**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP.HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**----o0o----**

**BÁO CÁO ĐỒ ÁN CUỐI KỲ**

**KIỂM ĐỊNH PHẦN MỀM**

**CHỦ ĐỀ: TÀI LIỆU SDS CHO DỰ ÁN WEBSITE QUẢN LÝ CỬA HÀNG BÁN MÁY TÍNH**

**GVHD**: Ngô Minh Anh Thư

**Sinh viên thực hiện:**

1. 2001222539 – Nguyễn Thanh Ly Ly

2. 2001222435 – Huỳnh Hoàng Hải Long

3. 2001224956 – Lại Phước Thịnh

*Thành phố Hồ Chí Minh, tháng 04 năm 2025*

**MỤC LỤC**

[CHƯƠNG 1. TÀI LIỆU THIẾT KẾ PHẦN MỀM (SOFTWARE DESIGN SPECIFICATION) 1](#_Toc200118732)

[1.1. Giới thiệu 1](#_Toc200118733)

[1.1.1. Tổng quan 1](#_Toc200118734)

[1.1.2. Mục đích 2](#_Toc200118735)

[1.1.3. Phạm vi đối tượng 3](#_Toc200118736)

[1.2. Kiến trúc hệ thống 4](#_Toc200118737)

[1.2.1. Tổng quan 4](#_Toc200118738)

[1.2.2. Yêu cầu chương trình 5](#_Toc200118739)

[1.2.3. Cơ sở dữ liệu 5](#_Toc200118740)

[1.2.4. Cấu trúc chương trình 6](#_Toc200118741)

[1.3. Mô tả các thành phần 8](#_Toc200118742)

[1.3.1. Tổng quan 8](#_Toc200118743)

[1.3.2. Các thành phần liên quan 8](#_Toc200118744)

[1.3.3. Mô tả logic xử lý nghiệp vụ 10](#_Toc200118745)

[1.4. Thiết kế giao diện người dùng 11](#_Toc200118746)

[1.4.1. Quy tắc thiết kế 11](#_Toc200118747)

[1.4.2. Mô tả giao diện 14](#_Toc200118748)

[1.4.2.1. Giao diện cho khách hàng 14](#_Toc200118749)

[1.4.2.2. Giao diện cho Admin 26](#_Toc200118750)

[1.5. Các sơ đồ liên quan 30](#_Toc200118751)

[1.5.1. Xây dựng biểu đồ thực thể liên kết ERD 30](#_Toc200118752)

[1.5.2. Biểu đồ trình tự (Sequence diagram) 31](#_Toc200118753)

[1.5.2.1. Đăng ký 31](#_Toc200118754)

[1.5.2.2. Đăng nhập 31](#_Toc200118755)

[1.5.2.3. Xem chi tiết 32](#_Toc200118756)

[1.5.2.4. Liên hệ admin 32](#_Toc200118757)

[1.5.2.5. Thêm giỏ hàng 33](#_Toc200118758)

[1.5.2.6. Cập nhật giỏ hàng 33](#_Toc200118759)

[1.5.2.7. Thanh toán 34](#_Toc200118760)

[1.5.2.8. Cập nhật thông tin cá nhân 34](#_Toc200118761)

[1.5.2.9. Xem lịch sử đơn hàng 35](#_Toc200118762)

[1.5.3. Sơ đồ hoạt động (Activity Diagram) 36](#_Toc200118763)

[1.5.3.1. Chức năng đăng ký 36](#_Toc200118764)

[1.5.3.2. Chức năng đăng nhập 37](#_Toc200118765)

[1.5.3.3. Chức năng xem chi tiết sản phẩm 38](#_Toc200118766)

[1.5.3.4. Chức năng liên hệ với Admin 39](#_Toc200118767)

[1.5.3.5. Chức năng thêm sản phẩm vào giỏ hàng 40](#_Toc200118768)

[1.5.3.6. Chức năng cập nhật giỏ hàng 41](#_Toc200118769)

[1.5.3.7. Chức năng thanh toán 42](#_Toc200118770)

[1.5.3.8. Chức năng cập nhật thông tin cá nhân 43](#_Toc200118771)

[1.5.3.9. Chức năng xem lịch sử đơn hàng của khách hàng 44](#_Toc200118772)

[1.5.3.10. Thêm, xoá, cập nhật các thông tin của sản phẩm 45](#_Toc200118773)

[1.5.3.11. Xem chi tiết (ẩn mật khẩu người dùng), tạm ngưng tài khoản khi có vi phạm 45](#_Toc200118774)

[1.5.3.12. Kiểm tra danh sách đơn hàng của toàn bộ cửa hàng 46](#_Toc200118775)

[1.5.3.13. Biểu đồ thống kê doanh thu 46](#_Toc200118776)

[1.5.3.14. Kiểm tra danh sách các phản hồi 47](#_Toc200118777)

[CHƯƠNG 2. TỔNG KẾT VÀ HƯỚNG PHÁT TRIỂN 48](#_Toc200118778)

[2.1. Tổng kết nội dung tài liệu 48](#_Toc200118779)

[2.2. Đánh giá chất lượng tài liệu 48](#_Toc200118780)

[2.3. Hướng phát triển và triển khai 49](#_Toc200118781)

[2.4. Kết luận 49](#_Toc200118782)

[TÀI LIỆU THAM KHẢO 50](#_Toc200118783)

**DANH MỤC TỪ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **STT** | **TỪ VIẾT TẮT** | **NGHĨA CỦA TỪ VIẾT TẮT** |
| 1 | SDS | Software Design Specification: Tài liệu thiết kế phần mềm. |
| 2 | API | Application Programming Interface: Giao diện lập trình ứng dụng, cho phép tích hợp giữa các hệ thống. |
| 3 | CRUD | Create, Read, Update, Delete: Bốn thao tác cơ bản trên dữ liệu. |

**LỜI NÓI ĐẦU**

Trong bối cảnh công nghệ thông tin phát triển không ngừng và nhu cầu mua sắm trực tuyến ngày càng tăng cao, việc xây dựng và triển khai một hệ thống website đặt mua điện thoại không chỉ nhằm đáp ứng nhu cầu mua sắm của người tiêu dùng mà còn góp phần thúc đẩy sự phát triển của thương mại điện tử tại địa phương và trên toàn quốc. Báo cáo tài liệu SDS (Software Design Specification – Quy định Thiết kế Phần mềm) được xây dựng với mục tiêu trình bày rõ ràng các yêu cầu, chức năng và kiến trúc thiết kế của hệ thống, từ đó tạo nền tảng vững chắc cho quá trình phát triển và vận hành website.

Tài liệu SDS tập trung mô tả chi tiết các giải pháp thiết kế kỹ thuật và cách thức triển khai nhằm đảm bảo hệ thống hoạt động ổn định và bảo trì dễ dàng trong quá trình vận hành thực tế.

Quá trình xây dựng tài liệu này được thực hiện dựa trên các nguyên tắc thiết kế phần mềm hiện đại, phù hợp với tiêu chuẩn công nghiệp và các yêu cầu của người sử dụng. Chúng tôi ghi nhận sự đóng góp, hợp tác và động viên nhiệt tình của các chuyên gia, kỹ sư phần mềm và các bên liên quan trong quá trình thu thập thông tin, phân tích yêu cầu và đưa ra giải pháp tối ưu cho hệ thống.

Mục đích của báo cáo này là cung cấp một cái nhìn tổng quan và đầy đủ nhất về hệ thống website đặt mua điện thoại, từ góc độ kinh doanh lẫn kỹ thuật. Qua đó, tài liệu sẽ là cơ sở tham khảo chính cho các nhà phát triển, các chuyên gia kiểm thử, và ban quản lý dự án trong suốt quá trình phát triển, kiểm thử và triển khai sản phẩm. Chúng tôi hy vọng rằng, thông qua việc nghiên cứu, triển khai và ứng dụng các đề xuất trong báo cáo, hệ thống sẽ đem lại trải nghiệm mua sắm trực tuyến tiện lợi, an toàn và hiệu quả cho khách hàng, đồng thời tạo ra giá trị thiết thực cho doanh nghiệp và cộng đồng.

# CHƯƠNG 1. TÀI LIỆU THIẾT KẾ PHẦN MỀM (SOFTWARE DESIGN SPECIFICATION)

Website Bán máy tính (Mô hình MVC)

Ngày: 012/04/2025

Nhóm thực hiện: Nhóm 6

## ****1.1. Giới thiệu****

Software Design Specification (SDS) là tài liệu trình bày chi tiết giải pháp thiết kế kỹ thuật của một hệ thống phần mềm, dựa trên các yêu cầu đã được xác định trong tài liệu SRS (Software Requirements Specification). SDS mô tả cách thức kiến trúc hệ thống được tổ chức, cách các module tương tác, cấu trúc dữ liệu, thuật toán, giao diện và các ràng buộc phi chức năng (hiệu năng, bảo mật, khả năng mở rộng…) nhằm hướng dẫn nhóm phát triển hiện thực và nhóm kiểm thử đảm bảo sản phẩm đúng như mong đợi.

### 1.1.1. Tổng quan

Trong tài liệu SDS này, các quy ước được áp dụng nhằm đảm bảo tính nhất quán, dễ tra cứu và bảo trì. Tài liệu được chia thành năm chương chính: Giới thiệu, Mô tả Kiến trúc Hệ thống, Mô tả Chi tiết Thành phần, Thiết kế Giao diện Người dùng và Phụ lục (bao gồm sơ đồ UML, wireframe và bảng dữ liệu). Mỗi chương–mục–tiểu mục đều được đánh số theo định dạng X.Y.Z (ví dụ 3.4.2) và bắt đầu bằng “Mục lục chương” liệt kê các mục con. Sơ đồ UML đánh số “Sơ đồ X.Y” (ví dụ Sơ đồ 2.1), hình ảnh và wireframe đánh số “Hình X.Y”, bảng dữ liệu đánh số “Bảng X.Y”, với tiêu đề in nghiêng đặt ngay bên dưới.

Toàn bộ nội dung sử dụng font Times New Roman cỡ 13, giãn dòng 1.5; tiêu đề chương cỡ 16 in đậm; tiêu đề mục cỡ 14 in đậm; đoạn code hoặc pseudo‑code trình bày bằng font Consolas cỡ 11, khung viền nhạt. Tất cả thuật ngữ chuyên ngành và từ viết tắt (MVC, API, UI, UX…) đều in hoa và được định nghĩa, chú giải trong phần “Thuật ngữ” đầu tài liệu. Danh sách không thứ tự dùng dấu gạch ngang, danh sách có thứ tự dùng số. Mọi tham chiếu tới tài liệu ngoài tuân theo chuẩn IEEE, đánh số trong ngoặc vuông và liệt kê trong mục Tài liệu Tham khảo cuối SDS.

### 1.1.2. Mục đích

Mục đích của tài liệu Software Design Specification (SDS) cho hệ thống website nhằm:

* **Chỉ dẫn hiện thực**: Cung cấp bản thiết kế chi tiết về kiến trúc phân lớp (Model–View–Controller), các thành phần module và cách thức chúng tương tác với nhau. Từ đó, lập trình viên có thể chuyển hóa trực tiếp vào mã nguồn một cách nhất quán, tránh sai sót và lặp lại công việc.
* **Đồng bộ hoá nhóm phát triển**: Định nghĩa rõ ràng các interface giữa Controller và View, giữa Controller và Model, bao gồm tên phương thức, tham số đầu vào, kết quả trả về. Điều này giúp nhiều thành viên làm việc song song trên các module khác nhau mà không vướng mắc về cách gọi hay trao đổi dữ liệu.
* **Làm cơ sở cho kiểm thử**: Mô tả chi tiết luồng dữ liệu, luồng xử lý cho từng chức năng (ví dụ: thêm sản phẩm vào giỏ, thanh toán, quản lý đơn hàng), cũng như các ràng buộc đầu vào/đầu ra. Tester sẽ dựa vào đây để thiết kế test case, kiểm tra tính đúng đắn và đầy đủ của từng module.
* **Hỗ trợ bảo trì & mở rộng**: Ghi chép kiến trúc, bối cảnh hoạt động, các thuật toán quan trọng và các ràng buộc phi chức năng (ví dụ: hiệu năng, bảo mật, khả năng mở rộng). Khi cần bổ sung tính năng mới hoặc tối ưu hiệu năng, đội ngũ bảo trì sẽ dễ dàng xác định điểm chặn và thực hiện sửa đổi một cách an toàn.
* **Làm tài liệu tham khảo cho triển khai**: Cung cấp hướng dẫn cài đặt, cấu hình môi trường (IIS, kết nối database, cấu hình dependency injection…), giúp nhóm vận hành triển khai hệ thống nhanh chóng, chính xác và nhất quán giữa các môi trường (dev, test, staging, production).

