng**

## **4.1. Yêu cầu chức năng**

|  |  |  |
| --- | --- | --- |
| **Mã yêu cầu** | **Người yêu cầu** | **Nội dung yêu cầu** |
| **BR1** | **Quản lý sản phẩm** | |
| BR1.1 | Khách hàng | Tìm kiếm, lọc và sắp xếp sản phẩm theo các tiêu chí như giá, tiêu đề, danh mục, CPU, RAM, GPU, Bộ nhớ, màn hình  Xem thông tin chi tiết của từng laptop |
| BR1.2 | quản trị viên hệ thống | Quản lý laptop như thêm, xóa hoặc chỉnh sửa laptop  Quản lý thành phần như thêm, xóa hoặc chỉnh CPU, GPU, RAM, bộ nhớ, thông số màn hình  Quản lý danh mục như thêm, xóa hoặc chỉnh sửa danh mục |
| **BR2** | **Quản lý giỏ hàng** | |
| BR2.1 | Khách hàng | Thêm sản phẩm vào giỏ hàng  Xem danh sách các sản phẩm đã thêm vào giỏ hàng  Chỉnh sửa số lượng sản phẩm hoặc xóa sản phẩm khỏi giỏ hàng |
| **BR3** | **Quản lý thanh toán** | |
| BR3.1 | Khách hàng | Nhận thông báo kết quả thanh toán (thành công hoặc thất bại) |
| BR3.2 | quản trị viên hệ thống | Theo dõi danh sách các giao dịch thanh toán thành công hoặc đã hoàn tiền |
| **BR4** | **Quản lý đơn hàng** | |
| BR4.1 | Khách hàng | Xem danh sách đơn hàng đã đặt và chi tiết từng đơn hàng  Đặt đơn hàng mới |
| BR4.2 | quản trị viên hệ thống | Cập nhật trạng thái đơn hàng thủ công  Theo dõi tất cả danh sách đơn hàng |
| **BR5** | **Kiểm soát truy cập** | |
| BR5.1 | Quản trị viên hệ thống | Thêm, xóa, cập nhật hoặc khóa tài khoản tất cả của người dùng trong hệ thống (không được khóa hoặc xóa chính tài khoản của mình) |
| BR5.1 | Khách hàng, quản trị viên hệ thống | Đăng nhập, đăng ký hoặc đăng xuất |

## **4.2. Yêu cầu phi chức năng**

|  |  |
| --- | --- |
| **BR6** | **Triển khai hệ thống** |
| BR6.1 | Phần mềm hoạt động trên môi trường Internet, cho phép nhiều người dùng cùng truy cập vào sử dụng cùng một lúc |
| **BR7** | **Giao diện** |
| BR7.1 | Giao diện được trình bày khoa học, thẩm mỹ và hài hòa phù hợp với mục đích của phần mềm, tuần thủ các chuẩn về web |
| **BR8** | **Bảo mật** |
| BR8.1 | Toàn bộ các dữ liệu đều có ràng buộc toàn vẹn và dữ liệu nhạy cảm phải được mã hóa |

# **Kế hoạch triển khai**

## **5.1. Phạm vi**

Tài liệu này là cơ sở để kiểm thử, nghiệm thu và đánh giá dự án “Website thương mại điện tử SkyWind”. Kiểm tra các chức năng và tính năng của website, giao diện và tính hợp lệ của dữ liệu và cơ sở dữ liệu.

## **5.2. Chiến lược kiểm thử**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Người thực hiện** | **Vai trò** | **Công việc cụ thể** |
| 1 | Nguyễn Minh Quân | Kiểm thử viên/Lập trình viên | Lập kế hoạch kiểm thử  Thực hiện kiểm thử hệ thống |
| 2 | Đào Cư Đạt | Kiểm thử viên/Lập trình viên | Thiết kế Test Case  Thực hiện kiểm thử đơn vị và kiểm thử tích hợp |
| 3 | Ngô Gia Hưng | Kiểm thử viên/ Lập trình viên | Viết mã CI/CD trong GitHub Actions  Ghi chép tài kiểm thử |
| 4 | Nguyễn Ngọc Sơn | Kiểm thử viên/ Lập trình viên | Lập review checklist  Ghi chép tài liệu kiểm thử |

**Môi trường và công cụ**

|  |  |
| --- | --- |
| **Mục đích** | **Thành phần** |
| Ghi chép tài liệu kiểm thử | Microsoft Word, Microsoft Excel |
| Môi trường trình duyệt | Chrome, Firefox, Safari |
| Phát triển Frontend | Vite, ReactJS, JavaScript |
| Phát triển Backend | Python |
| Quản lý mã nguồn | GitHub |
| CI/CD | GitHub Actions |
| Triển khai trên môi trường Production | Frontend và Backend sẽ deploy trên Render  Database PostgreSQL sẽ host trên Aiven |

# **Mục tiêu**

Mục tiêu của hoạt động kiểm thử website SkyWind là xác nhận rằng hệ thống hoạt động đúng theo các yêu cầu đã được đặc tả, phát hiện sớm các lỗi trong quá trình phát triển và đảm bảo chất lượng phần mềm trước khi triển khai.

Quá trình kiểm thử tập trung vào việc kiểm tra các chức năng nghiệp vụ chính, tính ổn định cơ bản của hệ thống và khả năng tương thích trên các trình duyệt, thiết bị phổ biến.

Kết quả kiểm thử sẽ là cơ sở để đánh giá mức độ hoàn thiện của hệ thống và hỗ trợ cho việc triển khai, vận hành và bảo trì phần mềm.

# **Chương 2. Thiết kế phần mềm và kiến trúc**

# **1. Thiết kế phần mềm**

## **1.1. Bối cảnh kinh doanh**

Website SkyWind được xây dựng nhằm đáp ứng nhu cầu mua sắm laptop trực tuyến của người dùng, với các chức năng cốt lõi Danh mục sản phẩm (Product Catalog), Giỏ hàng (Shopping Cart), Quy trình thanh toán (Payment Process), Quản lý sản phẩm (Inventory), Đánh giá sản phẩm (Rating) và Kiểm soát truy cập (Access Control).

Với Product Catalog, khách hàng có thể duyệt qua danh sách laptop với các bộ lọc theo hãng, giá, cấu hình (CPU, RAM, ổ cứng, card đồ họa, …). Khi chọn vào sản phẩm, khách hàng sẽ xem được thông tin chi tiết như tên, hình ảnh, thông số kỹ thuật, tình trạng tồn kho và đánh giá từ người mua khác (nếu có). Quản trị viên (SysAdmin) có quyền thêm, chỉnh sửa hoặc xóa sản phẩm, đồng thời gán sản phẩm vào kho phù hợp.

Với Shopping Cart, Người dùng có thể thêm laptop vào giỏ hàng hoặc mua ngay. Tại trang giỏ hàng, khách hàng sẽ thấy danh sách sản phẩm đã chọn cùng bảng tóm tắt gồm tên sản phẩm, giá sản phẩm, số lượng, giá trị đơn hàng, khuyến mãi, phí vận chuyển và tổng tiền cần thanh toán. Khi thêm hoặc xóa sản phẩm, giỏ hàng sẽ được cập nhật tự động. Sau đó, họ có thể thực hiện quy trình thanh toán bằng cách nhấn nút Thanh Toán. SysAdmin có thể theo dõi và xử lý các giỏ hàng bất thường để đảm bảo tính hợp lệ.

Với Payment Process, Sau khi khách hàng tiến hành đặt hàng (nhấn nút Thanh Toán), hệ thống sẽ xác thực thông tin sản phẩm, xử lý thanh toán trực tuyến và gửi email xác nhận kèm hóa đơn điện tử cho khách hàng.

Với Access Control, Người dùng cần đăng nhập để mua hàng, quản lý đơn hàng và đánh giá sản phẩm. SysAdmin khi đăng nhập sẽ truy cập vào trang quản trị để quản lý sản phẩm, đơn hàng và khách hàng. (Khách hàng và SysAdmin đăng nhập ở hai trang khác nhau)

Ngoài ra, một số công việc khởi tạo cần thiết khi triển khai hệ thống như: tạo tài khoản quản trị, tài khoản khách hàng mẫu, và nhập dữ liệu sản phẩm, tồn kho và đánh giá thử nghiệm.

## **1.2. Sơ đồ quy trình nghiệp vụ**

**Quy trình xử lý giỏ hàng**

Sau khi đăng nhập và tìm/chọn laptop, khách hàng có thể thêm sản phẩm vào giỏ hàng. Tại giao diện giỏ hàng, khách hàng có thể kiểm tra danh sách sản phẩm. Họ có thể điều chỉnh số lượng hoặc xóa laptop khỏi giỏ hàng. Sau đó, khách hàng sẽ tiến hành đặt hàng.

**A diagram of a flowchart

AI-generated content may be incorrect.**

Hình 2.0.1Sơ đồ quy trình xử lý giỏ hàng

**Quy trình đặt hàng và thanh toán**

Sau khi nhập thông tin giao hàng, khách hàng chọn phương thức thanh toán COD hoặc Momo/Ngân hàng.

Thanh toán COD thì khách hàng chuyển thẳng đến bước kiểm tra số lượng sản phẩm hiện có. Nếu đủ số lượng thì đơn hàng đặt hàng thành công.

Thanh toán Momo/Ngân hàng thì hệ thống sẽ tạo mã QR để khách hàng quét.

* Nếu thanh toán thất bại, khách hàng quay lại chọn lại phương thức thanh toán.
* Nếu thanh toán thành công, hệ thống tiếp tục kiểm tra số lượng sản phẩm mua. Nếu đủ, đơn hàng đặt hàng thành công. Nếu không đủ, hệ thống tự động gọi đến Momo/Ngân hàng để hoàn tiền.

A diagram of a flowchart

AI-generated content may be incorrect.

Hình 2.0.2Sơ đồ quy trình đặt hàng và thanh toán

**Quy trình quản trị hệ thống**

Sau khi đăng nhập, quản trị viên hệ thống và nhân viên bán hàng có thể thực hiện các chức năng quản lý danh mục, sản phẩm, thành phần, đơn hàng và thanh toán. Riêng chức năng quản lý người dùng chỉ được phép thực hiện bởi quản trị viên hệ thống.

Trong quá trình quản lý đơn hàng, nếu cập nhật trạng thái đơn hàng thanh toán bằng Momo thành hủy hoặc trả hàng, hệ thống sẽ tự động gửi yêu cầu hoàn tiền đến Momo để xử lý giao dịch hoàn trả.

A diagram with white text and black lines

AI-generated content may be incorrect.

Hình 2.0.3Sơ đồ quy trình quản trị hệ thống

## 1.3. Các tác nhân trong hệ thống

|  |  |  |
| --- | --- | --- |
| Actor | Vai trò | Quyền hạn chính |
| Buyer (Người mua) | Người sử dụng chính của website | Xem, tìm kiếm, mua hàng, thanh toán, đánh giá sản phẩm |
| System Admin (Quản trị viên) | Quản lý hệ thống | Thực hiện tất cả chức năng quản trị |
| Momo/Ngân hàng | Cổng thanh toán bên thứ 3 | Hiển thị QR code cho khách hàng quét mã thanh toán |

A group of people with white text

AI-generated content may be incorrect.

Hình 2.0.4 Sơ đồ các tác nhân tham gia hệ thống

## **1.4. Sơ đồ ngữ cảnh**

Khách hàng: Tương tác trực tiếp với SkyWind để nhận danh sách sản phẩm, đăng nhập, quản lý giỏ hàng, đặt hàng, thanh toán và xem danh sách đơn hàng.

Momo: Cổng thanh toán bên ngoài, tương tác với SkyWind để tạo mã QR, hoàn tiền và nhận kết quả giao dịch.

Quản trị viên hệ thống: Quản lý hệ thống qua các chức năng như quản lý sản phẩm, danh mục, đơn hàng, thanh toán và quản lý người dùng.

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.5Sơ đồ ngữ cảnh hệ thống

## **1.5. Use case**

**Use case summary**

Khách hàng thực hiện chức năng: quản lý giỏ hàng, quản lý sản phẩm, kiểm soát truy cập.

Quản trị viên hệ thống thực chức năng: quản lý sản phẩm, quản lý thanh toán, quản lý đơn hàng, kiểm soát truy cập.

Momo/Ngân hàng thực hiện chức năng: quản lý thanh toán.