→ Qua đó, tài liệu SDS không chỉ là “bản đồ thiết kế” cho giai đoạn phát triển mà còn là nền tảng để đảm bảo chất lượng, tính mở rộng và khả năng vận hành bền vững cho website bán laptop xây dựng trên nền tảng ASP.NET MVC.

### 1.1.3. Phạm vi đối tượng

* **Tính năng người dùng (Front‑end):**
* Xem danh mục sản phẩm (máy tính để bàn, laptop, linh kiện, phụ kiện).
* Tìm kiếm, lọc và sắp xếp sản phẩm theo giá, thương hiệu, thông số kỹ thuật.
* Xem chi tiết sản phẩm với hình ảnh, thông số, đánh giá và tồn kho.
* Thêm/xóa/cập nhật sản phẩm trong giỏ hàng.
* Thực hiện quy trình thanh toán (chọn địa chỉ, phương thức thanh toán, xác nhận đơn hàng).
* Đăng ký, đăng nhập, quản lý thông tin cá nhân và lịch sử đơn hàng.
* **Tính năng quản trị (Back‑office):**
* Quản lý danh mục và thêm/sửa/xóa sản phẩm.
* Quản lý đơn hàng (xem, xác nhận, hủy, cập nhật trạng thái).
* Quản lý người dùng (phân quyền Admin/User, khóa/mở tài khoản).
* Báo cáo đơn hàng, doanh thu theo ngày, tháng, năm.
* **Kiến trúc MVC và tích hợp:**
* **Models:** Thiết kế các lớp đối tượng đại diện cho sản phẩm, đơn hàng, người dùng, giỏ hàng.
* **Views:** Các trang Razor view cho giao diện người dùng và quản trị.
* **Controllers:** Xử lý luồng ứng dụng, tương tác với service layer và trả về view hoặc JSON (cho AJAX).
* **Data Access Layer:** Sử dụng Entity Framework cho CRUD lên cơ sở dữ liệu SQL Server.
* **API endpoints:** Định nghĩa RESTful API cho các thao tác AJAX (thêm giỏ hàng, load nhanh dữ liệu).
* **Wireframes & UML:** Bao gồm sơ đồ lớp, sequence, activity và wireframe cho các màn hình quan trọng.
* **Yêu cầu phi chức năng:**
* **Hiệu năng:** Thời gian phản hồi trang tối đa 2 giây dưới tải 100 người dùng đồng thời.
* **Bảo mật:** Xác thực JWT/session, mã hóa SSL, ngăn chặn XSS, CSRF, SQL Injection.
* **Khả năng mở rộng:** Hỗ trợ tách service layer, dễ dàng bổ sung module mới (ví dụ: khuyến mãi, hệ thống điểm tích lũy).
* **Khả năng bảo trì:** Logging chi tiết, cơ chế exception handling và báo cáo lỗi.

## 1.2. Kiến trúc hệ thống

Sẽ tóm lược cách các thành phần (frontend, backend, database, dịch vụ ngoài) kết nối và tương tác, phân tầng thành Presentation, Business Logic, Data Access, miêu tả module (MD‑xxx), công nghệ (MVC, EF, DI, REST), cùng luồng dữ liệu qua sơ đồ sequence. Đồng thời nêu cơ chế phi chức năng (caching, logging, bảo mật, exception handling) để đảm bảo hiệu năng, mở rộng và ổn định hệ thống.

### 1.2.1. Tổng quan

Phần này sẽ cung cấp cái nhìn khái quát về cách thức tổ chức và vận hành của website bán máy tính và phụ kiện MVC. Kiến trúc được phân tầng thành Presentation (giao diện), Business Logic (xử lý nghiệp vụ), Data Access (truy xuất dữ liệu) và Integration (tích hợp dịch vụ ngoài). Các module chính được định danh bằng mã MD‑xxx, áp dụng pattern MVC, Repository và Dependency Injection, sử dụng Entity Framework cho ORM và RESTful API cho tương tác Ajax. Đồng thời, cũng nhấn mạnh các cơ chế phi chức năng như caching, logging, bảo mật và exception handling để đảm bảo hệ thống đạt yêu cầu về hiệu năng, khả năng mở rộng và độ ổn định.

### 1.2.2. Yêu cầu chương trình

* Ngôn ngữ lập trình: ASP.NET với C#
* Kiến trúc: Mô hình MVC (Model - View - Controller)
* Cơ sở dữ liệu: SQL Server (file .bacpac được sử dụng để triển khai)
* Hệ điều hành triển khai: Windows, Linux.
* Yêu cầu trình duyệt: Google Chrome, Microsoft Edge, Firefox.
* Thiết bị hỗ trợ: PC, Laptop.

### 1.2.3. Cơ sở dữ liệu

Các bảng chính trong hệ thống bao gồm:

1. **DanhMuc (Danh mục sản phẩm)**
   * DanhMucID (PK)
   * TenDanhMuc
2. **MT (Thông tin máy tính/laptop)**
   * MTID (PK)
   * TenSP, MoTa, Gia, SoLuongKho, ThuongHieu, ViXuLy, RAM, LuuTru, KichThuocManHinh, MauSac, DungLuongPin
   * DuongDan (liên kết hình ảnh sản phẩm)
   * DanhMucID (FK → DanhMuc)
3. **KhachHang (Người dùng)**
   * KhachHangID (PK)
   * TenKhachHang, Email, MatKhau, SoDienThoai, DiaChi
4. **DonHang (Đơn hàng)**
   * DonHangID (PK)
   * KhachHangID (FK)
   * NgayDatHang, TongTien, TrangThaiDonHang, PhuongThucThanhToan, DiaChiGiaoHang
5. **CTDonHang (Chi tiết đơn hàng)**
   * DonHangID (PK, FK)
   * MTID (PK, FK)
   * SoLuong, Gia
6. **LienHe**
   * IDLienHe (PK)
   * KhachHangID (FK)
   * NoiDung

### 1.2.4. Cấu trúc chương trình

Website bán laptop được xây dựng dựa trên kiến trúc MVC (Model - View - Controller) giúp tách biệt rõ ràng giữa giao diện người dùng, logic xử lý và truy xuất dữ liệu, từ đó dễ bảo trì, mở rộng và phát triển.

**1. Model (Mô hình dữ liệu)**

Đây là lớp chịu trách nhiệm quản lý dữ liệu và nghiệp vụ chính của hệ thống. Model tương tác với cơ sở dữ liệu để thực hiện các thao tác như truy vấn, thêm, sửa, xoá thông tin.

Các model chính gồm:

* Laptop: Thông tin về sản phẩm (tên, giá, cấu hình, hình ảnh…)
* DanhMuc: Phân loại sản phẩm (Dell, HP, Asus…)
* TaiKhoan: Tài khoản người dùng
* GioHang: Giỏ hàng của người dùng
* DonHang: Thông tin đơn hàng
* ChiTietDonHang: Chi tiết sản phẩm trong đơn hàng
* BinhLuan: Đánh giá, nhận xét sản phẩm

**2. View (Giao diện)**

Là phần hiển thị dữ liệu ra trình duyệt để người dùng tương tác. View được viết bằng HTML, Razor, CSS và có thể sử dụng JavaScript để tăng tính năng động.

Một số view tiêu biểu:

* TrangChu.cshtml: Trang chủ hiển thị sản phẩm nổi bật
* ChiTietLaptop.cshtml: Chi tiết sản phẩm
* GioHang.cshtml: Xem giỏ hàng
* ThanhToan.cshtml: Thực hiện đặt hàng
* DangNhap.cshtml, DangKy.cshtml: Giao diện xác thực người dùng

**3. Controller (Bộ điều khiển)**

Controller là nơi tiếp nhận yêu cầu từ trình duyệt, xử lý logic, tương tác với model và trả về view phù hợp.

Một số controller chính:

* LaptopController: Quản lý hiển thị danh sách và chi tiết laptop
* GioHangController: Quản lý giỏ hàng (thêm, xoá, cập nhật)
* ThanhToanController: Quản lý quá trình đặt hàng
* TaiKhoanController: Quản lý đăng nhập, đăng ký
* AdminController: Quản lý sản phẩm, danh mục, đơn hàng (dành cho quản trị viên)

**4. Các thành phần hỗ trợ khác**

* **Service Layer (nếu có)**: Tầng trung gian xử lý logic nghiệp vụ trước khi đến Controller.
* **Data Access Layer (DAL)**: Xử lý giao tiếp trực tiếp với cơ sở dữ liệu, thường dùng Entity Framework.
* **Layout và Partial View**: Tạo các phần dùng chung như header, footer, menu.
* **RouteConfig**: Cấu hình định tuyến URL.
* **Authentication & Authorization**: Xác thực người dùng và phân quyền truy cập.

→ Kiến trúc này giúp hệ thống web bán laptop được tổ chức rõ ràng, dễ mở rộng, dễ bảo trì và cho phép phát triển các chức năng phức tạp mà vẫn giữ được tính linh hoạt trong triển khai.

## 1.3. Mô tả các thành phần

Sẽ trình bày chi tiết các thành phần cốt lõi của hệ thống:

### 1.3.1. Tổng quan

Bao gồm các lớp trong mô hình “Model” (quản lý dữ liệu và nghiệp vụ), các “Controller” (điều phối yêu cầu và xử lý logic), cùng với các “thành phần giao diện” và “dịch vụ xử lý chức năng nghiệp vụ”.

Mỗi thành phần sẽ được mô tả rõ ràng theo các khía cạnh: tên thành phần, vai trò chính trong hệ thống, chức năng thực hiện, luồng xử lý dữ liệu, cũng như mối quan hệ và tương tác với các thành phần khác. Mục tiêu của phần này là làm rõ cấu trúc triển khai và đảm bảo mọi bên liên quan hiểu được cách hoạt động tổng thể của hệ thống website bán laptop và phụ kiện.

### 1.3.2. Các thành phần liên quan

Dự án website bán laptop sử dụng mô hình MVC, có các thành phần chính như sau:

**1. Controllers (Điều phối logic và xử lý yêu cầu):**

* GioHangController.cs: Xử lý các chức năng liên quan đến giỏ hàng như thêm, xóa, xác nhận đơn hàng.
* KhachHangController.cs: Quản lý các chức năng của khách hàng như đăng ký, đăng nhập, cập nhật thông tin cá nhân, và quản lý đơn hàng.
* MayTinhController.cs: Xử lý các thao tác thêm, sửa, xóa, xem chi tiết sản phẩm máy tính.

**2. Models (Lớp dữ liệu và nghiệp vụ):**

* DonHang.cs: Mô hình dữ liệu đơn hàng.
* Forgot.cs: Xử lý chức năng quên mật khẩu.
* GioHang.cs: Mô hình dữ liệu của giỏ hàng.
* KhachHangViewModel.cs: ViewModel hỗ trợ việc hiển thị dữ liệu khách hàng.
* XacNhanDonHangViewModel.cs: ViewModel phục vụ quá trình xác nhận đơn hàng.
* SanPham.edmx: Mô hình cơ sở dữ liệu Entity Framework để quản lý bảng sản phẩm và các bảng liên quan.

**3. Views (Giao diện người dùng - Razor View .cshtml):**

* **Thư mục GioHang:**
* GioHang.cshtml: Hiển thị giỏ hàng hiện tại.
* ThemGioHang.cshtml: Giao diện thêm sản phẩm vào giỏ hàng.
* XacNhanDonHang.cshtml: Giao diện xác nhận đơn hàng.
* **Thư mục KhachHang:**
* DanhSachDonHang.cshtml: Hiển thị danh sách đơn hàng của khách.
* ThongTin.cshtml: Xem và cập nhật thông tin cá nhân.
* XemDonHang.cshtml: Chi tiết từng đơn hàng.
* **Thư mục MayTinh:**
* Add.cshtml: Thêm sản phẩm mới.
* Delete.cshtml: Xóa sản phẩm.
* Details.cshtml: Xem chi tiết sản phẩm.
* DSDonHangDetails.cshtml: Chi tiết đơn hàng liên quan đến sản phẩm.
* DSDonHang.cshtml: Danh sách đơn hàng.
* **Giao diện chung khác:**
* Index.cshtml, Index1.cshtml: Trang chủ.
* Login.cshtml, Register.cshtml, Forgot.cshtml: Đăng nhập, đăng ký, quên mật khẩu.
* LienHe.cshtml, GioiThieu.cshtml, Maps.cshtml: Trang giới thiệu, liên hệ, bản đồ.
* Edit.cshtml, test.cshtml, DSPhanhoi.cshtml: Các trang phụ hoặc chức năng mở rộng.