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.6 Sơ đồ use case summary

**UC1 Quản lý sản phẩm**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| **Use Case Number:** | **UC1** | | |
| **Use Case Name:** | Quản lý sản phẩm | | |
| **Actor (s):** | Admin | | |
| **Maturity:** | Focused | | |
| **Summary:** | Admin quản lí danh mục sản phẩm, với các chức năng thêm, sửa, xóa | | |
| **Basic Course of Events:** | **Actor Action** | **System Response** | |
|  | 1. Actor đăng nhập vào hệ thống |  | |
|  | 2. Hệ thống xác nhận tài khoản và hiển thị giao diện chức năng tương ứng | |
| 3. Admin chọn “Products” |  | |
|  | 4. Hệ thống cập nhật thao tác | |
|  | 5. Admin thực hiện thao tác thêm, sửa, xóa sản phẩm bất kì **E1** |  | |
|  |  | 6. Hệ thống hiển thị kết quả cập nhật | |
|  |  | 7. Hệ thống đồng bộ với frontend | |
| **Alternative Paths:** | **A1**. Admin cập nhật thông tin sản phẩm như giá, mô tả, hệ thống sẽ cập nhật và hiển thị lên frontend | | |
| **Exception Paths:** | **E1.** Admin nhập dữ liệu sai định dạng hoặc thiếu thông tin, format hiển thị thông báo lỗi và yêu cầu chỉnh sửa | | |
| **Extension Points:** | Không | | |
| **Triggers:** | Không | | |
| **Assumptions** | Không | | |
| **Preconditions** | Không | | |
| **Post Conditions** | Các thay đổi lưu thành công  Frontend hiển thị thông tin mới | | |
| **Authors** |  | | |
| **Activity Diagram** | | |  |
| **Sequence** | | |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| **Use Case Number:** | **UC1** | | |
| **Use Case Name:** | Danh mục sản phẩm | | |
| **Actor (s):** | Khách hàng | | |
| **Maturity:** | Focused | | |
| **Summary:** | Khách hàng tìm kiếm sản phẩm, tìm kiến lọc, xem chi tiết sản phẩm | | |
| **Basic Course of Events:** | **Actor Action** | **System Response** | |
|  | 1. Actor đăng nhập vào hệ thống |  | |
|  | 2. Hệ thống xác nhận tài khoản và hiển thị giao diện chức năng tương ứng | |
| 3. Khách hàng tìm kiếm lọc sản phẩm |  | |
|  | 4. Hệ thống cập nhật thao tác | |
|  | 5. Khách hàng xem chi tiết sản phẩm được chọn **E1** |  | |
|  |  | 6. Hệ thống hiển thị kết quả chi tiết sản phẩm | |
| **Alternative Paths:** |  | | |
| **Exception Paths:** |  | | |
| **Extension Points:** | Không | | |
| **Triggers:** | Không | | |
| **Assumptions** | Không | | |
| **Preconditions** | Không | | |
| **Post Conditions** |  | | |
| **Authors** |  | | |
| **Activity Diagram** | | |  |
| **Sequence** | | |  |

**UC2 Quản lý giỏ hàng**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| **Use Case Number:** | **UC2** | | |
| **Use Case Name:** | Quản Lý Giỏ Hàng | | |
| **Actor (s):** | Khách hàng | | |
| **Maturity:** | Focused | | |
| **Summary:** | Buyer thêm sản phẩm vào giỏ hàng, xác nhận đơn và thực hiện thanh toán trực tuyến | | |
| **Basic Course of Events:** | **Actor Action** | **System Response** | |
|  | 1. Buyer vào trang sản phẩm |  | |
|  | 2. Hệ thống hiển thị sản phẩm của cửa hàng | |
| 3. Buyer xem chi tiết sản phầm |  | |
|  | 4. Hệ thống hiển thị chi tiết sản phẩm | |
|  | 5. Buyer thêm sản phầm vào giỏ hàng |  | |
|  |  | 6. Hệ thống hiển thị cập nhật giỏ hàng | |
|  | 7. Buyer kiểm tra thông tin sản phẩm trong giỏ hàng **A1** |  | |
|  |  | 8. Hệ thống hiển thị số lượng sản phẩm | |
|  | 9. Buyer bấm “Thanh Toán” |  | |
|  |  | 10. Hệ thống chuyển sang trang “Thanh Toán”**E1** | |
| **Alternative Paths:** | **A1**. Buyer cập nhật số lượng sản phẩm -> Hệ thống cập nhật số lượng sản phẩm | | |
| **Exception Paths:** | **E1.** Hệ thống kiểm tra số lượng sản phẩm xem có vượt quá số lượt tồn kho hay không -> hệ thống cảnh báo số lượng vượt quá giới hạn | | |
| **Extension Points:** | Không | | |
| **Triggers:** | Không | | |
| **Assumptions** | Không | | |
| **Preconditions** | Người dùng đã đăng nhập ứng dụng  Giỏ hàng không trống | | |
| **Post Conditions** | Đơn hàng đã được xác nhận và thanh toán thành công  Hệ thống lưu thông tin đơn hàng, trạng thái và thanh toán  Hiển thị trạng thái “Đã hoàn tất” trên ứng dụng | | |
| **Authors** |  | | |
| **Activity Diagram** | | |
| **Sequence** | | |

**UC3 Quản lý đơn hàng**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| **Use Case Number:** | UC3 | | |
| **Use Case Name:** | Quản lý Đơn Hàng | | |
| **Actor (s):** | Admin | | |
| **Maturity:** | Focused | | |
| **Summary:** | Admin xem và xác nhận đơn hàng , cập nhật trạng thái | | |
| **Basic Course of Events:** | Actor Action | System Response | |
|  | 1. Admin nhận thông báo đơn hàng |  | |
|  | 2. Hệ thống hiển thị danh sách đơn hàng | |
| 3. Admin xác nhận và cập nhật trạng thái đơn hàng A1 |  | |
|  | 4. Hệ thống cập nhật trạng thái đơn hàng tương ứng E1 | |
|  |  | 5. Hệ thống ghi nhận trạng thái, gửi thông báo đến khách hàng | |
| **Alternative Paths:** | A1. Admin hủy đơn | | |
| **Exception Paths:** | E1. Đơn hàng bị hủy thông báo cho Buyer ở phần lịch sử đơn hàng | | |
| **Extension Points:** | Không | | |
| **Triggers:** | Không | | |
| **Assumptions** | Không | | |
| **Preconditions** | Đơn hàng được xác nhận  Hệ thống cập nhật trạng thái theo tiến độ  Giao diện Buyer thấy trạng thái thực tế của đơn hàng | | |
| **Post Conditions** | Admin phải xác nhận đơn và cập nhật trạng thái đơn hàng  Hệ thống đồng bộ trạng thái đơn hàng theo thời gian | | |
| **Authors** |  | | |
| **Activity Diagram** | | |
| **Sequence** | | |

**UC4 Quản lý thanh toán**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| **Use Case Number:** | **UC4** | | |
| **Use Case Name:** | **Thanh Toán** | | |
| **Actor (s):** | **Buyer** | | |
| **Maturity:** | **Focused** | | |
| **Summary:** | **Buyer thực hiện điền thông tin và chọn phương thức khi thanh toán** | | |
| **Basic Course of Events:** | **Actor Action** | **System Response** | |
|  | **1.Buyer bấm “Thanh toán”, hoặc “Mua Ngay” từ trang chi tiết sản phẩm** |  | |
|  | **2. Hệ thống hiển thị trang thanh toán** | |
|  | **3.Hệ thống cập nhật thông tin địa chỉ mặc định của Buyer A1** | |
| **4. Buyer kiểm tra thông tin và địa chỉ nhận hàng A2** |  | |
|  | **5. Hệ thống cập nhật thông tin** | |
|  | **6. Buyer chọn phương thức thanh toán** |  | |
|  |  | **7. Hệ thống cập nhật phương thức thanh toán** | |
|  | **8. Buyer bấm “Thanh Toán Ngay”** |  | |
|  |  | **9. Hệ thống thông báo thanh toán thành công E1** | |
| **Alternative Paths:** | **A1. Hệ thống sẽ lấy thông tin địa chỉ của người dùng có sẵn**  **A2. Buyer thay đổi thông tin địa chỉ khác -> hệ thông cập nhật thông tin mới** | | |
| **Exception Paths:** | **E1.Buyer không chọn phương thức thanh toán sẽ thông báo** | | |
| **Extension Points:** | **Không** | | |
| **Triggers:** | **Không** | | |
| **Assumptions** | **Không** | | |
| **Preconditions** | **Không** | | |
| **Post Conditions** | **Đơn hàng được giao thành công hay sự cố đều được phản hồi sự cố rõ ràng**  **Admin có quyền phê duyệt, điều chỉnh**  **Hệ thống đảm bảo dữ liệu trạng thái đồng bộ liên tục giữa Admin, Hệ thống** | | |
| **Authors** |  | | |
| **Activity Diagram** | | |  |
| **Sequence** | | |  |

**UC5 Kiểm soát truy cập**

|  |  |  |  |
| --- | --- | --- | --- |
| A diagram of a system  AI-generated content may be incorrect. | | | |
| **Use Case Number:** | **UC5** | | |
| **Use Case Name:** | Đăng nhập, Đăng xuất | | |
| **Actor (s):** | Buyer, Admin | | |
| **Maturity:** | Focused | | |
| **Summary:** | Buyer và admin thực hiện đăng nhập vào hệ thống, đăng xuất ra hệ thống | | |
| **Basic Course of Events:** | **Actor Action** | **System Response** | |
|  | 1. Đăng nhập |  | |
|  | 2. Hệ thống truy dữ liệu và loại tài khoản từ hệ thống rồi bắt đầu phiên  **E1** | |
|  | 3. Hệ thống hiển thị chức năng thực thi dựa theo loại tài khoản ( Buyer,Admin ) | |
|  | 4. Hệ thống dẫn người đùng đến trang chủ | |
| **Alternative Paths:** | Không | | |
| **Exception Paths:** | **E1.** Nếu tài khoản không đúng tài khoản hay mật khẩu, hệ thống sẽ báo lỗi sai thông tin đăng nhập, điều người dùng về lại phiên đăng nhập | | |
| **Extension Points:** | Không | | |
| **Triggers:** | Không | | |
| **Extension Points** | Không | | |
| **Triggers** | Người dùng truy cập vào trang nhưng chưa bắt đầu một phiên | | |
| **Assumptions** | Không | | |
| **Preconditions** | Không | | |
| **Post Conditions** | Không | | |
| **Authors** |  | | |
| **Activity Diagram** | | |
| **Sequence** | | |

## **1.6. User stories**

**US Quản lý sản phẩm**

Là một khách hàng, tôi muốn xem, tìm kiếm và lọc danh sách laptop theo tên, hãng, thành phần, giá sản phẩm để dễ dàng tìm thấy cuốn sách tôi cần.

Là một khách hàng, tôi muốn xem thông tin chi tiết của một dòng laptop để biết rõ nội dung, giá, hình ảnh và thông tin sản phẩm khi quyết định mua.

Là một quản trị viên hệ thống, tôi muốn thêm mới, chỉnh sửa hoặc xóa sản phẩm để đảm bảo danh mục sản phẩm luôn chính xác và được cập nhật.

Là một quản trị viên hệ thống, tôi muốn quản lý danh mục, thành phần để tổ chức laptop theo hệ thống rõ ràng và dễ tra cứu.

**US Quản lý giỏ hàng**

Là một khách hàng đã đăng nhập, tôi muốn thêm laptop vào giỏ hàng từ trang chi tiết sản phẩm để lưu lại những dòng laptop tôi muốn mua.

Là một khách hàng, tôi muốn xem lại toàn bộ các sản phẩm trong giỏ hàng trước khi thanh toán.

Là một khách hàng, tôi muốn thay đổi số lượng hoặc xóa sản phẩm trong giỏ hàng.

**US Quản lý thanh toán**

Là một khách hàng, tôi muốn lựa chọn phương thức thanh toán Momo/Ngân hàng hoặc COD để có thể thanh toán theo cách thuận tiện nhất cho mình.

Là một khách hàng, tôi muốn nhận được thông báo khi thanh toán thành công để biết rằng giao dịch của tôi đã được ghi nhận.

Là một quản trị viên hệ thống, tôi muốn xem và quản lý danh sách các giao dịch thành công hoặc đã hoàn tiền.

**US Kiểm soát truy cập**

Là một khách truy cập, tôi muốn đăng ký tài khoản mới để có thể đăng nhập và sử dụng các chức năng của hệ thống.

Là một quản trị viên hệ thống, tôi muốn thêm, chỉnh sửa hoặc xóa tài khoản người dùng.

Là một quản trị viên hệ thống, tôi muốn khóa hoặc mở khóa tài khoản người dùng để kiểm soát quyền truy cập của các tài khoản vi phạm hoặc không còn hoạt động.

## **1.7. Domain driven design**

**Mô tả miền nghiệp vụ**

Những lý do chính để chọn miền nghiệp vụ này:

1. Đã có hệ thống thực tế cho miền nghiệp vụ này, vì vậy mọi người có thể đối chiếu phần triển khai với một trang web đang hoạt động.
2. Miền nghiệp vụ cũng không quá đơn giản, vẫn có một số nghiệp vụ phức tạp và logic, không đơn thuần là các thao tác CRUD.
3. Miền nghiệp vụ không quá phức tạp nên dễ hiểu và không quá lớn và dễ triển khai.

Sản phẩm có các thực thể chính gồm Laptop, Thành phần, Danh mục. Khách hàng có thể xem, tìm kiếm và lọc danh sách Laptop. Quản trị viên hệ thống và nhân viên bán hàng có thể CRUD đối với Laptop, Danh mục, CPU, GPU, RAM, SSD.

Giỏ hàng chỉ khách hàng đã đăng nhập mới có thể thêm sản phẩm vào Giỏ hàng. Khách hàng có thể xem lại, thay đổi số lượng hoặc xóa sản phẩm khỏi Giỏ hàng và sau đó tiến hành đặt hàng và thanh toán.

Đơn hàng có thực thể chính là Đơn hàng và Chi tiết đơn hàng. Sau khi đã thêm sản phẩm vào giỏ hàng, khách hàng nhập thông tin giao hàng và phương thức thanh toán sẽ tạo đơn hàng. Quản trị viên hệ thống và nhân viên bán hàng có thể cập nhật trạng thái Đơn hàng.

Thanh toán có thực thể chính là Thanh toán, Phương thức thanh toán, Tình trạng thanh toán. Khách hàng có thể lựa chọn thanh toán bằng Momo/Ngân hàng. Quản trị viên hệ thống có thể xem danh sách các giao dịch Thanh toán.

Kiểm soát truy cập có thực thể chính là Người dùng và Chức vụ. Người dùng gồm Khách hàng, Quản trị viên hệ thống. Mỗi người dùng có thể đăng nhập vào hệ thống, đăng xuất khỏi hệ thống. Quản trị viên hệ thống có quyền quản lý các tài khoản người dùng.

**Mô hình khái niệm**

Sản phẩm: gồm Laptop, thành phần chính, Danh mục; khách hàng xem, tìm kiếm, lọc; quản trị viên hệ thống CRUD dữ liệu.

Giỏ hàng: khách hàng đăng nhập thêm, sửa, xóa sản phẩm và đặt hàng.

Đơn hàng: gồm Đơn hàng và Chi tiết đơn hàng; khách hàng tạo đơn khi đặt hàng, nhân viên bán hàng/quản trị viên hệ thống cập nhật trạng thái.

Thanh toán: gồm Thanh toán, Phương thức thanh toán và Tình trạng thanh toán; khách hàng có thể thanh toán bằng Momo/Ngân hàng, quản trị viên hệ thống xem giao dịch.

Kiểm soát truy cập: gồm Người dùng (khách hàng, quản trị viên hệ thống) và Chức vụ; quản trị viên hệ thống quản lý tài khoản, người dùng đăng nhập/đăng xuất.

A computer screen with white lines and dots

AI-generated content may be incorrect.

Hình 2.7 Sơ đồ mô hình khái niệm

## **1.8. Data model**

Mô hình thực thể kết hợp mức khái niệm mô tả các thực thể chính gồm Laptop, ImageLaptop, Brand, Cart, Category, Order, User, Payment và mối quan hệ giữa các thực thể gồm belong to và have.

Mô hình thực thể kết hợp mức logic được xây dựng từ mô hình khái niệm, trong đó mỗi thực thể được cụ thể hóa thành bảng dữ liệu với các khóa chính (PK), khóa ngoại (FK), thuộc tính và quan hệ giữa các bảng.

Mô hình thực thể kết hợp mức vật lý thể hiện cấu trúc lưu trữ dữ liệu thực tế trong cơ sở dữ liệu, được xây dựng từ mô hình logic. Mô hình bao gồm các bảng với các thuộc tính, kiểu dữ liệu và ràng buộc như khóa chính, khóa ngoại, được liên kết chặt chẽ để đảm bảo toàn vẹn dữ liệu và hỗ trợ truy vấn.

A computer screen shot of a computer screen

AI-generated content may be incorrect.

Hình 2.8 Mô hình thực thể kết hợp mức vật lý

## **1.9. Wireflow**

Khách hàng truy cập trang chính SkyWind để xem các danh mục và sản phẩm nổi bật, đồng thời có thể tìm kiếm và chọn dòng Laptop mong muốn. Khi bấm vào sản phẩm, hệ thống hiển thị thông tin chi tiết laptop, giá, mô tả, cấu hình.

Nếu muốn mua, khách hàng chọn “Thêm vào giỏ”. Nếu chưa đăng nhập, hệ thống sẽ yêu cầu đăng nhập trước khi tiếp tục. Sau khi đăng nhập, khách hàng có thể thêm sản phẩm vào giỏ và kiểm tra lại các mặt hàng đã chọn, thay đổi số lượng nếu cần.

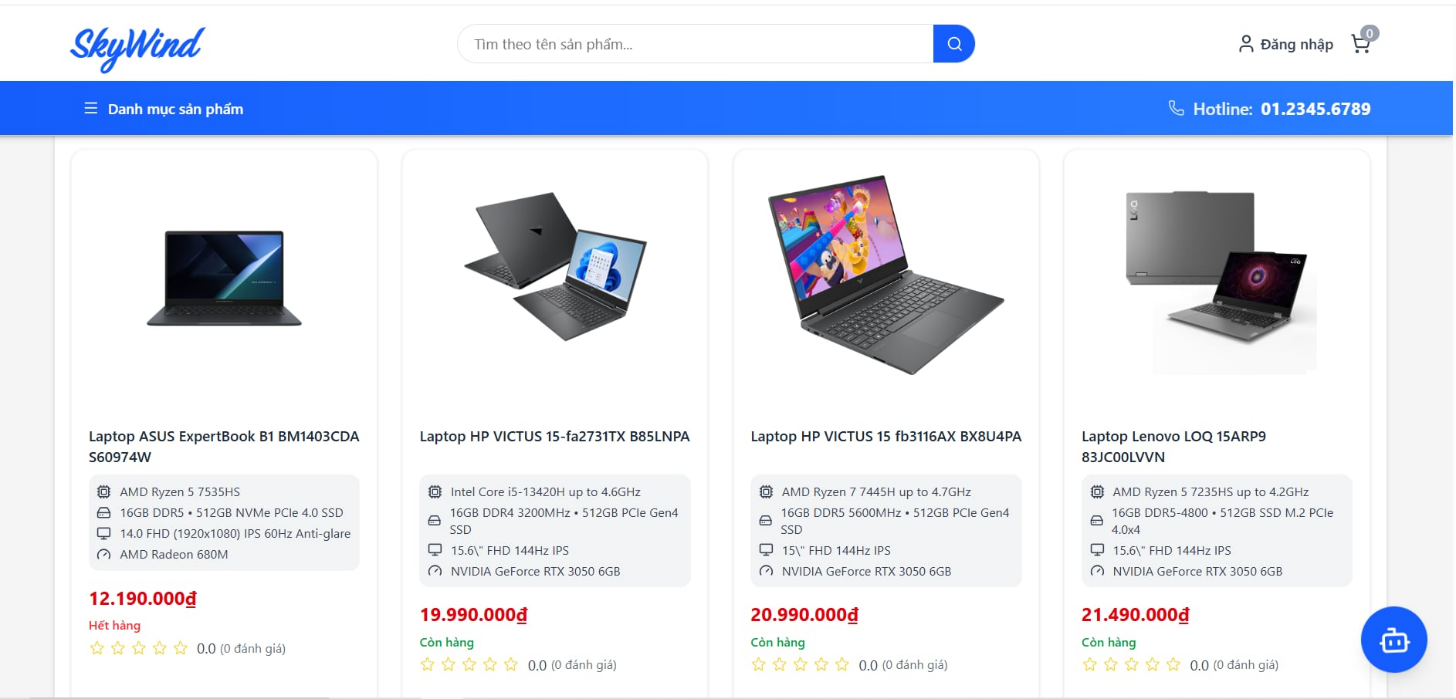
Khách hàng vào giỏ hàng để xem các sách đã thêm, điều chỉnh số lượng nếu cần, rồi bấm nút “Thanh toán” để chuyển sang bước tiếp theo. Ở trang thanh toán, khách hàng cần nhập thông tin giao hàng, chọn phương thức thanh toán (COD hoặc Momo), sau đó tiến hành đặt hàng và thanh toán.

Hệ thống sẽ hiển thị kết quả đặt hàng ở trang kết quả đặt hàng. Khách hàng có thể theo dõi đơn hàng bằng cách truy cập vào mục “Đơn hàng”. Khi bấm nút “Xem chi tiết”, khách hàng có thể xem toàn bộ thông tin chi tiết về đơn hàng.

## **1.10. Giao diện**

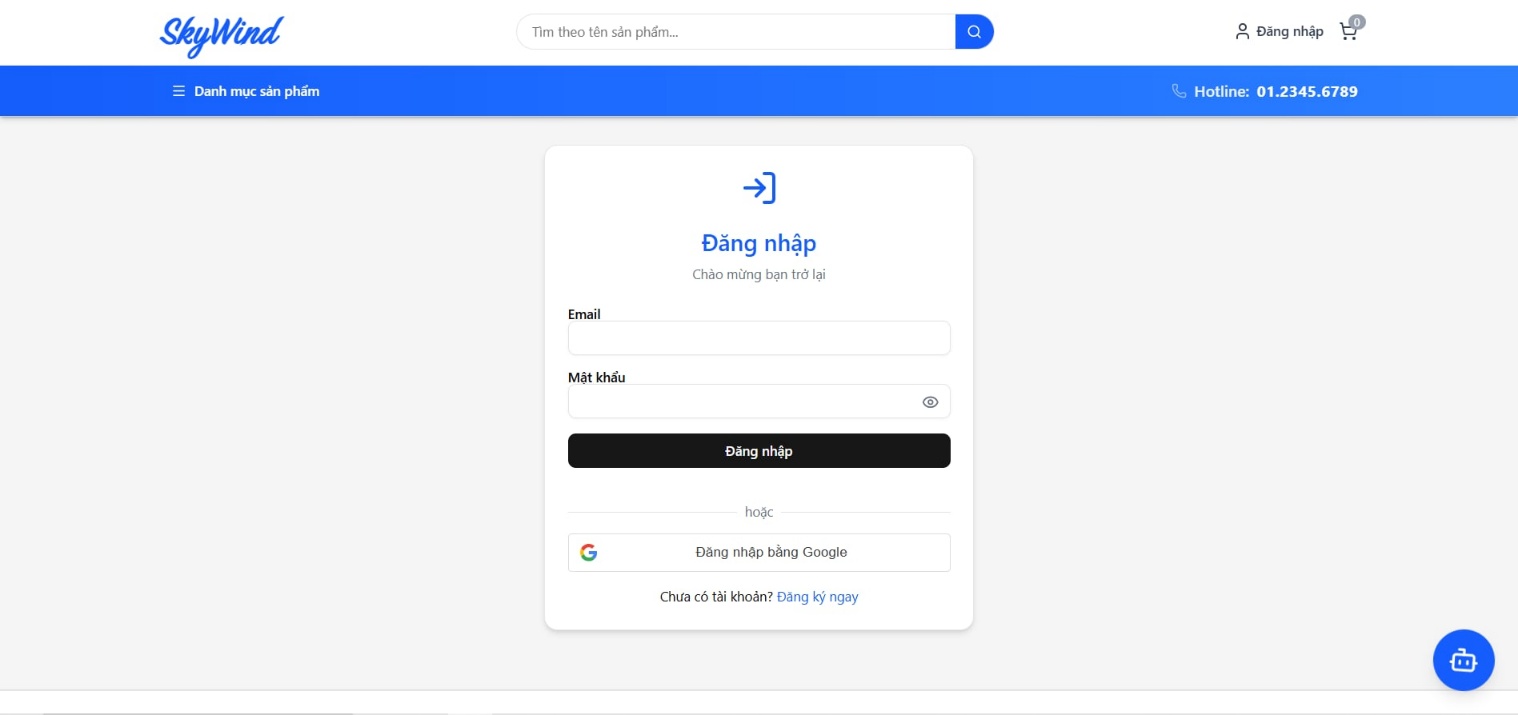
**Giao diện các trang khách hàng**

Trang chính là trang xuất hiện đầu tiên khi khách hàng truy cập vào website SkyWind.



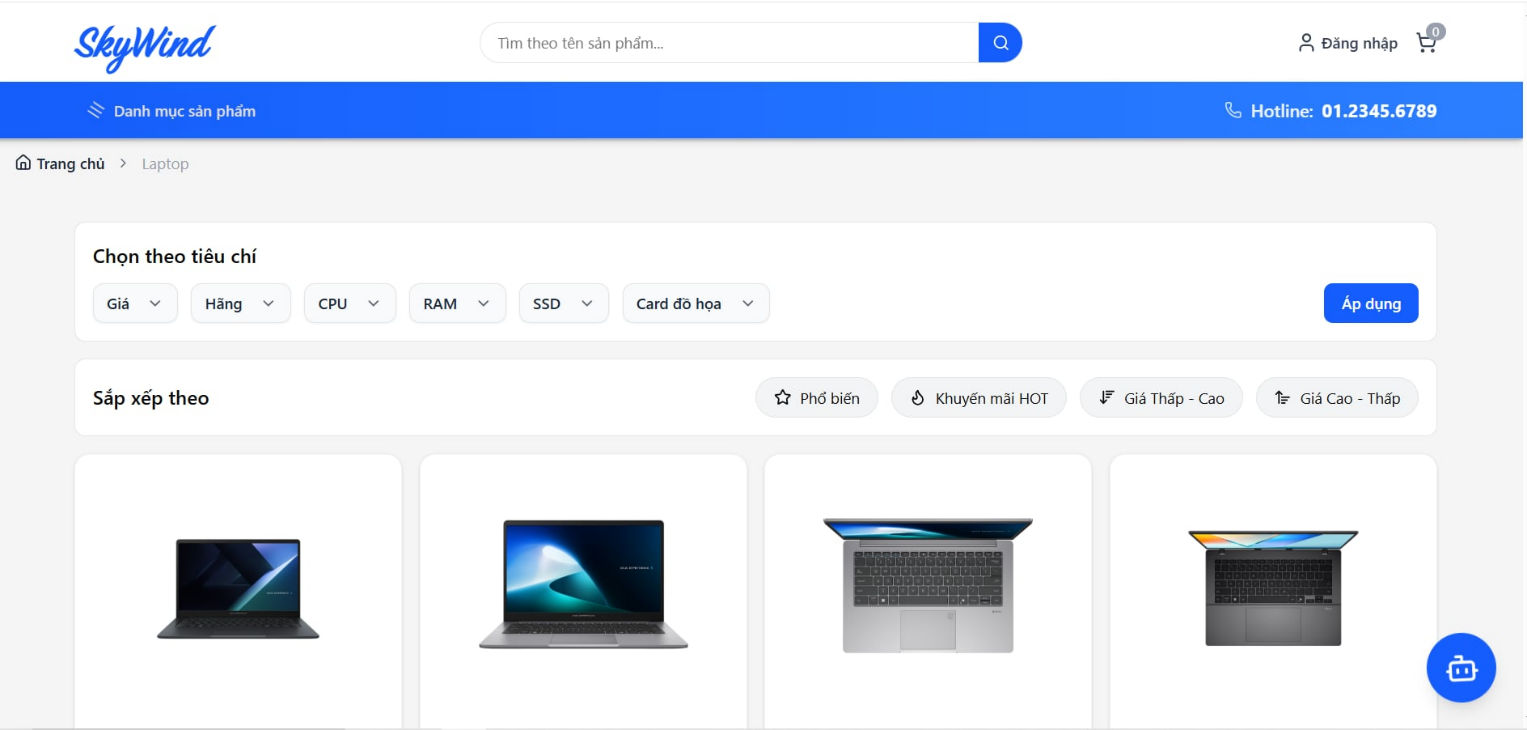
Hình 2.9 Giao diện trang chính

Trang đăng nhập khách hàng là trang yêu cầu khách hàng phải nhập email và mật khẩu đăng nhập thì mới được mua hàng trong website SkyWind.