### 1.3.3. Mô tả logic xử lý nghiệp vụ

* **Tính tổng tiền:** Tổng = SUM(Gia \* SoLuong)
* **Cập nhật tồn kho:** Khi đặt hàng, số lượng sản phẩm trong MT.SoLuongKho giảm tương ứng
* **Xác thực người dùng:** Kiểm tra Email + MậtKhau khi đăng nhập
* **Kiểm tra tồn kho:** Trước khi đặt hàng, kiểm tra số lượng còn lại trong kho

## 1.4. Thiết kế giao diện người dùng

Thông qua các bản thiết kế giao diện, nhóm có thể thống nhất cách hiển thị nội dung, đảm bảo tính thẩm mỹ, tính dễ sử dụng và trải nghiệm người dùng tốt. Ngoài ra, phần này còn giúp dễ dàng triển khai đúng yêu cầu, và giúp người dùng đối chiếu để kiểm tra tính đúng đắn của sản phẩm sau khi phát triển.

### 1.4.1. Quy tắc thiết kế

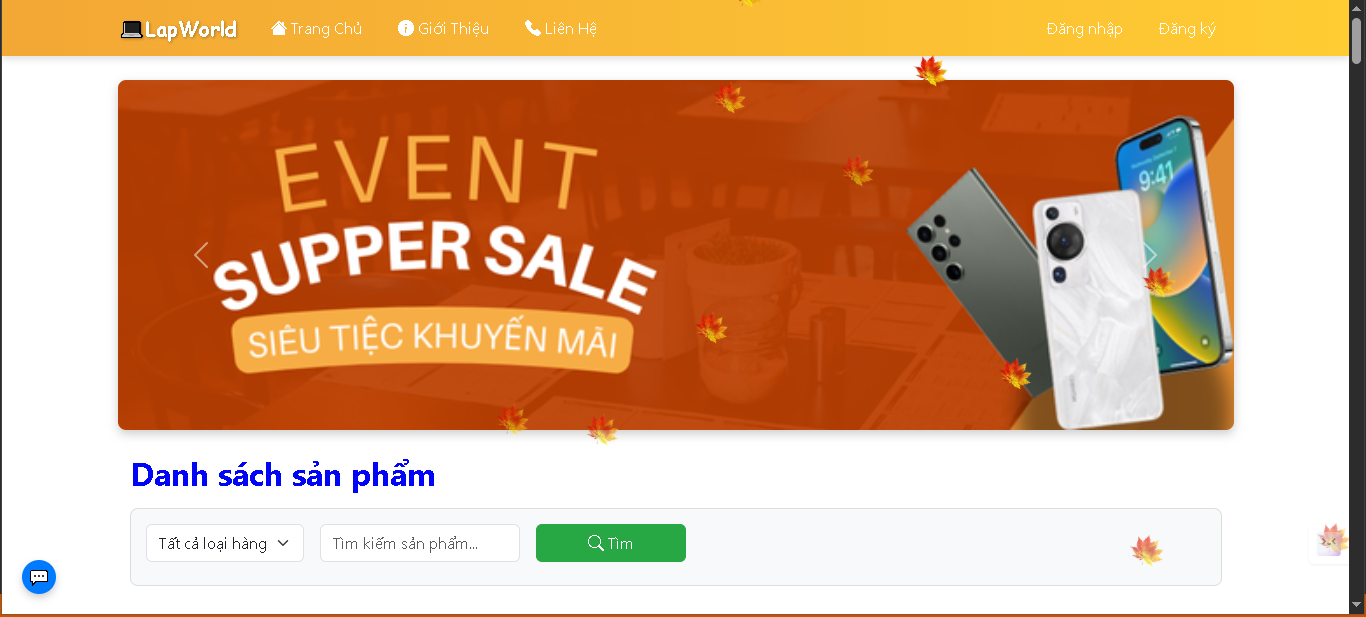
* **Kiến trúc MVC (Model-View-Controller)**
* **Model (Mô hình):**  
  Quản lý dữ liệu, logic kinh doanh, và các truy vấn đến cơ sở dữ liệu (sản phẩm, đơn hàng, người dùng…).
* **View (Giao diện):**  
  Sử dụng HTML5, CSS3 và Bootstrap để tạo giao diện hiện đại, đẹp mắt và dễ bảo trì.
* **Controller (Bộ điều khiển):**  
  Xử lý các yêu cầu từ người dùng, liên lạc với model để lấy dữ liệu và chuyển dữ liệu đó sang view.
* **Chuẩn thiết kế web hiện đại**
* **HTML5, CSS3:**
  + Sử dụng các thẻ sematic của HTML5 (header, nav, section, footer) nhằm cải thiện SEO và khả năng truy cập.
  + CSS3 cho phép sử dụng các hiệu ứng chuyển đổi (transitions), animation, media queries để tạo giao diện động và responsive.
* **Bootstrap:**
  + Tích hợp Bootstrap để nhanh chóng xây dựng giao diện với lưới (grid system) và các thành phần giao diện (buttons, forms, navbars…).
  + Tùy chỉnh giao diện theo tông màu chủ đạo bằng cách ghi đè các biến Sass của Bootstrap.
* **Giao diện Responsive – tự động co giãn hiển thị trên các thiết bị**
* **Media Queries:**  
  Sử dụng CSS media queries để đảm bảo giao diện hiển thị tốt trên cả desktop, tablet và mobile.
* **Lưới linh hoạt:**  
  Dựa vào hệ thống lưới (grid system) của Bootstrap để bố trí các thành phần tự động co giãn, sắp xếp lại khi kích thước màn hình thay đổi.
* **Responsive Images:**  
  Sử dụng thuộc tính img-fluid của Bootstrap hoặc CSS max-width: 100% để hình ảnh tự động điều chỉnh theo kích thước container.
* **Màu sắc, font chữ và phong cách thiết kế**
* **Màu sắc:**
  + Chủ đạo theo tông trắng – đen – pastel, tạo cảm giác thanh lịch và dễ chịu.
  + Sử dụng màu sắc tương phản cho các phần nổi bật (ví dụ: nút hành động, tiêu đề sản phẩm) để thu hút sự chú ý.
* **Font chữ:**
  + Chọn font sans-serif dễ đọc (như Roboto, Open Sans hoặc Helvetica) vì chúng tương thích tốt trên đa số trình duyệt và thiết bị.
  + Dùng kích thước chữ hợp lý, khoảng cách dòng và margin phù hợp để cải thiện khả năng đọc.
* **Phong cách thiết kế (Style):**
  + Thiết kế phải gọn gàng, hiện đại, ưu tiên sự tối giản nhằm làm nổi bật sản phẩm bán.
  + Tận dụng khoảng trắng (white space) để tạo không gian cho từng thành phần giao diện, giúp người dùng dễ dàng định vị và tương tác.
* **Trải nghiệm người dùng (UX)**
* **Điều hướng đơn giản, rõ ràng:**
  + Menu điều hướng, thanh tìm kiếm và các liên kết chính phải có vị trí nổi bật.
  + Sử dụng breadcrumbs để người dùng dễ định vị vị trí hiện tại trên website.
* **Hiển thị thông tin sản phẩm nổi bật:**
  + Trình bày thông tin sản phẩm (hình ảnh, mô tả ngắn, giá bán, đánh giá) một cách rõ ràng, sử dụng các card hoặc grid để hiển thị sản phẩm.
* **Nút thao tác dễ nhận biết:**
  + Các nút “Thêm vào giỏ”, “Mua ngay” và “Tìm kiếm” được thiết kế nổi bật với màu sắc đối lập và hiệu ứng hover mượt mà.
  + Dùng biểu tượng, icon liên quan kèm theo văn bản để tăng độ nhận diện.
* **Xác nhận hành động quan trọng:**
  + Hiển thị hộp thoại xác nhận khi người dùng thực hiện các thao tác nhạy cảm như xóa sản phẩm khỏi giỏ hàng hoặc thanh toán.
  + Xác nhận này có thể được thực hiện bằng modal dialog của Bootstrap, đảm bảo giao diện thống nhất và thân thiện.
* **Áp dụng các quy tắc vào hệ thống MVC cho web bán hàng điện thoại**
* **Model (Mô hình):**
* Định nghĩa các lớp đối tượng: Product, Cart, Order, User…
* Sử dụng cơ sở dữ liệu quan hệ (ví dụ: MySQL, PostgreSQL) với các bảng dữ liệu riêng cho sản phẩm, đơn hàng, và người dùng.
* Model chịu trách nhiệm xử lý các phép toán kiểm tra, tính toán giá, khuyến mãi theo quy tắc kinh doanh.
* **View (Giao diện):**
* **HTML5/CSS3 & Bootstrap:**  
  Tạo các view template cho trang chủ, danh mục sản phẩm, chi tiết sản phẩm, giỏ hàng và thanh toán.
* **Responsive:**  
  Các view sử dụng lưới (grid) của Bootstrap để tự động sắp xếp nội dung theo kích thước màn hình.
* **Theme và Styling:**  
  Áp dụng các quy tắc màu sắc (trắng, đen, pastel) và font chữ sans-serif thông qua CSS tùy chỉnh hoặc sử dụng các framework CSS như Bootstrap + Sass để ghi đè style.
* **UX:**
* Sử dụng modal của Bootstrap để xác nhận hành động xóa hoặc thanh toán.
* Các nút và biểu tượng cần có hiệu ứng hover để phản hồi hành động người dùng.
* **Controller (Bộ điều khiển):**
* Xử lý các yêu cầu từ giao diện, ví dụ khi người dùng nhấn “Thêm vào giỏ” hoặc “Mua ngay”.
* Điều phối luồng dữ liệu giữa Model và View.
* Triển khai các chức năng tìm kiếm, lọc sản phẩm và phân trang.