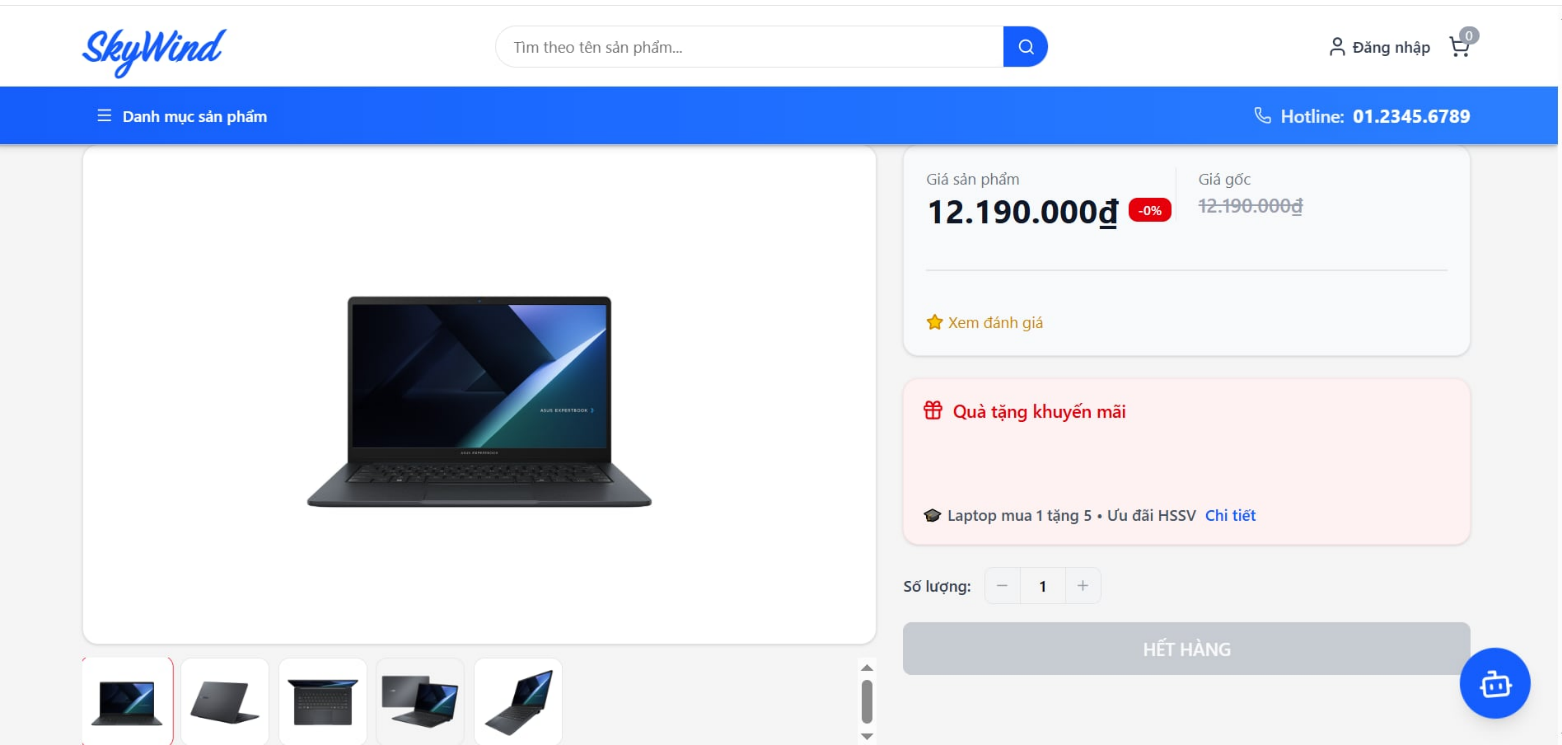
Hình 2.10 Giao diện trang đăng nhập

Trang danh sách laptop là trang khách hàng tìm kiếm sản phẩm bằng thanh tìm kiếm, chọn danh mục laptop hoặc bấm vào mục “Giảm giá” sẽ đến trang danh sách sách hiển thị các sách tương ứng.



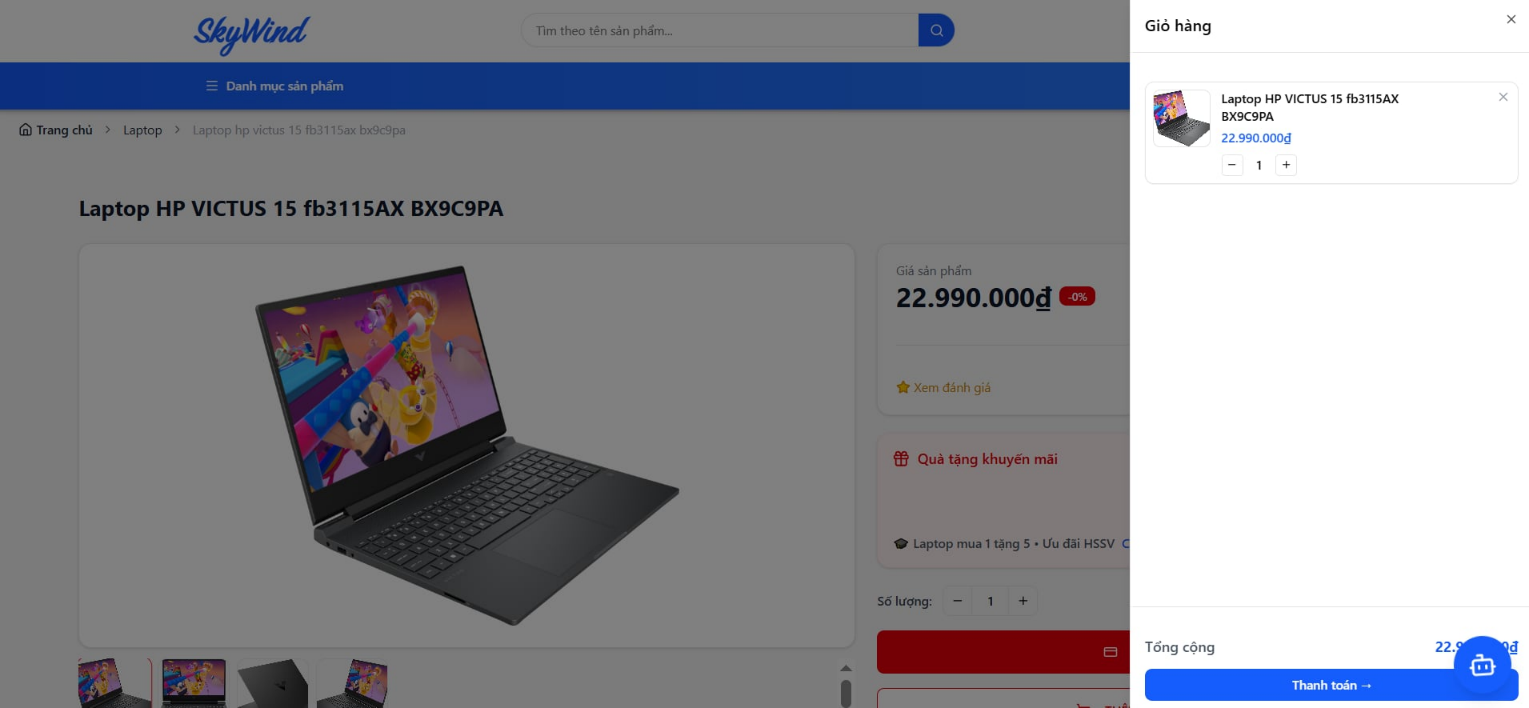
Hình 2.11 Giao diện danh sách sản phẩm

Trang chi tiết sản phẩm là trang khách hàng có thể xem các thông tin chi tiết của laptop và chọn số lượng mua để thêm sách vào giỏ hàng.



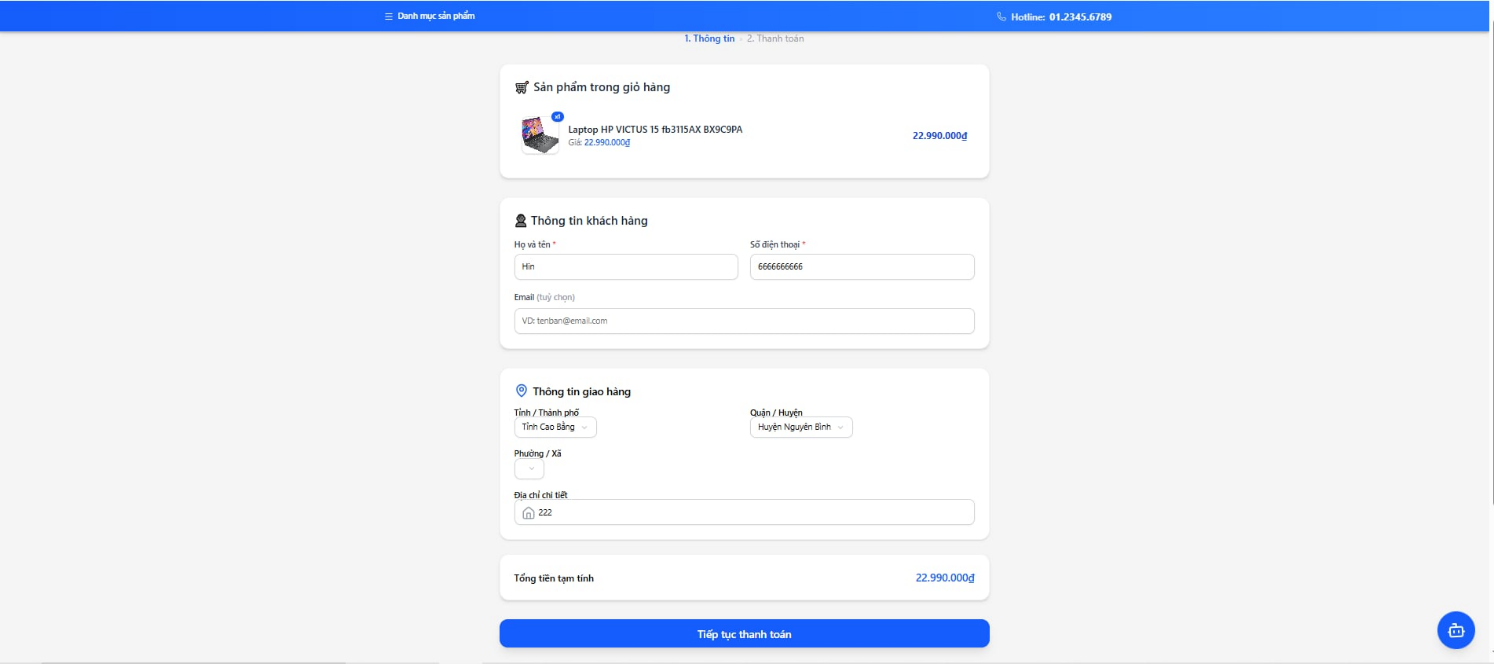
Hình 2.12 Giao diện chi tiết sản phẩm

Trang giỏ hàng là trang khách hàng sau khi đăng nhập có thể thêm laptop vào giỏ hàng, điều chỉnh số lượng mua, hoặc xóa laptop khỏi giỏ. Khi đã hoàn tất việc chọn sản phẩm, khách hàng nhấn vào nút “Thanh toán” để đặt hàng và thanh toán.

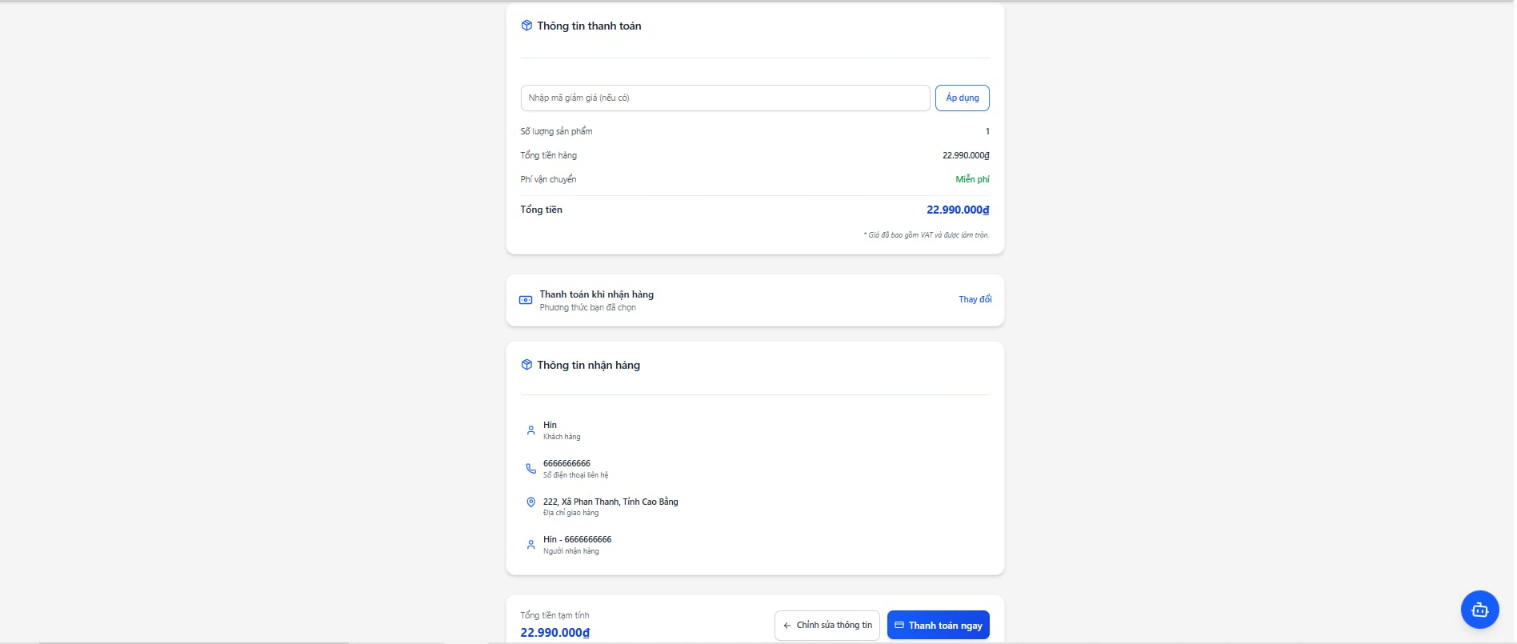


Hình 2.13 Giao diện giỏ hàng

Trang thanh toán là trang khách hàng đã thêm sản phẩm vào giỏ hàng và tiến thành đặt hàng và hoàn tất thanh toán.