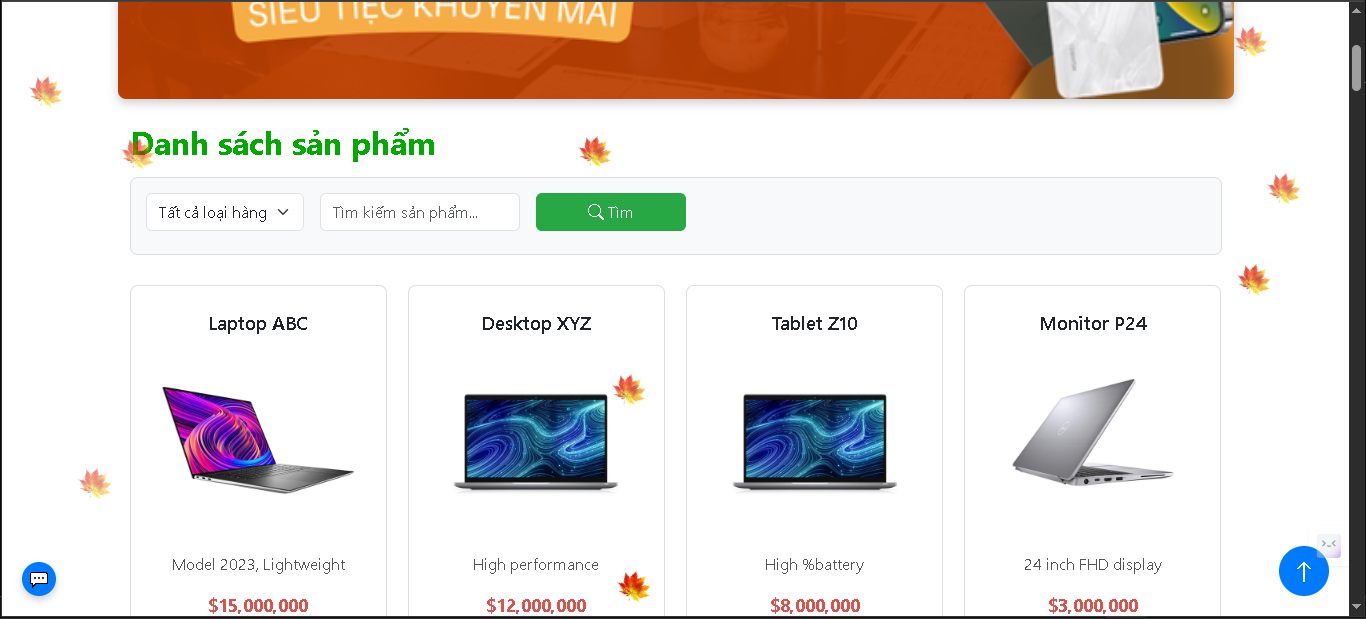
### 1.4.2. Mô tả giao diện

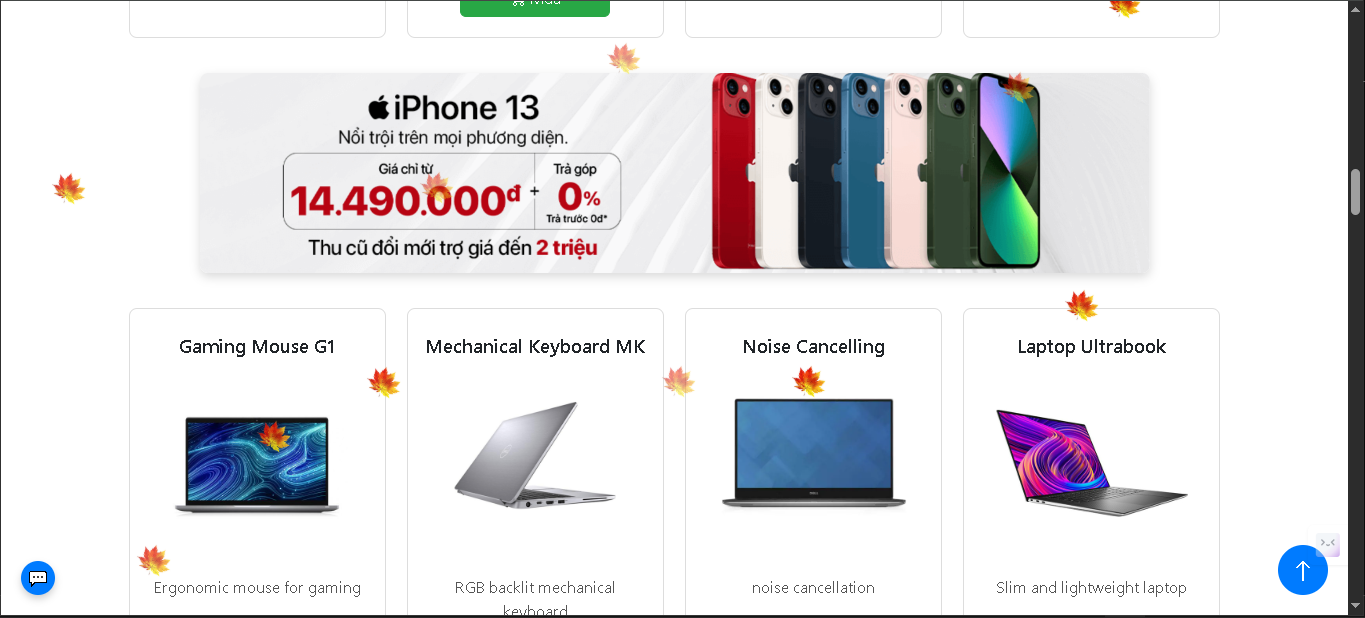
#### **1.4.2.1. Giao diện cho khách hàng**

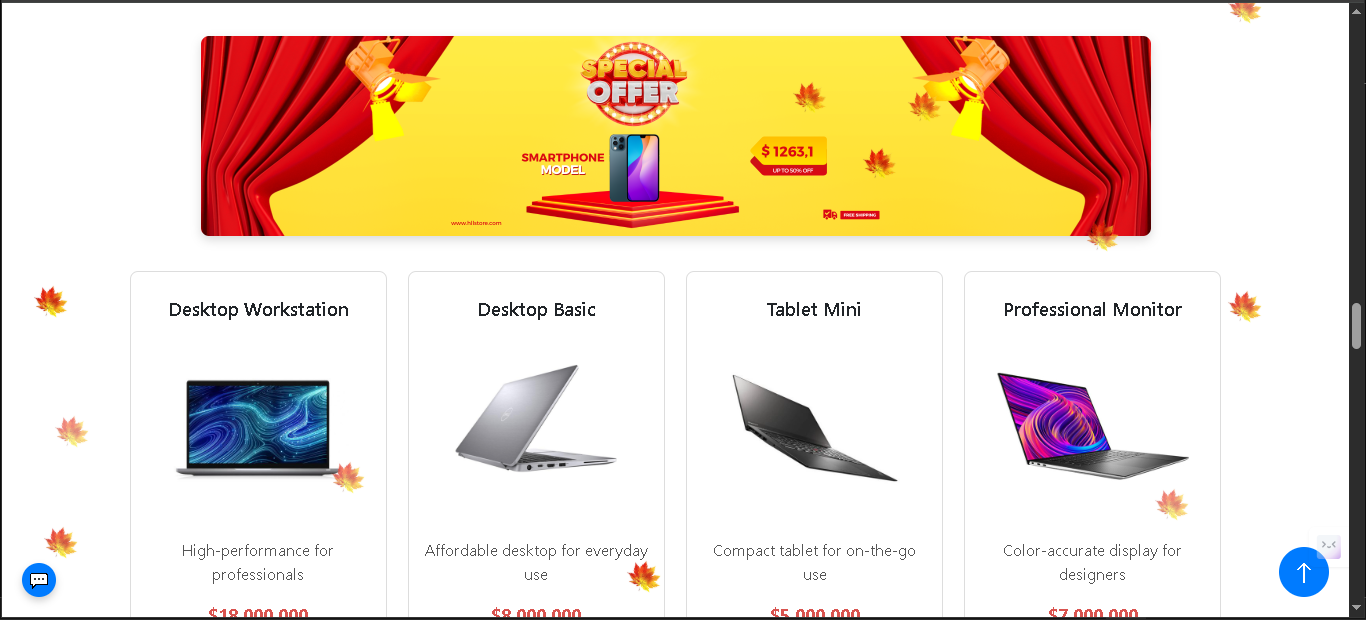
##### **1.4.2.1.1. Trang chủ (Home Page)**

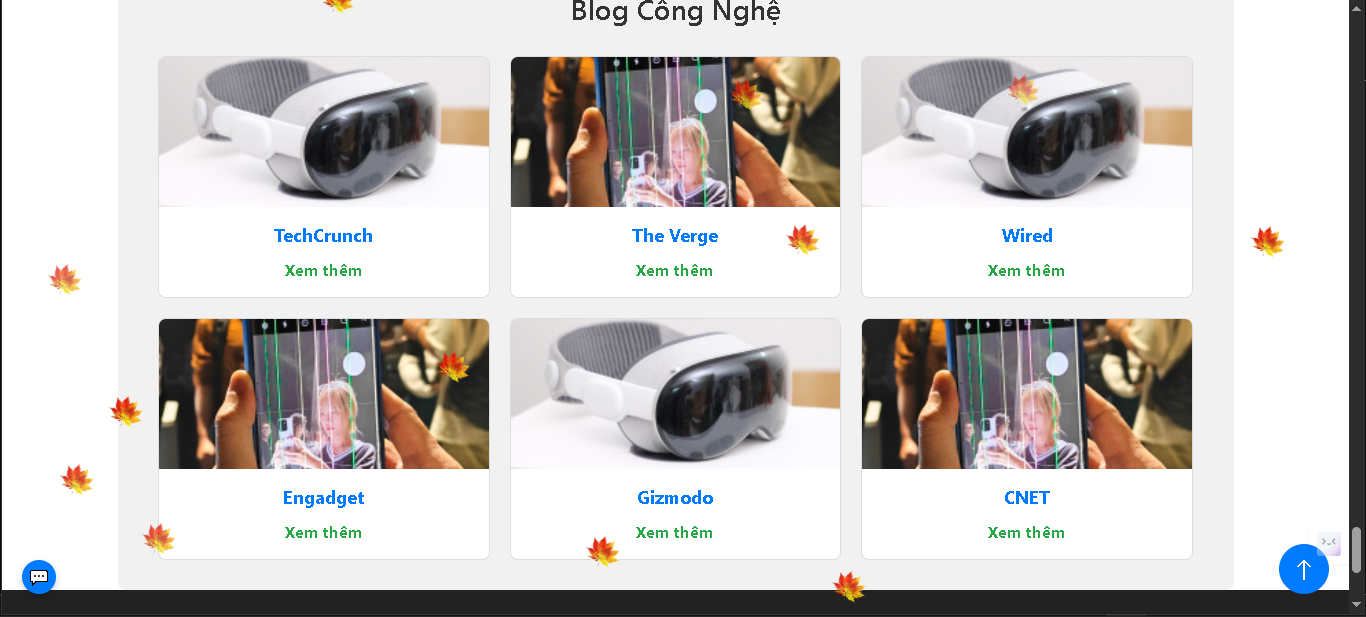
* **Mục đích chính:** Hiển thị các sản phẩm cho người dùng chọn mua.
* Hiển thị logo, thanh điều hướng (menu)
* Danh sách danh mục
* Vùng hiển thị danh sách sản phẩm nổi bật/laptop mới
* Hiển thị tất cả sản phẩm theo danh mục
* Mỗi sản phẩm có ảnh, tên, giá, và các nút "Xem chi tiết", "Mua ngay"
* Tìm kiếm, lọc theo RAM, CPU, thương hiệu

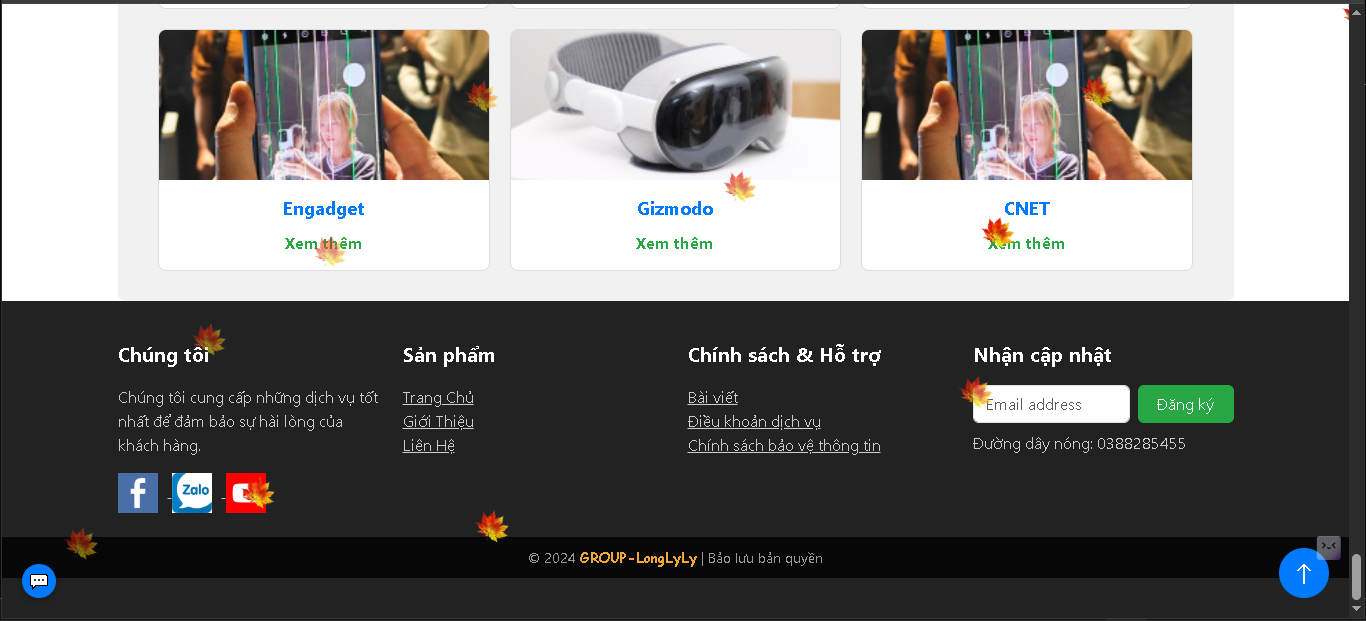






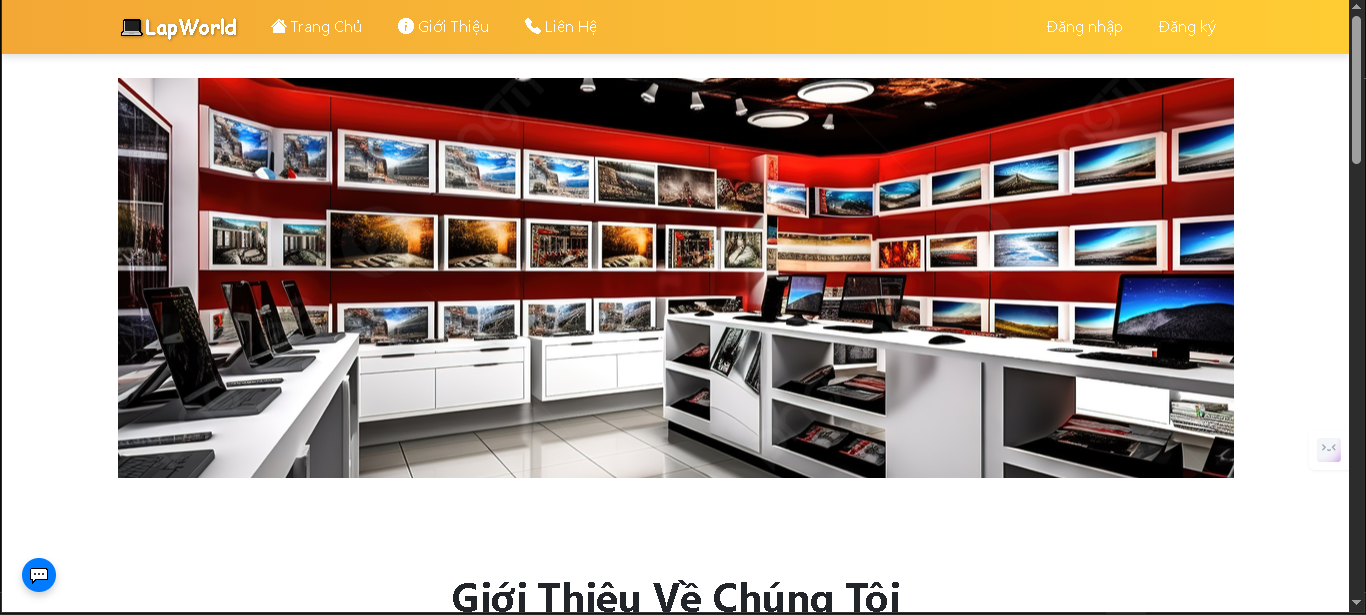






##### **1.4.2.1.2. Trang giới thiệu cửa hàng**

* **Mục đích chính:** hiển thị các thông tin chi tiết về cửa hàng.
* Hiển thị logo, thanh điều hướng (menu)
* Hiển thị các đoạn text mô tả cửa hàng



A person standing in front of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

##### **1.4.2.1.3. Trang liên hệ với quản trị viên**

* **Mục đích chính:** Giúp người dùng có thể liên lạc với quản trị viên khi có vấn đề phát sinh.
* Hiển thị một text box nhập nội dung và nút gửi cho admin
* Một frame Google Map hiển thị địa chỉ giúp người dùng dễ theo dõi
* Biểu mẫu gửi phản hồi / góp ý
* Lưu thông tin vào bảng LienHe

A screenshot of a phone

AI-generated content may be incorrect.

A screenshot of a computer

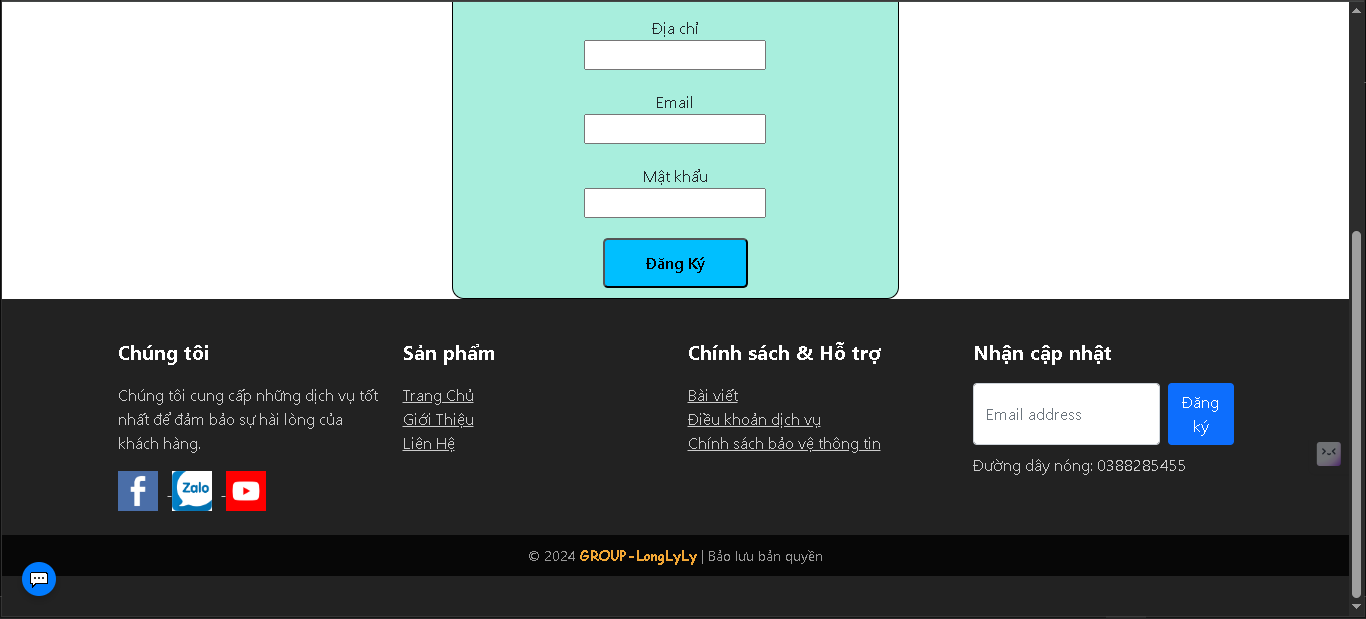
AI-generated content may be incorrect.

##### **1.4.2.1.4. Trang đăng ký tài khoản**

* **Mục đích chính:** Tạo tài khoản giúp người dùng mua hàng.
* Hiển thị các ô text box cho phép nhập các thông tin tương ứng và nút đăng ký để tạo tài khoản
* Kiểm tra email/mật khẩu hợp lệ

A screenshot of a computer

AI-generated content may be incorrect.



##### **1.4.2.1.5. Trang đăng nhập**

* **Mục đích chính:** giúp người dùng đăng nhập vào tài khoản đã có để mua hàng.
* Hiển thị các ô text box cho phép nhập các thông tin tương ứng và nút đăng nhập để vào tài khoản
* Kiểm tra email/mật khẩu hợp lệ

A screenshot of a computer

AI-generated content may be incorrect.

##### **1.4.2.1.6. Trang “quên mật khẩu”**

* **Mục đích chính:** Giúp người dùng khôi phục lại mật khẩu trong trường hợp khách hàng bị quên.
* Hiển thị 1 text box cho phép nhập email của tài khoản khách hàng, sau đó admin sẽ gửi lại mật khẩu tài khoản vào email mà khách hàng đã đăng ký

A screenshot of a computer

AI-generated content may be incorrect.

##### **1.4.2.1.7. Trang giỏ hàng**

* **Mục đích chính:** Kiểm tra lại các sản phẩm dự định mua.
* Hiển thị các sản phẩm đã chọn
* Tùy chọn cập nhật số lượng hoặc xóa
* Tổng tiền

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

##### **1.4.2.1.8. Trang thanh toán**

* **Mục đích chính:** Kiểm tra lại thông tin đơn mua.
* Nhập địa chỉ giao hàng
* Chọn phương thức thanh toán
* Xác nhận đặt hàng

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

##### **1.4.2.1.9.** **Trang cập nhật thông tin tài khoản khách hàng**

* **Mục đích chính:** Giúp khách hàng kiểm tra lại thông tin tài khoản và cập nhật nếu cần.
* Xem lịch sử mua hàng.
* Nhập các thông tin tài khoản vào các text box tương ứng và nút cập nhật.
* Có hiển thị chức năng xem lịch sử mua hàng

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

##### **1.4.2.1.10. Trang xem lịch sử đơn hàng**

* **Mục đích chính:** Giúp khách hàng kiểm tra lại danh sách đơn hàng.