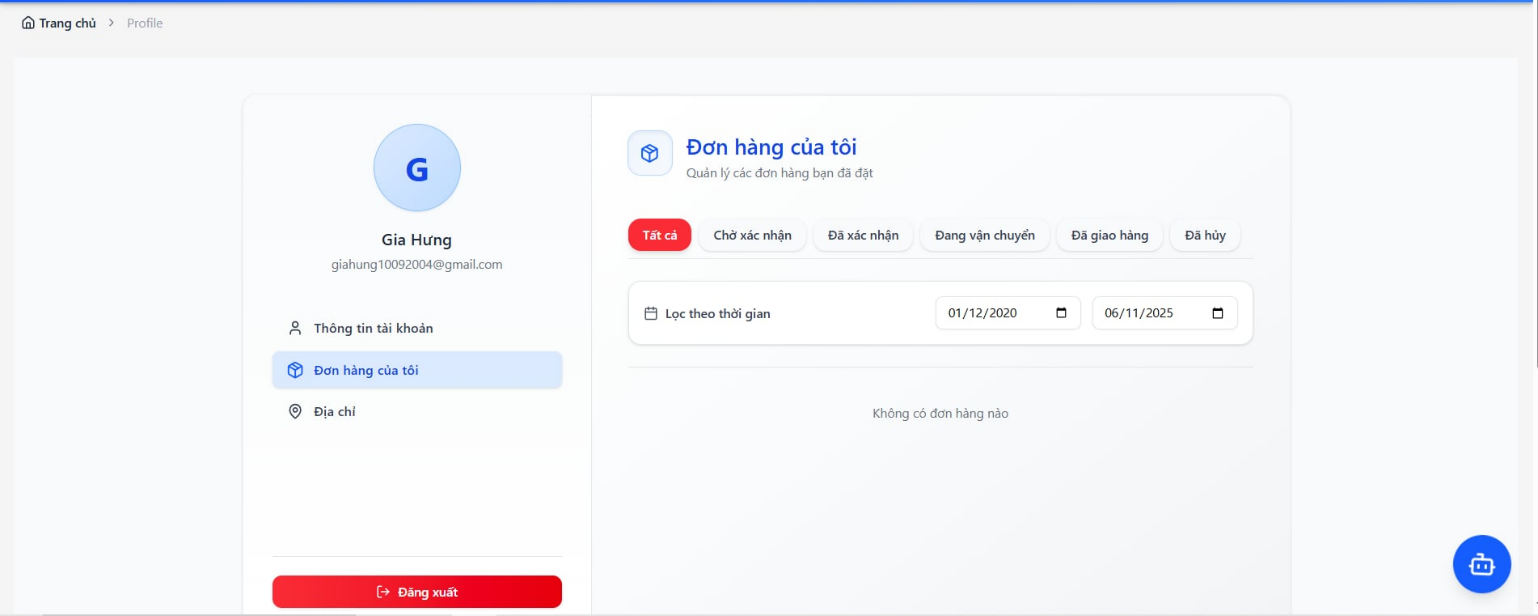


Hình 2.0.14 Giao diện trang kiểm tra thông tin thanh toán



Hình 2.15 Giao diện trang thanh toán

Trang danh sách đơn hàng là trang khách hàng đặt hàng thành công thì đơn hàng sẽ xuất hiện ở giao diện trang danh sách đơn hàng.



Hình 2.16 Giao diện trang đơn hàng

**Giao diện các trang quản trị viên**

Trang đăng nhập quản trị viên yêu cầu quản trị viên (quản trị viên hệ thống phải nhập email và mật khẩu để đăng nhập thì mới được vào quản trị hệ thống website SkyWind.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 2.17 Giao diện trang đăng nhập Admin

Trang danh sách sản phẩm là trang cho phép quản trị viên hệ thống hoặc nhân viên bán hàng xem danh sách các laptop, đồng thời cập nhật trạng thái hoặc xóa laptop.

Trang danh sách đơn hàng là trang quản trị viên hệ thống hoặc nhân viên bán hàng xem danh sách các đơn hàng, đồng thời cập nhật trạng thái đơn hàng.

Trang chi tiết đơn hàng là trang quản trị viên hệ thống hoặc nhân viên bán hàng xem đầy đủ chi tiết đơn hàng.

Trang danh sách thanh toán là trang quản trị viên hệ thống hoặc nhân viên bán hàng xem danh sách các giao dịch thanh toán.

Trang danh sách người dùng là trang quản trị viên hệ thống xem danh sách các tài khoản quản trị viên và khách hàng, đồng thời có thể khóa, mở khóa hoặc xóa tài khoản người dùng.

2. Thiết kế kiến trúc

## **2.1. Sơ đồ khối**

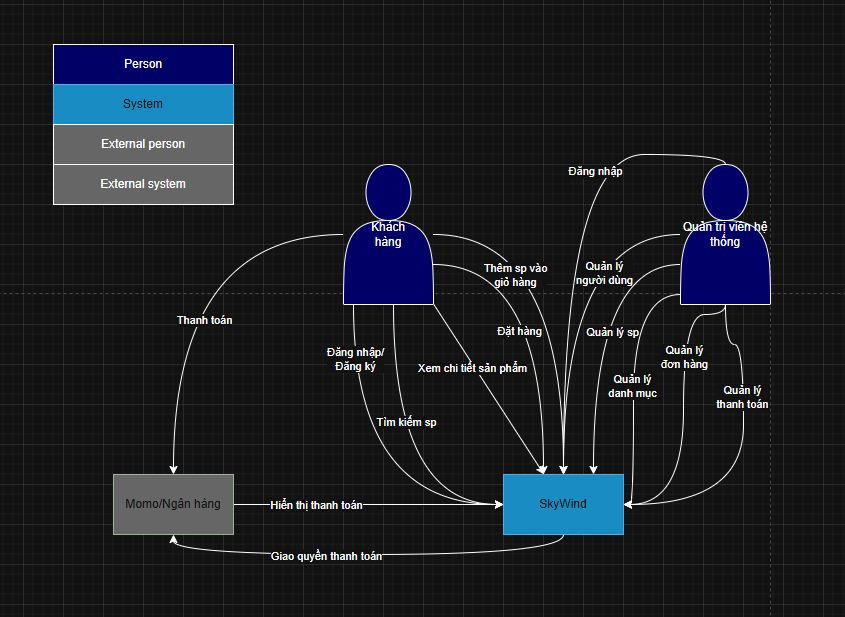
## **2.2. Kiến trúc C4**

**C1 – System context**

Khách hàng sử dụng hệ thống để đăng nhập/đăng ký, tìm kiếm, xem chi tiết và đặt hàng và thanh toán.

Quản trị viên hệ thống đăng nhập vào hệ thống để quản lý laptop, danh mục, cấu hình, thành phần, đơn hàng và thanh toán; quản trị viên có thể quản lý người dùng.

Hệ thống kết nối với cổng thanh toán Momo/Ngân hàng để xử lý giao dịch, nhận kết quả thanh toán và hiển thị giao diện thanh toán cho Khách hàng.



Hình 2.18. Sơ đồ system C1

**C2 – Container**

Container SPA ReactJS là giao diện người dùng, nơi khách hàng, quản trị viên và nhân viên bán hàng truy cập qua HTTP. Container này gửi yêu cầu đến SkyWind System API (Python) để xử lý nghiệp vụ.

Container Backend Python là nơi tương tác và chịu trách nhiệm truy vấn dữ liệu từ container PostgreSQL, nơi lưu trữ thông tin của bảng Lapptop, Order, Category...

Backend cũng giao tiếp hai chiều qua HTTP với Cổng thanh toán Momo/Ngân hàng để thực hiện thanh toán và nhận kết quả giao dịch.

A screenshot of a computer screen

AI-generated content may be incorrect.

Hình 2.18 Sơ đồ container C2

**C3 – Component (high-level)**

Giao diện người dùng (ReactJS) kết nối với tầng API do Python triển khai. Tầng API được chia thành các module riêng biệt theo nghiệp vụ: Sản phẩm, Giỏ hàng, Đơn hàng, Thanh toán và Kiểm soát truy cập.

Mỗi module sử dụng bảng dữ liệu riêng trong Database.

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.19 Sơ đồ component C3

**C4 – Code**

**Sơ đồ lớp**

Sơ đồ lớp biểu hiện các lớp Controller nhận yêu cầu từ người dùng, Service xử lý nghiệp vụ, Repository truy xuất dữ liệu, và Entity đại diện cho bảng trong cơ sở dữ liệu.

A black screen with white text

AI-generated content may be incorrect.

Hình 2.20 Sơ đồ lớp

## **2.3. Sơ đồ luồng giao tiếp**

Trình duyệt của người dùng giao tiếp với Frontend ReactJS qua HTTP (cổng 3000) bằng HTML/JSON và dùng JWT để xác thực.

Frontend (SPA ReactJS) sử dụng thư viện Axios và useSWR để gọi API qua HTTP (cổng 8080) đến Backend Python.

Backend là Python có các lớp Controller, Service, Repository, Entity tương tác với cơ sở dữ liệu MySQL qua giao thức TCP (cổng 3306), sử dụng MySQL driver/JPA + Hibernate. Backend còn kết nối với cổng thanh toán Momo qua HTTPS (cổng 443) để xử lý giao dịch.

## **2.4. Sơ đồ triển khai**

**A diagram of a server

AI-generated content may be incorrect.**

Hình 2.21. Sơ đồ triển khai

# Chương 3. Kế hoach kiểm thử

1. Giới thiệu

## **1.1. Mục đích**

Tài liệu kế hoạch kiểm thử cho dự án SkyWind được xây dựng nhằm xác định phương pháp, phạm vi, và nguồn lực kiểm thử cần thiết, đảm bảo phần mềm đáp ứng đầy đủ các yêu cầu nghiệp vụ và kỹ thuật. Đồng thời, tài liệu giúp phát hiện sớm các lỗi, nâng cao chất lượng sản phẩm, hướng dẫn triển khai kiểm thử một cách thống nhất và kiểm soát được toàn bộ quy trình kiểm thử.

## **1.2. Phạm vi**

Tài liệu kế hoạch kiểm thử này mô tả các hoạt động kiểm thử cho website SkyWind trong suốt vòng đời phát triển và triển khai. Phạm vi bao gồm kiểm thử chức năng, unit test, integration test, hệ thống và chấp nhận, dựa trên các yêu cầu trong tài liệu SRS, được thực hiện trên môi trường Development và Production theo kế hoạch.

2. Hạng mục được kiểm thử

## **2.1. Chức năng**

Đăng Nhập

* Xác minh rằng hệ thống cho phép đăng nhập cho chức vụ là khách hàng, quản trị viên.
* Xác minh rằng hệ thống cho phép đăng ký tài khoản cho khách hàng.
* Xác minh rằng hệ thống cho phép đăng xuất đối với tất cả người dùng đã đăng nhập.
* Xác minh rằng hệ thống thực thi phân quyền truy cập đúng, chỉ cho phép người dùng truy cập các chức năng phù hợp với vai trò của mình.

Quản Lý Sản Phẩm:

* Xác minh rằng hệ thống cung cấp các thao tác CRUD chức năng quản lý sản phẩm cho quản trị viên.
* Xác minh rằng hệ thống hỗ trợ quản lý liên quan đến sản phẩm, gồm có: thông tin sản phẩm, trạng thái sản phẩm , giá sản phầm
* Xác minh rằng hệ thống cho phép khách hàng truy cập và xem danh sách sản phẩm.
* Xác minh rằng hệ thống hỗ trợ các chức năng tìm kiếm, lọc và sắp xếp sản phẩm nhằm giúp khách hàng dễ dàng tìm kiếm sản phẩm mong muốn.
* Xác minh rằng hệ thống đảm bảo tính nhất quán và chính xác của dữ liệu sản phẩm giữa giao diện người dùng và hệ thống backend.

Quản Lý Giỏ Hàng:

* Xác minh rằng hệ thống tự động cập nhật và hiển thị chính xác thông tin giỏ hàng, bao gồm danh sách sản phẩm, số lượng và tổng tiền.
* Xác minh rằng hệ thống đảm bảo tính nhất quán dữ liệu giỏ hàng trong suốt quá trình người dùng thao tác và chuyển đổi giữa các trang.
* Xác minh rằng hệ thống có các ràng buộc nghiệp vụ đối với giỏ hàng giới hạn số lượng mua một sản phẩm theo số lượng hiện có.
* Xác minh rằng hệ thống cho phép khách hàng tiếp tục quy trình mua hàng từ giỏ hàng để chuyển sang bước đặt hàng.

Quản Lý Đơn Hàng:

* Xác minh rằng hệ thống cung cấp đầy đủ quản lý đơn hàng cho khách hàng, bao gồm đặt hàng, xem danh sách đơn hàng và xem chi tiết đơn hàng đã đặt.
* Xác minh rằng hệ thống cung cấp đầy đủ các chức năng quản lý đơn hàng cho quản trị viên, bao gồm xem danh sách, tìm kiếm, lọc và cập nhật trạng thái đơn hàng.
* Xác minh rằng hệ thống hỗ trợ quy trình xử lý đơn hàng theo đúng nghiệp vụ, từ lúc tạo đơn hàng đến khi hoàn tất
* Xác minh rằng hệ thống đảm bảo tính chính xác và nhất quán của dữ liệu đơn hàng, bao gồm trạng thái đơn hàng, thông tin sản phẩm và tổng tiền.

Quản Lý Thanh Toán:

* Xác minh rằng hệ thống cung cấp đầy đủ các phương thức thanh toán cho khách hàng, bao gồm thanh toán khi nhận hàng và thanh toán Banking và thanh toán MoMo
* Xác minh rằng hệ thống đảm bảo tính nhất quán và chính xác của dữ liệu thanh toán và đơn hàng.

## **2.2. Khả năng sử dụng**

Xác minh rằng hệ thống cung cấp thanh điều hướng rõ ràng, cho phép người dùng truy cập các trang chính như trang chủ, danh sách sản phẩm, giỏ hàng và đơn hàng.

Xác minh rằng người dùng có thể dễ dàng quay lại trang trước hoặc trang chủ trong quá trình sử dụng.

Xác minh rằng các nút chức năng chính được hiển thị rõ ràng, dễ nhận biết và dễ thao tác.

## **2.3. Tương thích**

Xác minh rằng giao diện website hiển thị đúng và đồng nhất trên các trình duyệt phổ biến như Chrome, Firefox và Safari.

Xác minh rằng hệ thống tự động điều chỉnh giao diện phù hợp với các kích thước màn hình khác nhau (Desktop, Tablet, Mobile).

Xác minh rằng các thành phần giao diện không bị vỡ hoặc lệch khi thay đổi kích thước màn hình.

Xác minh rằng người dùng có thể thao tác đầy đủ các chức năng trên các thiết bị khác nhau.

## **2.4. Giao diện**

Xác minh rằng các màu sắc, font chữ và kiểu hiển thị đúng theo thiết kế.