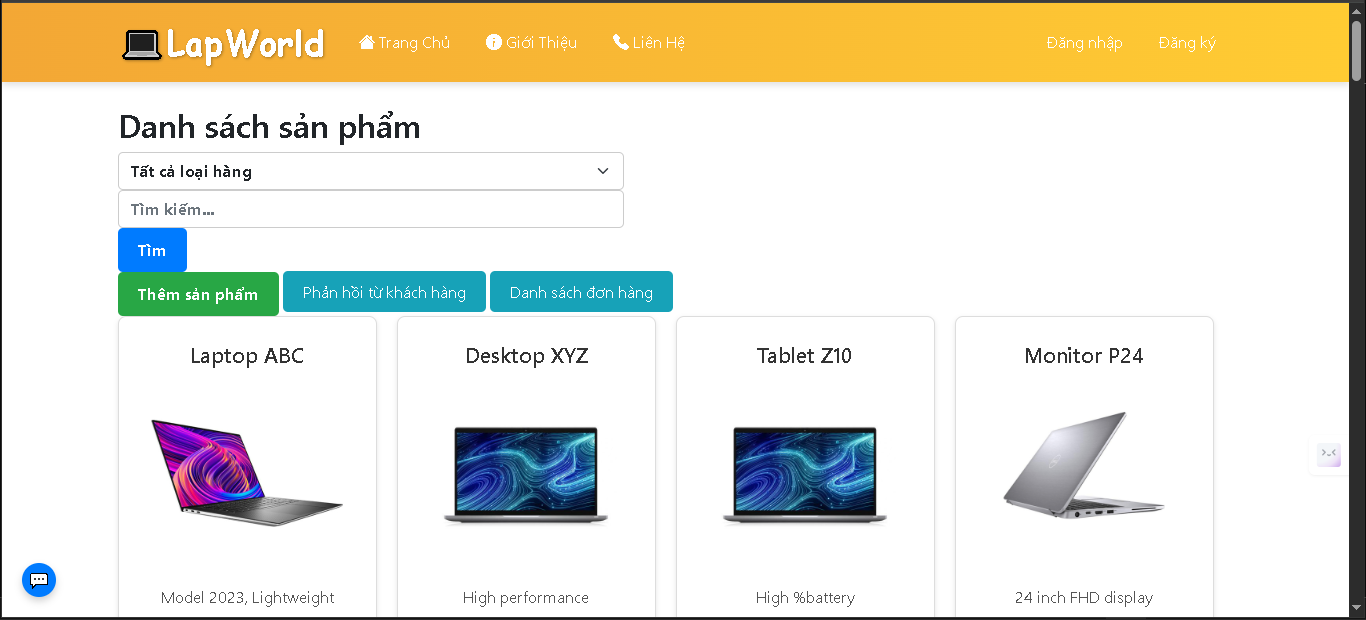
A screenshot of a computer

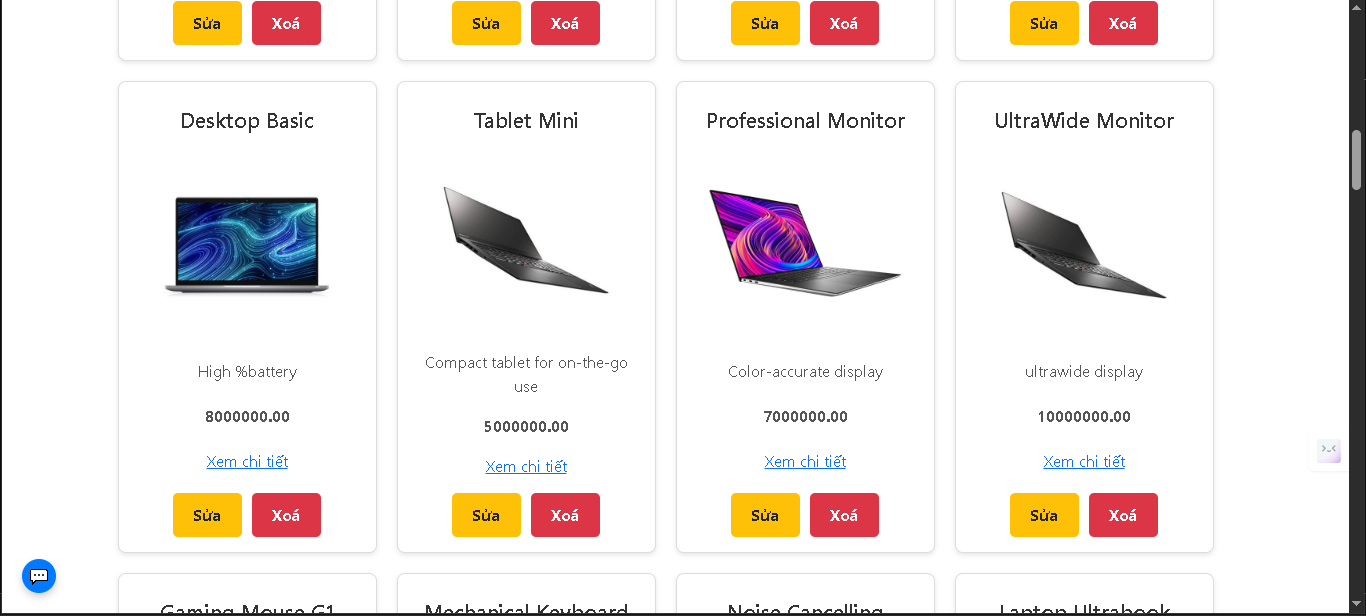
AI-generated content may be incorrect.

#### **1.4.2.2. Giao diện cho Admin**

##### **1.4.2.2.1. Màn hình chính**

* **Mục đích:** Hiển thị tất cả các chức năng liên quan giúp admin quản lý cửa hàng.



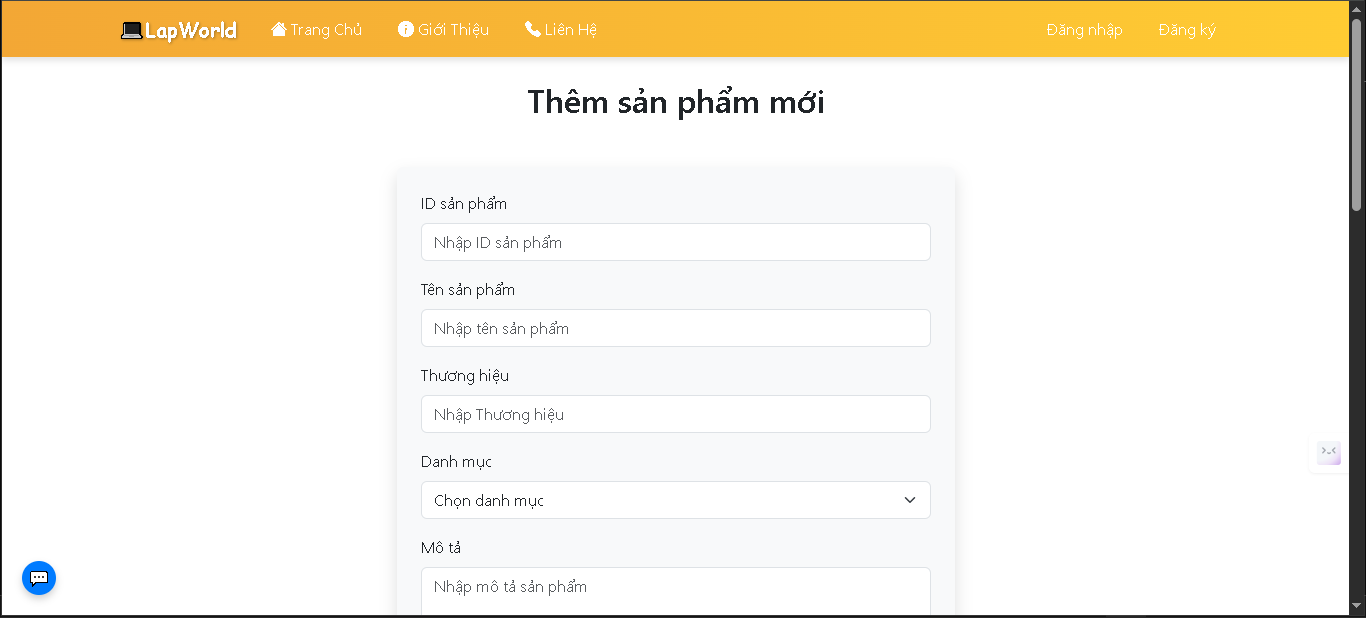


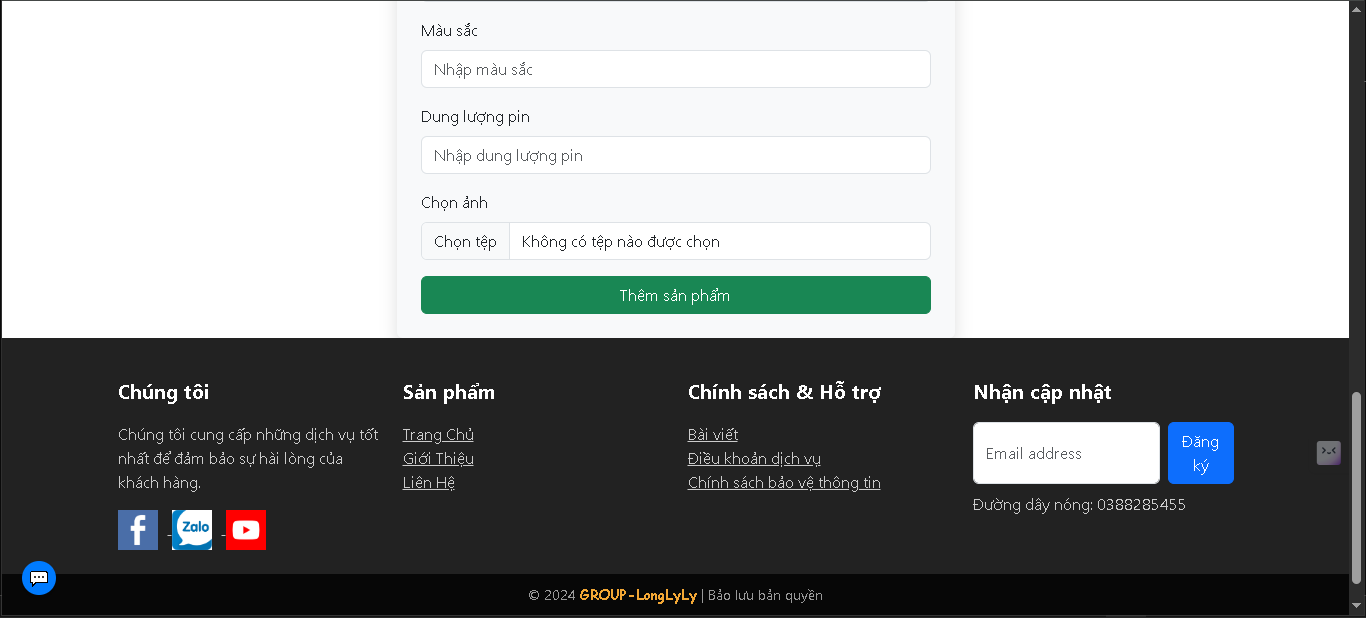
A screenshot of a computer

AI-generated content may be incorrect.

##### **1.4.2.2.2. Màn hình thêm sản phẩm**

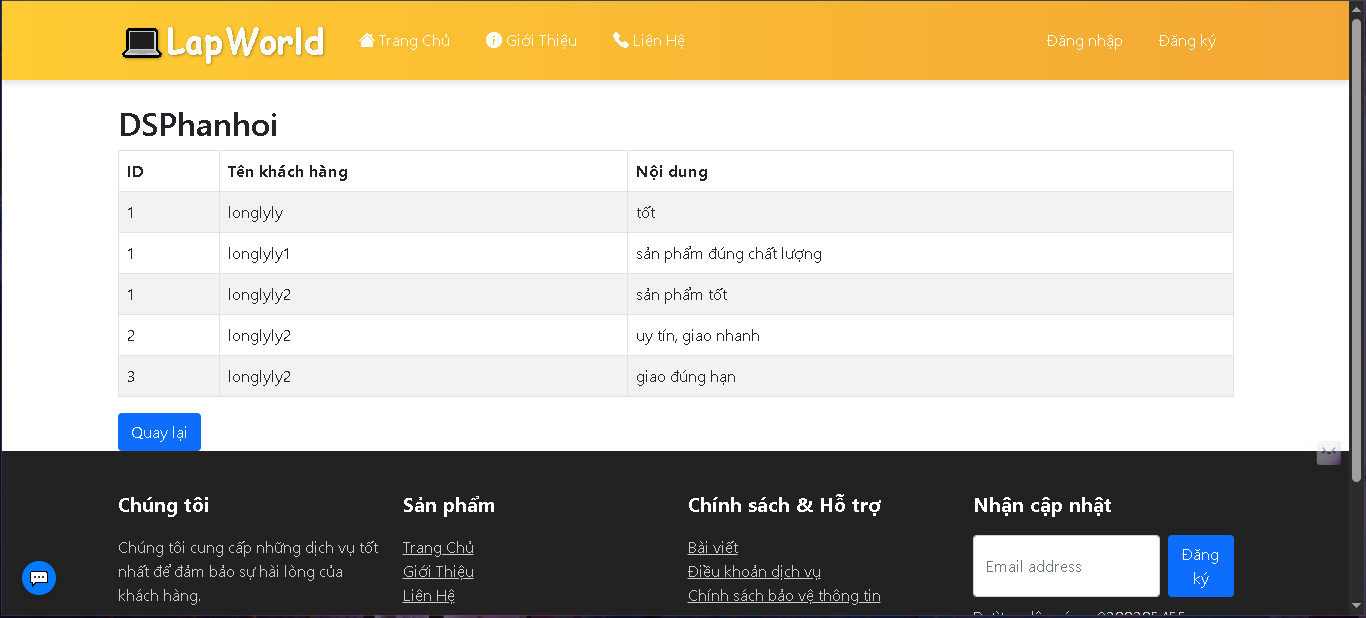
* **Mục đích:** Giúp người quản trị có thể thêm sản phẩm vào website.
* Hiển thị các text box tương ứng các thông tin của sản phẩm và nút thêm để thêm sản phẩm vào web