Xác minh rằng logo, hình ảnh, icon được hiển thị đúng vị trí và không bị biến dạng.

Xác minh rằng các thông báo lỗi, cảnh báo, hoặc thông tin trạng thái hiển thị đúng theo ngữ cảnh nghiệp vụ.

## **2.5. Bảo mật**

Xác minh rằng mật khẩu người dùng được mã hóa an toàn khi lưu trữ trong hệ thống.

Xác minh rằng hệ thống thực thi đúng cơ chế kiểm soát truy cập theo vai trò và không cho phép truy cập trái phép.

Xác minh rằng các API quan trọng được bảo vệ bằng cơ chế xác thực và phân quyền phù hợp.

## **2.6. Cơ sở dữ liệu**

Xác minh rằng các thao tác CRUD trên hệ thống được phản ánh chính xác trong cơ sở dữ liệu.

Xác minh rằng dữ liệu được lưu trữ đúng định dạng và kiểu dữ liệu theo thiết kế.

Xác minh rằng các mối quan hệ giữa các bảng dữ liệu (khóa chính, khóa ngoại và các ràng buộc) được duy trì chính xác.

Xác minh rằng các truy vấn tìm kiếm, lọc và báo cáo dữ liệu trả về kết quả đúng và trong thời gian chấp nhận được.

Xác minh rằng hệ thống có thể chịu được tải cực đại mà không bị crash (stress testing).

## **2.7. Hồi quy**

Xác minh rằng các chức năng hiện có của hệ thống vẫn hoạt động đúng sau khi có thay đổi hoặc nâng cấp.

Xác minh rằng các endpoint API, giao diện người dùng, cơ sở dữ liệu và các quy trình nghiệp vụ vẫn tương thích và chính xác.

Xác minh rằng các luồng nghiệp vụ chính của hệ thống vẫn hoạt động ổn định.

## **2.8. API**

Xác minh rằng các endpoint API hoạt động đúng chức năng theo thiết kế.

Xác minh rằng dữ liệu trả về từ API đầy đủ, chính xác và tuân theo định dạng quy định.

Xác minh rằng các API được bảo vệ đúng quyền truy cập và không cho phép người dùng truy cập trái phép.

Xác minh rằng các API tương tác đúng với cơ sở dữ liệu, giao diện người dùng và các endpoint liên quan khác.

Xác minh rằng các luồng nghiệp vụ chính thông qua API vẫn hoạt động ổn định sau khi có thay đổi hệ thống.

## **2.9. Kiểm thử xử lý lỗi và dữ liệu biên**

Xác minh rằng hệ thống xử lý đúng các trường hợp dữ liệu đầu vào không hợp lệ, rỗng hoặc vượt quá giới hạn cho phép.

Xác minh rằng hệ thống hiển thị thông báo lỗi rõ ràng và không gây gián đoạn hoặc dừng hoạt động hệ thống.

Xác minh rằng hệ thống ngăn chặn các thao tác nghiệp vụ không hợp lệ.

## **2.10. Kiểm thử hiệu năng (Performance Testing – mức cơ bản)**

Xác minh rằng thời gian phản hồi của các chức năng chính như xem sản phẩm, thêm giỏ hàng và đặt hàng nằm trong ngưỡng cho phép.

Xác minh rằng hệ thống vẫn hoạt động ổn định khi có nhiều yêu cầu truy cập đồng thời ở ức giả lập.

3. Hạng mục không được kiểm thử

## **3.1. Các chức năng thuộc hệ thống bên thứ ba**

Các chức năng xử lý nội bộ của cổng thanh toán Momo/Ngân hàng không được đưa vào phạm vi kiểm thử.

Nhóm chỉ kiểm thử các luồng tích hợp giữa hệ thống SkyWind và cổng thanh toán Momo/Ngân hàng thông qua API, bao gồm việc gửi yêu cầu thanh toán và nhận kết quả phản hồi.

Việc đảm bảo tính ổn định, hiệu năng và bảo mật nội bộ của hệ thống Momo không thuộc phạm vi kiểm thử của dự án.

## **3.2. Kiểm thử bảo mật nâng cao**

Kiểm thử xâm nhập (Penetration Testing) và các bài kiểm thử bảo mật chuyên sâu không được thực hiện.

Các lỗ hổng bảo mật ở mức hệ thống, hạ tầng và mạng không nằm trong phạm vi kiểm thử.

Nhóm chỉ thực hiện kiểm thử bảo mật ở mức cơ bản, tập trung vào kiểm soát truy cập và phân quyền người dùng.

## **3.3. Kiểm thử khả năng sử dụng chuyên sâu**

Kiểm thử khả năng sử dụng ở mức chuyên sâu (Usability Testing với người dùng thật, khảo sát UX) không được thực hiện.

Việc đánh giá cảm nhận, mức độ hài lòng hoặc hành vi người dùng không nằm trong phạm vi kiểm thử.

Nhóm chỉ thực hiện kiểm thử khả năng sử dụng ở mức cơ bản nhằm xác minh hệ thống có thể thao tác và sử dụng được.

4. Tiêu chí kiểm thử chấp nhận

## **4.1. Độ bao phủ kiểm thử**

Ít nhất 95% yêu cầu chức năng của hệ thống phải được thiết kế và thực thi test case.

Tất cả các use case quan trọng đều phải được kiểm thử.

## **4.2. Tỷ lệ trường hợp kiểm thử đạt**

100% test case mức Critical và Major phải đạt trạng thái Pass.

Tổng tỷ lệ test case đạt phải đạt ≥ 98%.

## **4.3. Số lượng trường hợp kiểm thử**

Toàn bộ test case ở các cấp độ sau phải được thực thi:

* Kiểm thử đơn vị (Unit Test).
* Kiểm thử tích hợp (Integration Test).
* Kiểm thử hệ thống (System Test).

## **4.4. Số lượng lỗi**

Không được tồn tại lỗi mức Critical tại thời điểm chấp nhận.

Số lỗi mức Major không vượt quá 2 lỗi và phải có giải pháp khắc phục được thống nhất.

Các lỗi mức Minor không ảnh hưởng đến chức năng chính của hệ thống.

## **4.5. Mật độ kiểm thử đơn vị**

Mật độ unit test phải đạt tối thiểu 80 Unit Test Case trên mỗi 1000 dòng mã nguồn (80 UTC/KLOC).

## **4.6. Độ bao phủ mã nguồn**

Các chỉ số bao phủ mã nguồn phải đạt:

* Statement Coverage ≥ 90%.
* Branch Coverage = 100%.
* Path Coverage = 100%.

## **4.7. Quy trình CI/CD**

Quy trình GitHub Actions CI/CD phải chạy hoàn tất với trạng thái thành công.

Không được tồn tại bất kỳ lỗi kiểm thử nào trong pipeline tự động.

5. Chiến lược kiểm thử

## **5.1. Phương pháp kiểm thử**

## **5.2. Loại kiểm thử**

|  |  |  |  |
| --- | --- | --- | --- |
| Loại kiểm thử | Mục đích | Kỹ thuật / Công cụ | Tiêu chí chấp nhận |
| Kiểm thử chức năng | Đảm bảo các chức năng kiểm soát truy cập, quản lý sản phẩm, giỏ hàng, đơn hàng, thanh toán và địa chỉ hoạt động đúng theo yêu cầu nghiệp vụ | Manual Testing; Automation Testing với pytest (Backend), Selenium (UI) | Các chức năng hoạt động đúng nghiệp vụ; không có lỗi nghiêm trọng ảnh hưởng đến quy trình mua hàng |
| Kiểm thử khả năng sử dụng | Đảm bảo người dùng có thể dễ dàng thao tác, điều hướng và sử dụng website | Kiểm thử thủ công trên giao diện ReactJS | Giao diện dễ sử dụng, luồng nghiệp vụ rõ ràng, các thao tác thuận tiện |
| Kiểm thử tương thích | Đảm bảo website hiển thị và hoạt động đúng trên nhiều trình duyệt và thiết bị khác nhau | Manual Testing trên Chrome, Firefox, Safari; Responsive Testing | Giao diện hiển thị đúng, không bị vỡ layout; người dùng thao tác đầy đủ chức năng |
| Kiểm thử giao diện | Đảm bảo giao diện hiển thị đúng thiết kế và thông tin được trình bày chính xác | Manual Testing | Màu sắc, font chữ, hình ảnh và thông báo hiển thị đúng theo thiết kế |
| Kiểm thử bảo mật | Đảm bảo an toàn thông tin người dùng và kiểm soát truy cập hệ thống | Kiểm thử xác thực và phân quyền API bằng pytest | Không cho phép truy cập trái phép; mật khẩu được mã hóa an toàn |
| Kiểm thử cơ sở dữ liệu | Đảm bảo dữ liệu được lưu trữ, cập nhật và liên kết chính xác trong hệ quản trị CSDL | Integration Testing với pytest và PostgreSQL | Dữ liệu chính xác, toàn vẹn, đúng ràng buộc khóa chính và khóa ngoại |
| Kiểm thử hồi quy | Đảm bảo các chức năng hiện có không bị ảnh hưởng sau khi hệ thống thay đổi hoặc nâng cấp | Automation Testing với pytest và Selenium | Các luồng nghiệp vụ chính vẫn hoạt động ổn định |
| Kiểm thử API | Đảm bảo các endpoint API hoạt động đúng chức năng, bảo mật và nhất quán dữ liệu | API Testing với pytest và thư viện requests | API trả về dữ liệu đúng, đúng định dạng và đúng quyền truy cập |
| Kiểm thử xử lý lỗi và dữ liệu biên | Đảm bảo hệ thống xử lý đúng các trường hợp dữ liệu không hợp lệ hoặc vượt giới hạn | Manual Testing; Automation Testing với pytest | Hệ thống hiển thị thông báo lỗi phù hợp; không xảy ra lỗi hệ thống |
| Kiểm thử hiệu năng (mức cơ bản) | Đảm bảo hệ thống phản hồi trong thời gian chấp nhận được đối với các chức năng chính | Kiểm thử thủ công, đo thời gian phản hồi | Thời gian phản hồi nằm trong ngưỡng cho phép; hệ thống hoạt động ổn định |

## **5.3. Cấp độ kiểm thử**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Loại** | **Cấp độ kiểm thử** | | | |
| **Kiểm thử đơn vị** | **Kiểm thử tích hợp** | **Kiểm thử hệ thống** | **Kiểm thử chấp nhận** |
| Kiểm thử chức năng | x | x | x | x |
| Kiểm giao diện |  |  | x | x |
| Kiểm thử cơ sở dữ liệu | x | x |  |  |
| Kiểm thử tương thích |  |  | x | x |
| Kiểm thử bảo mật | x | x | x |  |
| Kiểm thử hiệu năng |  |  | x |  |
| Kiểm thử khả năng sử dụng |  |  | x | x |
| Kiểm thử hồi quy | x | x | x |  |
| Kiểm thử API | x | x |  |  |

6. Tài nguyên

## **6.1. Nguồn nhân lực**

|  |  |  |
| --- | --- | --- |
| **Người thực hiện** | **Vai trò** | **Công việc cụ thể** |
| Nguyễn Mình Quân | Kiểm thử viên/Lập trình viên | Lập kế hoạch kiểm thử, thực hiện kiểm thử hệ thống |
| Đào Cư Đạt | Kiểm thử viên/Lập trình viên | Thiết kế Test Case, thực hiện kiểm thử đơn vị và kiểm thử tích hợp. |
| Ngô Gia Hưng | Kiểm thử viên/Lập trình viên | viết mã CI/CD trong GitHub Actions, ghi chép tài kiểm thử |
| Nguyễn Ngọc Sơn | Kiểm thử viên/Lập trình viên | Lập review checklist, ghi chép tài liệu kiểm thử |
| TS. Đỗ Như Tài | Product Owner | Thực hiện kiểm thử chấp nhận |

## **6.2. Lịch trình kiểm thử**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nhiệm vụ** | **Người phụ trách** | **Thời gian bắt đầu** | **Thời gian kết thúc** |
| Lập kế hoạch kiểm thử | Nguyễn Minh Quân | 01/10/2025 | 12/10/2025 |
| Thiết kế Test Case | Đào Cư Đạt | 13/10/2025 | 22/10/2025 |
| Lập review checklist | Nguyễn Ngọc Sơn | 23/10/2025 | 31/10/2025 |
| Thực hiện kiểm thử đơn vị và tích hợp | Đào Cư Đạt | 01/11/2025 | 15/11/2025 |
| Thực hiện kiểm thử hệ thống | Nguyễn Minh Quân | 16/11/2025 | 20/11/2025 |
| Viết mã CI/CD trong GitHub Actions | Ngô Gia Hưng | 21/11/2025 | 25/11/2025 |
| Ghi chép tài liệu kiểm thử | Ngô Gia Hưng, Nguyễn Ngọc Sơn | 26/11/2025 | 16/12/2025 |
| Thực hiện kiểm thử chấp nhận | TS. Đỗ Như Tài | 21/12/2025 | 26/12/2025 |

7. Môi trường

## **7.1. Phần cứng**

|  |  |  |
| --- | --- | --- |
| **Mục** | **Thành phần** | **Mục đích** |
| Development | Máy tính cá nhân | Lập trình, chạy kiểm thử đơn vị và kiểm thử tích hợp |
| Thiết bị | Desktop, Android, iOS | Kiểm thử giao diện và kiểm thử tương thích |

## **7.2. Phần mềm**

|  |  |  |
| --- | --- | --- |
| **Mục** | **Thành phần** | **Mục đích** |
| Development | VSCode, ReactJS, Python, Local PostgreSQL | Lập trình, chạy kiểm thử đơn vị và kiểm thử tích hợp |
| Trình duyệt | Chrome, Firefox, Safari | KIểm thử giao diện và kiểm thử tương thích |

## **7.3. Hạ tầng**

|  |  |  |
| --- | --- | --- |
| **Mục** | **Thành phần** | **Mục đích** |
| Production | Frontend và Backend deploy trên Render  PostgreSQL host | Triển khai phần mềm thực tế, thực hiện kiểm thử hệ thống và chấp nhận |
| CI/CD | GitHub Actions | Tự động hóa quy trình Build, Test và Deploy |

8. Tài liệu bàn giao

Lập tài liệu word bao gồm tài liệu Architecture Design, Database Design, Test Plan, Software Requirements Specification, Bug Report, và báo cáo tổng thể về phần mềm và kiểm thử.