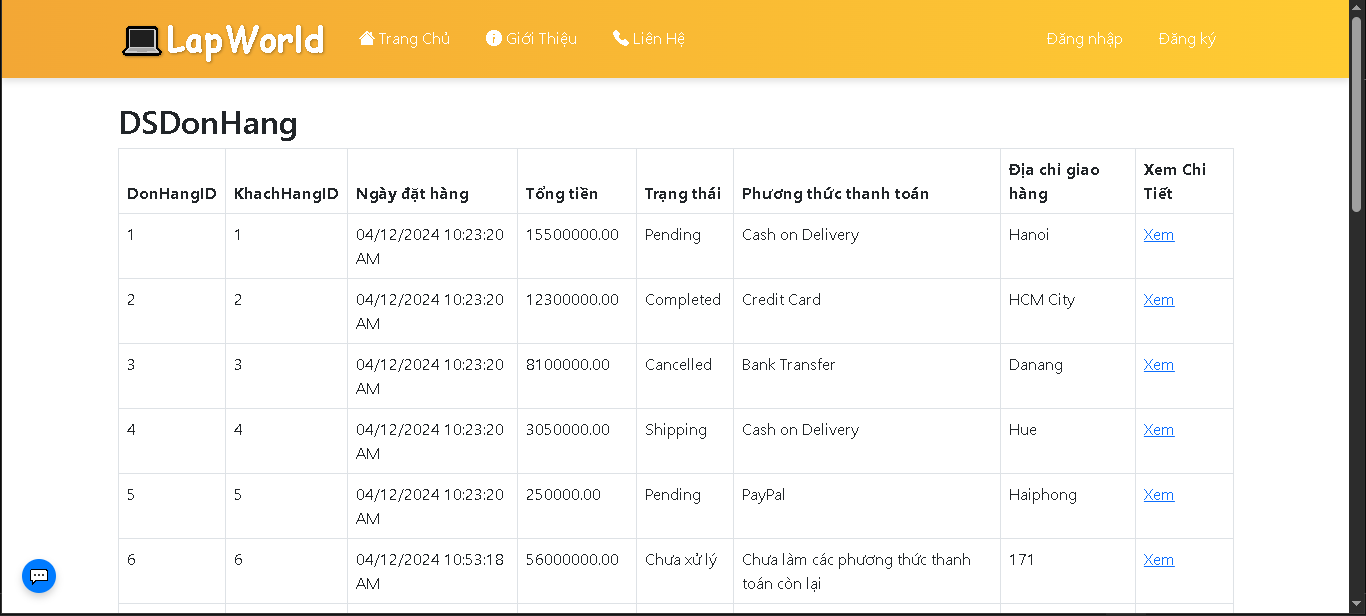
##### **1.4.2.2.3. Màn hình xem các phản hồi từ khách hàng**

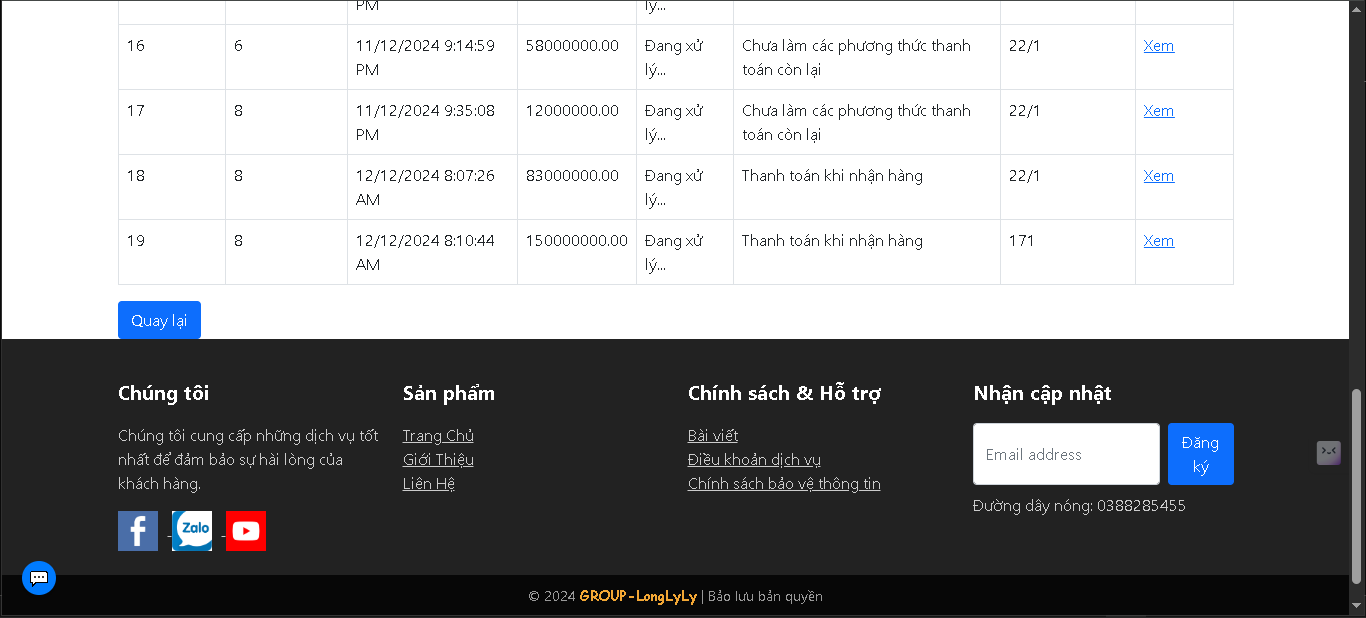
* **Mục đích:** Giúp người quản trị theo dõi các phản hồi từ người dùng.
* Hiển thị danh mục của bảng DSPhanhoi.



##### **1.4.2.2.4. Màn hình xem danh sách các đơn hàng**

* **Mục đích:** Giúp người quản trị theo dõi các phản hồi từ người dùng.
* Hiển thị danh mục các đơn hàng từ người dùng đã mua và các thông tin liên quan đơn hàng đó.





## 1.5. Các sơ đồ liên quan

Các sơ đồ mô tả chức năng và cấu trúc của website.

### 1.5.1. Xây dựng biểu đồ thực thể liên kết ERD

A screenshot of a computer

AI-generated content may be incorrect.

### 1.5.2. Biểu đồ trình tự (Sequence diagram)

#### **1.5.2.1. Đăng ký**

A screenshot of a computer program

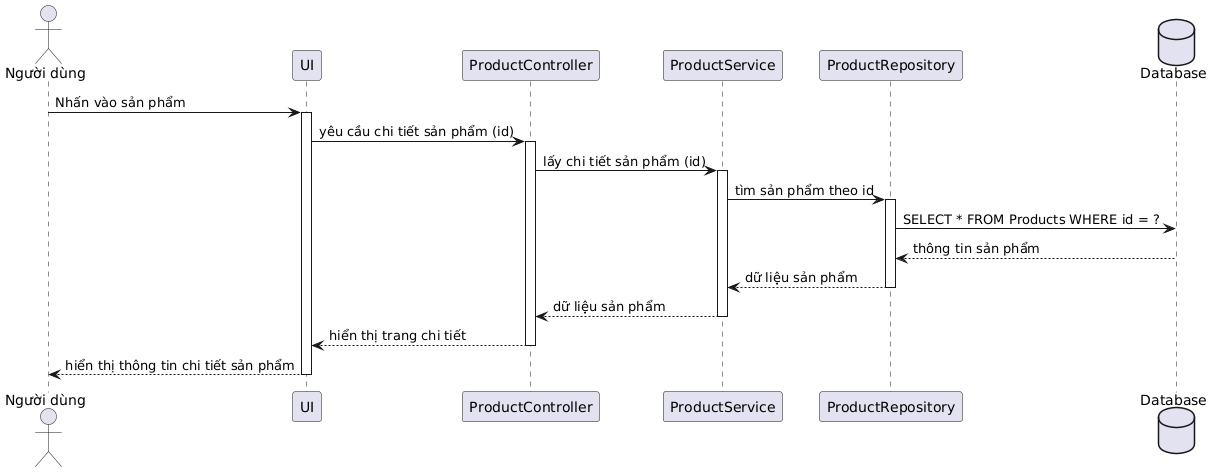
AI-generated content may be incorrect.

#### **1.5.2.2. Đăng nhập**

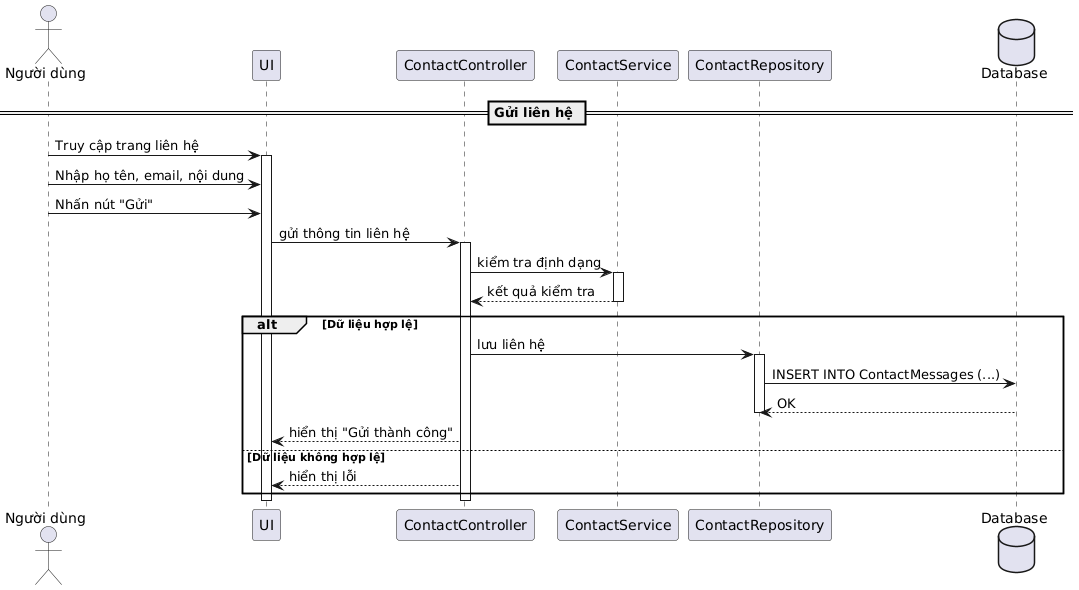
A diagram of a software project

AI-generated content may be incorrect.