Lập tài liệu excel bao gồm tài liệu tất cả giao diện, đặc tả các Use Case, Review Checklist, Test Report, Test Case, Test Design.

Mã nguồn của dự án được lưu trên GitHub bao gồm các file kiểm thử đơn vị, kiểm thử tích hợp và kiểm thử hệ thống và pipeline CI/CD trên GitHub Actions (.github/workflows/ci-cd.yml).

# Chương 4. Thiết kế kiểm thử

# **1. Giới thiệu**

Chương này trình bày quy trình và kết quả của việc thiết kế các trường hợp kiểm thử (Test Case) chi tiết dựa trên chiến lược V-Model dựa vào các tài liệu ở Chương 1, Chương 2, Chương 3.

# **2. Quy trình thiết kế kiểm thử theo V-model**

Quy trình thiết kế kiểm thử của nhóm được xây dựng hoàn toàn dựa trên mô hình V-Model – một mô hình nhấn mạnh tính đối xứng và tương đối hỗ giữa hoạt động phát triển và hoạt động kiểm thử.

Trong cấu trúc dạng chữ V, mỗi giai đoạn phân tích và thiết kế ở nhánh trái đều có một hoạt động kiểm thử tương ứng ở nhánh phải, nhằm đảm bảo rằng mọi yêu cầu đều có thể được xác minh thông qua một trường hợp kiểm thử cụ thể.

## **2.1. Phân tích yêu cầu – 1a**

Ở giai đoạn đầu tiên, nhóm tiến hành thu thập, phân tích và làm rõ các yêu cầu nghiệp vụ của hệ thống bán laptop trực tuyến. Mục tiêu của bước này là đảm bảo tất cả yêu cầu được mô tả đầy đủ, đúng phạm vi và có thể kiểm thử được. Nhóm xây dựng tài liệu đặc tả yêu cầu phần mềm (SRS) làm cơ sở cho toàn bộ hoạt động kiểm thử về sau.

Từ quá trình phân tích, nhóm xác định hệ thống có hai đối tượng sử dụng chính (Khách hàng và Quản trị viên) và phân rã thành 6 module chức năng: Quản lý sản phẩm, Giỏ hàng, Đơn hàng, Thanh toán, Kiểm soát truy cập

Kết quả phân tích yêu cầu là nền tảng trực tiếp cho việc xây dựng các trường hợp kiểm thử chấp nhận. Các yêu cầu được chuyển hóa thành tiêu chí nghiệm thu, đảm bảo rằng mọi chức năng người dùng mong đợi đều có thể được xác minh thông qua kiểm thử.

## **2.2. Thiết kế hệ thống – 2a**

Sau khi yêu cầu đã được xác định rõ, nhóm tiến hành thiết kế hệ thống ở mức tổng thể. Nội dung thiết kế bao gồm việc mô tả các chức năng chính, các luồng nghiệp vụ, các sơ đồ use case và mối quan hệ giữa các tác nhân tương tác với hệ thống.

Thông tin từ giai đoạn này được dùng để xây dựng kịch bản kiểm thử hệ thống. Các chức năng được mô tả trong sơ đồ use case sẽ trở thành đối tượng kiểm thử, bảo đảm rằng hệ thống vận hành đúng theo luồng nghiệp vụ tổng quan.

## **2.3. Thiết kế kiến trúc – 3a**

Trong giai đoạn thiết kế kiến trúc, nhóm áp dụng kiến trúc C4 Model (Context – Container – Component – Code) để phân rã hệ thống từ cấp độ ngữ cảnh đến cấp độ thành phần. Mục đích là xác định rõ giao tiếp giữa các tầng, sự phân tách trách nhiệm và luồng dữ liệu giữa các thành phần trong kiến trúc tổng thể.

Kết quả thiết kế kiến trúc là cơ sở để xây dựng kiểm thử tích hợp. Các điểm giao tiếp giữa các module – như giữa API và database, giữa backend và frontend – được liệt kê và phân tích để thiết kế các trường hợp kiểm thử nhằm đánh giá tính chính xác của việc trao đổi dữ liệu giữa các thành phần.

## **2.4. Thiết kế module – 4a**

Trong bước thiết kế module, nhóm thực hiện phân tích và thiết kế từng module của hệ thống. Các sơ đồ lớp (Class Diagram) được sử dụng để mô tả cấu trúc dữ liệu, thuộc tính, phương thức và mối quan hệ giữa các lớp.

Tài liệu từ bước này được sử dụng để xây dựng các trường hợp kiểm thử đơn vị. Các lớp và phương thức trọng yếu của từng module được xác định để viết kiểm thử tự động, giúp phát hiện sớm lỗi ở mức độ thấp nhất trước khi module được tích hợp vào hệ thống.

## **2.5. Kiểm thử đơn vị - 1b**

Dựa trên thiết kế module, nhóm tiến hành xây dựng và thực thi các trường hợp kiểm thử đơn vị để đánh giá tính đúng đắn của từng phương thức, hàm hoặc lớp độc lập.

Mục tiêu của kiểm thử đơn vị là đảm bảo từng thành phần nhỏ nhất hoạt động chính xác theo mô tả trong thiết kế, từ đó giảm thiểu rủi ro lỗi lan truyền sang các giai đoạn tích hợp và triển khai.

Các ca kiểm thử đơn vị được triển khai kiểm thử tự động bằng PyTest kết hợp với Mockito để mô phỏng (mock) các đối tượng phụ thuộc, giúp tách biệt logic kiểm thử khỏi các thành phần bên ngoài như database hay service khác. Việc sử dụng Mockito cho phép nhóm cô lập và kiểm tra từng hàm một cách chính xác, đặc biệt trong các tình huống cần mô phỏng dữ liệu hoặc hành vi phức tạp.

Toàn bộ kiểm thử đơn vị được tích hợp vào môi trường CI/CD thông qua GitHub Actions, cho phép các bài test chạy tự động mỗi khi có cập nhật mã nguồn. Điều này giúp nhóm phát hiện lỗi sớm, ổn định chất lượng phần mềm trong suốt quá trình phát triển.

Ví dụ kiểm thử đơn vị của dự án:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Unit under test** | **Test description** | **Test data** | **Result** |
| UT-AUTH-003 | registerUser | Đăng ký người dùng mới Email chưa tồn tại | Email chưa tồn tại | Status 201 |
| UT-AUTH-004 | requireAdmin | Kiểm tra quyền admin hợp lệ Role = admin | Role = admin | Cho phép truy cập |

## **2.6. Kiểm thử tích hợp – 2b**

Sau khi từng module đã ổn định thông qua kiểm thử đơn vị, nhóm tiến hành kiểm thử tích hợp để đánh giá sự tương tác giữa các thành phần trong hệ thống. Kiểm thử tích hợp tập trung vào việc xác minh dữ liệu truyền tải qua các API, các tầng giao tiếp giữa Controller – Service – Repository, cũng như sự tương thích giữa backend và database.

Nhóm sử dụng PyTest kết hợp MockMvc để kiểm thử tích hợp API trong môi trường mô phỏng gần giống thực tế. MockMvc cho phép gửi request, nhận response và kiểm tra toàn bộ luồng xử lý mà không cần chạy ứng dụng thực tế trên server. Ngoài ra, cơ sở dữ liệu trong quá trình kiểm thử được thay thế bằng H2 Database (in-memory) nhằm đảm bảo kiểm thử có tính cô lập, lặp lại và không gây ảnh hưởng đến dữ liệu thật.

Các ca kiểm thử tích hợp được thiết kế dựa trên những điểm giao tiếp đã xác định trong giai đoạn Thiết kế kiến trúc, đảm bảo quá trình kiểm thử tuân thủ đúng cấu trúc hệ thống và bao phủ toàn bộ các luồng tương tác quan trọng.

Ví dụ kiểm thử tích hợp của dự án:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Unit under test** | **Test description** | **Test data** | **Result** |
| IT-PAY-001 | Payment API | Tạo payment | POST /api/payment | 201 Created |
| IT-PROD-001 | Product API | Lấy danh sách sản phẩm | GET /api/products | 200 OK |

## **2.7. Kiểm thử hệ thống – 3b**

Ở cấp độ cao hơn, nhóm tiến hành kiểm thử hệ thống nhằm đánh giá toàn bộ hoạt động của phần mềm như một thể thống nhất. Kiểm thử hệ thống được thực hiện theo cách tiếp cận thủ công dựa trên tập Test Case đã được xây dựng ở giai đoạn thiết kế kiểm thử, bao gồm đầy đủ các tình huống theo use case và luồng nghiệp vụ chính của người dùng.

Các hoạt động kiểm thử bao gồm:

* Kiểm thử chức năng: kiểm tra các luồng nghiệp vụ từ đầu đến cuối (end-to-end) như đăng nhập, mua hàng, thanh toán…
* Kiểm thử giao diện: đảm bảo giao diện trực quan, nhất quán và đáp ứng tiêu chuẩn hiển thị.
* Kiểm thử bảo mật cơ bản: xác minh luồng đăng nhập, phân quyền và kiểm soát truy cập.

Kết quả kiểm thử hệ thống phản ánh chất lượng tổng thể của phần mềm trước khi chuyển sang giai đoạn nghiệm thu.

## **2.8. Kiểm thử chấp nhận – 4b**

Kiểm thử chấp nhận là giai đoạn cuối cùng của quy trình kiểm thử, nhằm đánh giá hệ thống dưới góc nhìn người dùng thực tế. Ở bước này, nhóm đối chiếu các tiêu chí nghiệm thu với tài liệu SRS để đảm bảo hệ thống đáp ứng đầy đủ yêu cầu của khách hàng.

Sau khi hoàn thành kiểm thử hệ thống nội bộ, phần mềm được bàn giao cho Product Owner tiến hành kiểm thử chấp nhận. Product Owner sẽ đóng vai trò người dùng cuối, mô phỏng các tình huống thực tế và đưa ra đánh giá về mức độ đáp ứng nhu cầu, tính tiện dụng, độ ổn định và mức độ phù hợp với nghiệp vụ.

Kết quả kiểm thử chấp nhận là căn cứ quan trọng để đưa ra quyết định cuối cùng về việc triển khai hệ thống lên môi trường vận hành.

# **3. Kỹ thuật thiết kế kiểm thử**

## **3.1. Kiểm thử hộp đen**

Kiểm thử hộp đen là kỹ thuật kiểm thử chính được áp dụng xuyên suốt trong đồ án. Với kỹ thuật này, hệ thống được kiểm tra dựa trên hành vi bên ngoài thông qua dữ liệu đầu vào và kết quả đầu ra, không xét đến cấu trúc hay logic cài đặt bên trong mã nguồn. Kỹ thuật kiểm thử hộp đen được sử dụng trong toàn bộ các test case chức năng, bao gồm kiểm thử đăng nhập, đăng ký, phân quyền, quản lý sản phẩm, giỏ hàng, đơn hàng và thanh toán. Đối với API Testing, các endpoint được kiểm tra thông qua các request hợp lệ và không hợp lệ để xác minh mã trạng thái HTTP, dữ liệu phản hồi và thông báo lỗi. Việc áp dụng kiểm thử hộp đen giúp đảm bảo hệ thống đáp ứng đúng yêu cầu nghiệp vụ và phù hợp với góc nhìn của người dùng cuối.

## **3.2. Phân lớp tương đương (Equivalence Partitioning)**

Kỹ thuật phân lớp tương đương được áp dụng nhằm giảm số lượng test case nhưng vẫn đảm bảo độ bao phủ kiểm thử. Các dữ liệu đầu vào được chia thành các lớp hợp lệ và không hợp lệ, sau đó lựa chọn các giá trị đại diện để thực hiện kiểm thử. Kỹ thuật này được sử dụng rõ rệt trong các chức năng xác thực và nhập liệu, bao gồm đăng nhập, đăng ký tài khoản, cập nhật thông tin người dùng, thêm hoặc chỉnh sửa sản phẩm và nhập thông tin thanh toán. Ví dụ, trong chức năng đăng nhập, dữ liệu được chia thành các lớp như tài khoản hợp lệ, tài khoản không tồn tại, mật khẩu sai và dữ liệu trống. Việc áp dụng phân lớp tương đương giúp phát hiện lỗi hiệu quả mà không cần kiểm thử toàn bộ các trường hợp dữ liệu có thể xảy ra.

## **3.3. Kiểm thử giá trị biên (Boundary Value Analysis)**

Kiểm thử giá trị biên được áp dụng cho các chức năng có ràng buộc về giới hạn dữ liệu nhằm đảm bảo hệ thống xử lý chính xác các trường hợp tại ngưỡng. Kỹ thuật này được sử dụng chủ yếu trong các chức năng liên quan đến số lượng sản phẩm trong giỏ hàng, tồn kho sản phẩm, số lượng đặt mua và các trường bắt buộc trong biểu mẫu. Các trường hợp như số lượng bằng 0, bằng tồn kho, vượt quá tồn kho hoặc để trống trường dữ liệu được kiểm thử nhằm phát hiện các lỗi tiềm ẩn tại ranh giới điều kiện nghiệp vụ. Việc áp dụng kiểm thử giá trị biên giúp hạn chế các lỗi nghiêm trọng có thể ảnh hưởng trực tiếp đến dữ liệu và quy trình mua hàng.

## **3.4. Kiểm thử dựa trên luồng nghiệp vụ (Workflow-based Testing)**

Kiểm thử dựa trên luồng nghiệp vụ được áp dụng thông qua việc xây dựng TestDesign\_Workflow từ các sơ đồ activity của hệ thống. Mỗi workflow phản ánh đầy đủ các bước xử lý nghiệp vụ từ khi người dùng bắt đầu thao tác cho đến khi kết thúc chức năng. Kỹ thuật này được sử dụng cho các chức năng chính như đăng nhập, duyệt và tìm kiếm sản phẩm, thêm vào giỏ hàng, đặt hàng, thanh toán và quản lý đơn hàng của quản trị viên. Các test case được liên kết trực tiếp với từng bước trong workflow nhằm đảm bảo mọi nhánh xử lý, bao gồm cả luồng thành công và luồng lỗi, đều được kiểm thử. Kiểm thử theo luồng nghiệp vụ giúp đánh giá tính toàn vẹn của quy trình và khả năng phối hợp giữa frontend, backend và cơ sở dữ liệu.

## **3.5. Kiểm thử dựa trên vai trò và phân quyền (Role-based Testing)**

Kỹ thuật kiểm thử dựa trên vai trò được áp dụng để đảm bảo hệ thống thực thi đúng cơ chế phân quyền truy cập. Các test case được thiết kế riêng cho từng vai trò như Buyer và Admin nhằm kiểm tra khả năng truy cập chức năng, giới hạn quyền và xử lý truy cập trái phép. Kỹ thuật này được sử dụng trong các chức năng truy cập trang quản trị, quản lý sản phẩm, quản lý đơn hàng và các API dành riêng cho quản trị viên. Việc kiểm thử theo vai trò giúp đảm bảo tính bảo mật, ngăn chặn truy cập trái phép và tuân thủ đúng yêu cầu nghiệp vụ đã đặt ra.

## **3.6. Kiểm thử API (API Testing)**

Kiểm thử API được áp dụng để kiểm tra các dịch vụ backend độc lập với giao diện người dùng. Các API liên quan đến xác thực, sản phẩm, giỏ hàng, đơn hàng và thanh toán được kiểm thử bằng công cụ Postman và MockMvc. Các test case tập trung vào việc kiểm tra phương thức HTTP, dữ liệu request, dữ liệu response, mã trạng thái và xử lý lỗi. Đối với các API tích hợp bên thứ ba như thanh toán MoMo, các trường hợp redirect, thành công và thất bại đều được kiểm thử nhằm đảm bảo hệ thống xử lý đúng các tình huống phát sinh. Kiểm thử API giúp phát hiện lỗi sớm ở tầng backend và hỗ trợ hiệu quả cho kiểm thử tích hợp.

## **3.7. Kiểm thử khả năng sử dụng (Usability Testing)**

Kiểm thử khả năng sử dụng được thực hiện dưới hình thức kiểm thử thủ công nhằm đánh giá trải nghiệm người dùng. Các test case usability được thiết kế cho các chức năng chính như đăng nhập, tìm kiếm sản phẩm, thao tác giỏ hàng, nhập địa chỉ và thanh toán. Nội dung kiểm thử tập trung vào bố cục giao diện, khả năng nhận biết nút chức năng, thông báo phản hồi và mức độ dễ hiểu của các biểu mẫu. Việc áp dụng kiểm thử usability giúp đảm bảo hệ thống thân thiện với người dùng và đáp ứng tốt nhu cầu sử dụng thực tế.

# **4. Phương pháp thiết kế kiểm thử**

## **4.1. Kiểm thử thủ công**

Dự án áp dụng kiểm thử thủ công cho giai đoạn kiểm thử chấp nhận. Phương pháp này được sử dụng nhằm đánh giá hệ thống dưới góc nhìn người dùng cuối, kiểm tra các luồng nghiệp vụ chính và xác nhận mức độ đáp ứng của hệ thống so với các yêu cầu đã đặc tả.

Các ca kiểm thử thủ công được thực hiện dựa trên Test Case và Review Checklist đã thiết kế, kết quả kiểm thử được ghi nhận và đối chiếu với kết quả mong đợi trước khi tiến hành nghiệm thu hệ thống.

## **4.2. Kiển thử tự động**

Dự án áp dụng kiểm thử tự động thông qua việc tích hợp quy trình CI/CD bằng GitHub Actions nhằm tự động hóa quá trình build, kiểm thử và triển khai hệ thống. Kiểm thử tự động được sử dụng cho các kiểm thử hộp trắng bao gồm kiểm thử đơn vị và kiểm thử tích hợp và kiểm thử hệ thống.

Mỗi khi có thay đổi mã nguồn được đẩy lên kho lưu trữ GitHub, pipeline CI/CD sẽ tự động kích hoạt job test Job này thực hiện các bước build dự án và chạy toàn bộ các bộ kiểm thử tự động đã được xây dựng.

* Nếu job test thất bại, pipeline sẽ tự động tạo Issue trên GitHub để ghi nhận lỗi phát sinh, hỗ trợ nhóm phát triển theo dõi và xử lý kịp thời.
* Nếu job test thành công, pipeline sẽ tự động đóng các Issue liên quan đến lỗi kiểm thử trước đó, đảm bảo trạng thái hệ thống luôn được cập nhật chính xác.

Khi mã nguồn được merge vào nhánh chính (Main), pipeline CI/CD sẽ kích hoạt job deploy, thực hiện gọi đến nền tảng Render để tiến hành triển khai lại hệ thống.

# **5. Áp dụng GenAI trong việc sinh Test Case**

Trong đồ án này, Generative AI (GenAI) được sử dụng như một công cụ hỗ trợ trong giai đoạn thiết kế test case. GenAI không thay thế vai trò của tester mà đóng vai trò phân tích yêu cầu và gợi ý các kịch bản kiểm thử ban đầu, từ đó giúp nâng cao hiệu quả và độ bao phủ kiểm thử.

## **5.1. Đầu vào GenAI**

Đầu vào của GenAI bao gồm các tài liệu như: Yêu cầu hệ thống, SRS và User Story. Các tài liệu này cung cấp thông tin về chức năng, nghiệp vụ và các ràng buộc của hệ thống.

## **5.2. Giai đoạn xử lý của GenAI**

Trong giai đoạn GenAI Processing, GenAI thực hiện các bước sau:

* Phân tích ngữ nghĩa yêu cầu nhằm hiểu mục tiêu chức năng và nghiệp vụ của hệ thống.
* Sinh test scenario dựa trên các luồng chính và luồng phụ của chức năng.
* Sinh test case thô, bao gồm các trường hợp kiểm thử cơ bản, trường hợp biên và trường hợp lỗi.

## **5.3. Hoàn thiện test case**

Sau khi được chỉnh sửa và bổ sung, các test case đạt yêu cầu sẽ trở thành test case chính thức và được sử dụng trong quá trình thực thi kiểm thử (thủ công hoặc tự động).

# Chương 5. Báo cáo kiểm thử

# **1. Tổng quan về quá trình kiểm thử**

Sau khi hoàn tất giai đoạn thiết kế kiểm thử ở Chương 4, nhóm đã tiến hành thực hiện kiểm thử cho hệ thống SkyWind theo đúng kế hoạch đã đề ra. Quá trình kiểm thử được triển khai trong 6 tuần bao gồm các giai đoạn từ kiểm thử đơn vị, kiểm thử tích hợp, kiểm thử hệ thống và kiểm thử chấp nhận.

Nhóm tập trung vào kiểm thử toàn bộ các chức năng nghiệp vụ chính của hệ thống .Các Test Case được thực hiện một cách cẩn thận, chi tiết và được ghi lại đầy đủ kết quả trong các tài liệu bàn giao.

# **2. Báo cáo trường hợp kiểm thử**

## **2.1. Giới thiệu**

Các trường hợp kiểm thử này được phân loại thành hai nhóm Positive (kiểm tra hành vi đúng với mong đợi của hệ thống) và Negative (kiểm tra khả năng xử lý lỗi, ngoại lệ và điều kiện không hợp lệ).

Bộ kiểm thử được xây dựng cho 5 module chức năng chính của hệ thống gồm kiểm soát truy cập, sản phẩm, giỏ hàng, đơn hàng, thanh toán và sổ địa chỉ, với tổng cộng 78 trường hợp kiểm thử.

## **2.2. Phạm vi bao phủ**

Nhóm đã xây dựng 78 Test Case bao phủ 100% các chức năng chính của hệ thống. Phạm vi bao phủ của các phi chức năng là 0%.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Module** | **Mã** | **Số lượng** | **Mô tả** | **Kết quả** |
| Authentification | TC-AUTH-SYS | 11 | Kiểm tra các quy trình đăng nhập, đăng ký  Kiểm tra các thao tác thêm, sửa, xóa người dùng | 11 Pass |
| Product Management | TC-PROD-SYS | 12 | Kiểm tra khả năng hiển thị, tìm kiếm, lọc sản phẩm cho khách hàng  Kiểm tra các thao tác thêm, sửa, xóa cho quản trị viên | 11 Pass  1 Fail |
| Cart | TC-CART-SYS | 8 | Kiểm tra các thao tác thêm, sửa, xóa và kiểm tra số lượng hiện có khi thêm sản phẩm vào giỏ hàng | 8 Pass |
| Order | TC-ORDER-SYS | 5 | Kiểm tra quản lý và cập nhật trạng thái đơn hàng của quản trị viên | 5 Pass |
| Payment | TC-ORDER-SYS | 8 | Kiểm tra các phương thức thanh toán (COD và Momo) và xử lý các kịch bản liên quan đến kết quả giao dịch | 8 Pass |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Module** | **Mã** | **Số lượng** | **Mô tả** | **Kết quả** |
| Authentification | UT-AUTH | 5 | Unit test | 5 Pass |
| Product Management | UT-PROD | 7 | Unit test | 7 Pass |
| Product Management | IT\_PROD | 4 | Intergration test | 4 Pass |
| Order | UT-ORDER | 6 | Unit test | 6 Pass |
| Payment | UT-PAY | 4 | Unit test | 4 Pass |
| Payment | IT-PAY | 1 | Intergration test | 1 Pass |
| User | IT-User | 2 | Intergration test | 2 Fail |
| Selenium | SEL | 2 | Selenium test | 2 Pass |
| Performance | UT-PERM | 3 | Performance test | 3 Pass |

## **2.3. Kết quả thực thi**

Biểu đồ tròn hiện thị tỷ lệ phần trăm thực thi của tất cả các Test Case.

Tổng cộng 78 Test Case trong đó có 75 thành công và 3 thất bại.

* 97.2% các Test Case thành công (Passed).
* 2.8% các Test Case thất bại (Failed).
* 0% các Test Case bị chặn (Blocked).
* 0% các Test Case chưa chạy (Not run).

Hình 5.1. Biểu đồ tròn tỷ lệ phần trăm thực thi Test Case.

# **3. Báo cáo lỗi**

## **3.1. Phân loại lỗi theo mức độ nghiêm trọng**

Trong quá trình kiểm thử Test Case, nhóm đã phát hiện và ghi nhận 3 lỗi. Các lỗi này được phân loại theo mức độ nghiêm trọng như sau:

Severity 2 – High: có 1 lỗi đây là lỗi liên quan đến hệ thống. Lỗi khá nghiêm trọng là quản trị viên không xóa được sản phẩm.

Severity 4 – Low: có 1 lỗi đây là lỗi nhỏ, khi kiểm thử intergration test thì không tạo được user mới nhưng trên trang sản phẩm của người dùng vẫn thêm được user mới.

Severity 4 – Low: có 1 lỗi đây là lỗi nhỏ, khi kiểm thử intergration test thì không lấy được IDuser mới nhưng trên trang sản phẩm của người dùng vẫn thêm được user mới.

## **3.2. Thống kê lỗi**

Biểu đồ cột hiện thị số lượng lỗi của Test Case theo mức độ nghiêm trọng.

Tổng cộng 3 lỗi được phát hiện, trong đó:

* 66% là 2 lỗi mức thấp (low).
* 33% là 1 lỗi mức cao (high)
* 0 lỗi mức nghiêm trọng (critical) và trung bình (medium).

Số lượng lỗi đã được sửa đã sửa 2 lỗi mức thấp và 1 lỗi mức cao

Số lượng lỗi còn tồn đọng là 0 lỗi ở bất kỳ mức độ nào.

A graph with blue bars

AI-generated content may be incorrect.

Hình 5.2. Biểu đồ cột số lượng lỗi theo mức độ nghiêm trọng Test Case.

## **3.3. Quá trình xử lý lỗi**

Quá trình xử lý lỗi được thực hiện theo các bước chuẩn, đảm bảo mọi lỗi phát hiện đều được ghi nhận, phân loại và xử lý kịp thời trong dự án. Các bước cụ thể như sau:

**Bước 1: Ghi nhận lỗi**

Khi thực hiện kiểm thử, mọi lỗi được phát hiện đều được ghi lại thông tin bao gồm: mô tả lỗi, module liên quan, Test Case/Test Script, mức độ nghiêm trọng (Severity), người phát hiện lỗi và thời điểm phát hiện.

**Bước 2: Phân loại lỗi**

Nhóm kiểm thử phân loại lỗi theo mức độ nghiêm trọng:

Phân loại giúp nhóm phát triển ưu tiên xử lý lỗi theo mức độ quan trọng.

**Bước 3: Phân công xử lý lỗi**

Lỗi được chuyển cho lập trình viên hoặc nhóm phát triển phụ trách module liên quan. Mỗi lỗi được gán trạng thái (Assigned) và người chịu trách nhiệm xử lý.

**Bước 4: Sửa lỗi**

Nhóm phát triển tiến hành sửa lỗi theo nguyên nhân đã xác định:

* Lỗi do dữ liệu: chỉnh sửa logic xử lý hoặc kiểm tra dữ liệu đầu vào.
* Lỗi giao diện: điều chỉnh giao diện, thêm chức năng thiếu.
* Lỗi trong CI/CD: sửa tên lớp, sửa code theo format Spotless theo Google Java.

**Bước 5: Kiểm thử lại**

Sau khi lỗi được sửa, nhóm kiểm thử tiến hành kiểm thử lại để xác nhận lỗi đã được khắc phục. Nếu lỗi vẫn tồn tại, quay lại bước 3 để sửa tiếp.

**Bước 6: Đóng lỗi**

Lỗi được đánh dấu Closed khi:

* Lỗi đã được sửa triệt để và kiểm thử lại thành công.
* Không còn ảnh hưởng đến các chức năng khác của hệ thống.

Tất cả các lỗi đã đóng được ghi nhận trong báo cáo tổng kết.

**Bước 7: Báo cáo và theo dõi lỗi**

Nhóm kiểm thử tổng hợp số lượng lỗi theo mức độ nghiêm trọng, trạng thái xử lý và Test Case/Test Script liên quan.

Báo cáo này được sử dụng để đánh giá chất lượng phần mềm và cải thiện quy trình kiểm thử trong các giai đoạn tiếp theo.