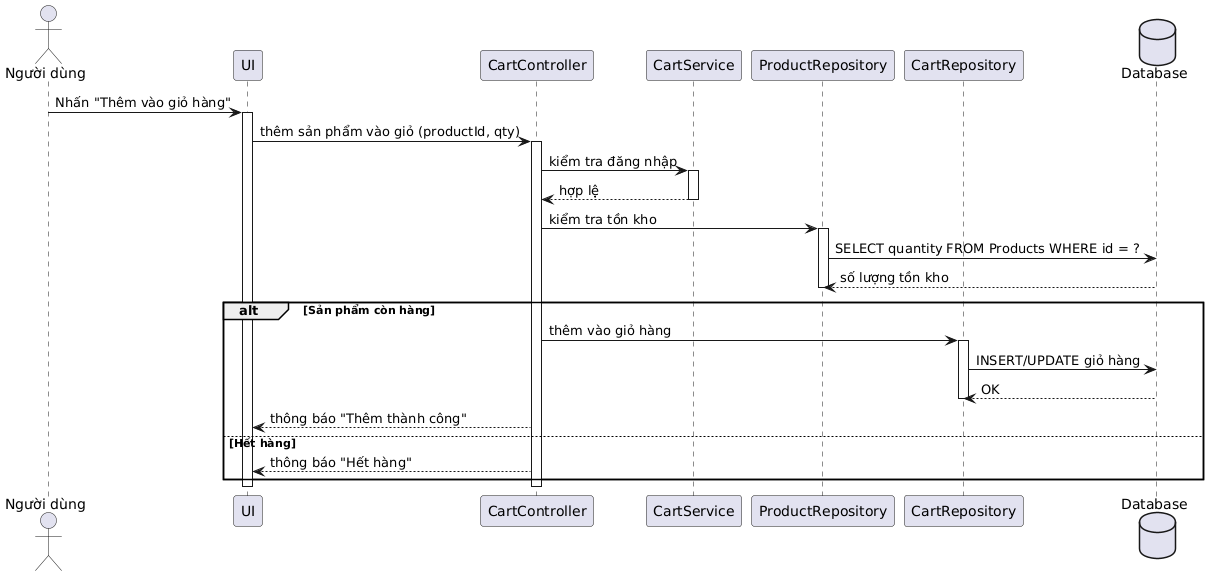
#### **1.5.2.3. Xem chi tiết**



#### **1.5.2.4. Liên hệ admin**



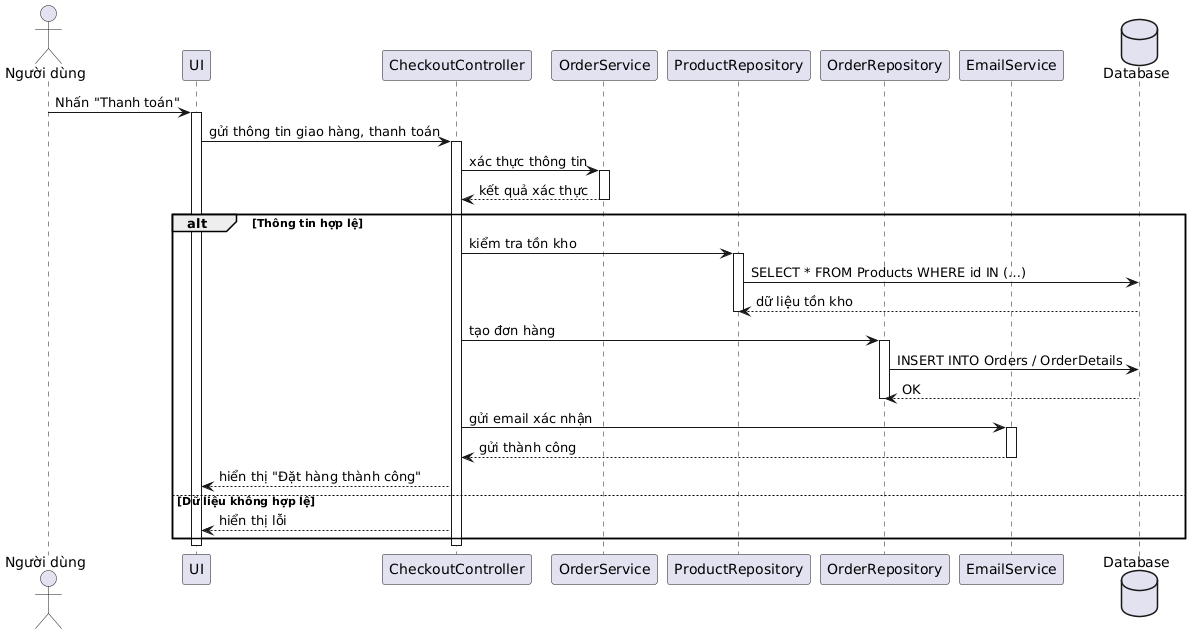
#### **1.5.2.5. Thêm giỏ hàng**



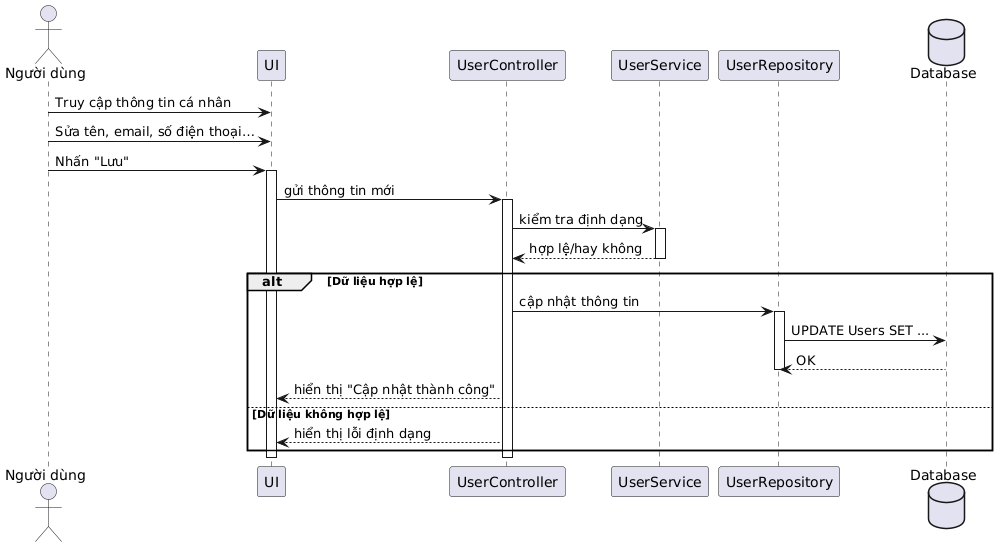
#### **1.5.2.6. Cập nhật giỏ hàng**



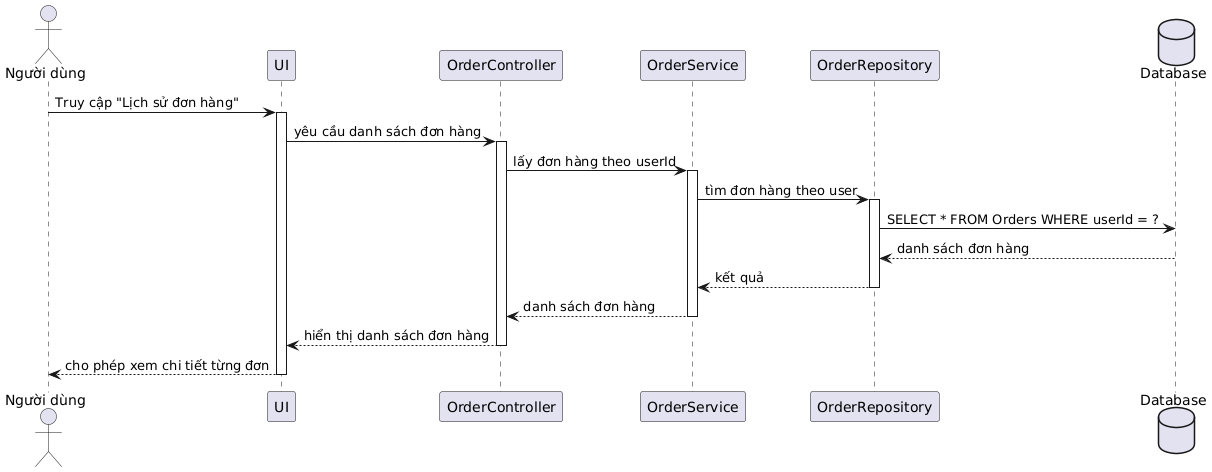
#### **1.5.2.7. Thanh toán**



#### **1.5.2.8. Cập nhật thông tin cá nhân**



#### **1.5.2.9. Xem lịch sử đơn hàng**



### 1.5.3. Sơ đồ hoạt động (Activity Diagram)

#### **1.5.3.1. Chức năng đăng ký**

A diagram of a flowchart

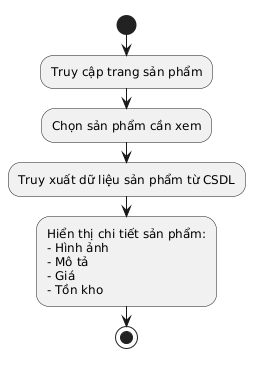
AI-generated content may be incorrect.

#### **1.5.3.2. Chức năng đăng nhập**

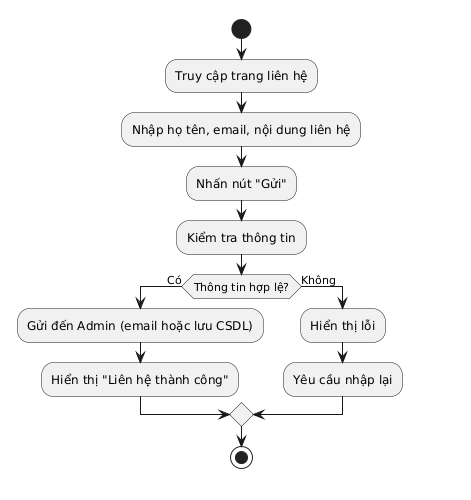
A diagram of a flowchart

AI-generated content may be incorrect.

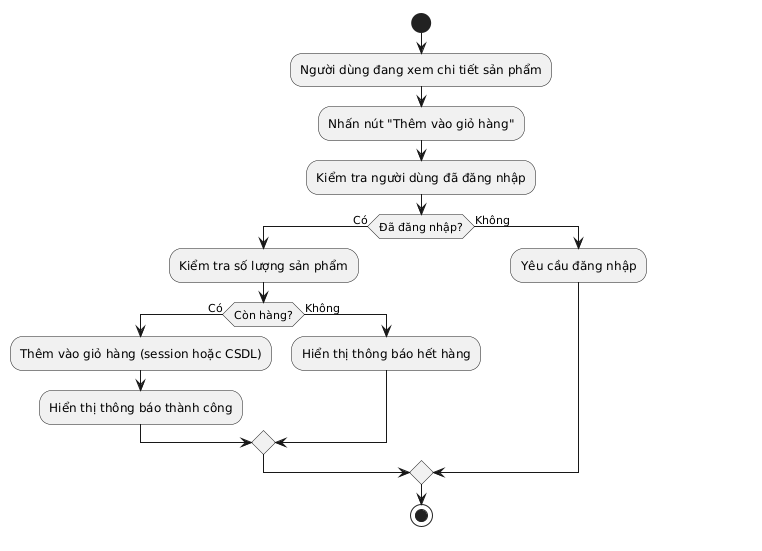
#### **1.5.3.3. Chức năng xem chi tiết sản phẩm**

****

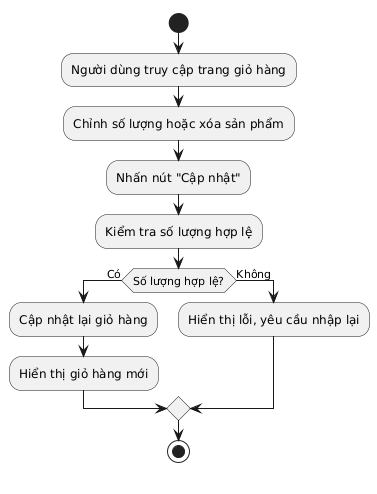
#### **1.5.3.4. Chức năng liên hệ với Admin**

****

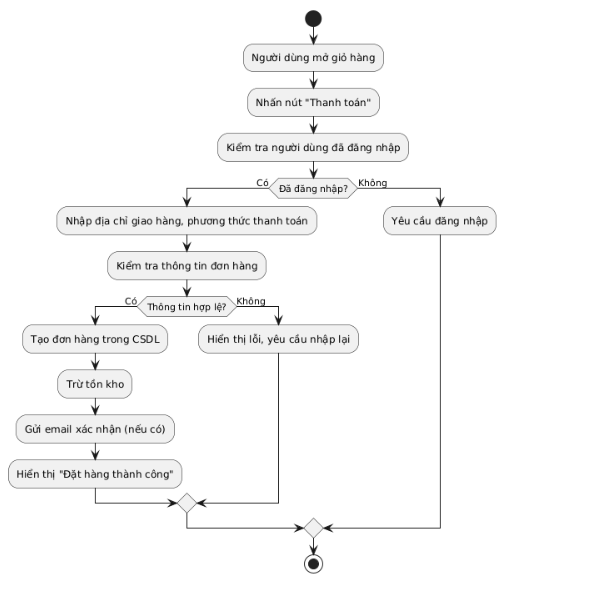
#### **1.5.3.5. Chức năng thêm sản phẩm vào giỏ hàng**

****

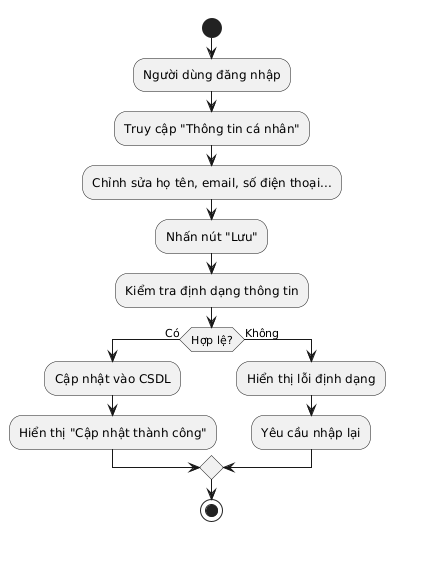
#### **1.5.3.6. Chức năng cập nhật giỏ hàng**

****

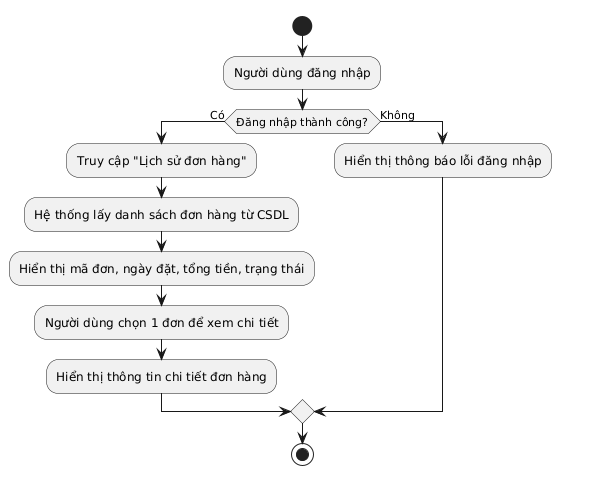
#### **1.5.3.7. Chức năng thanh toán**

****

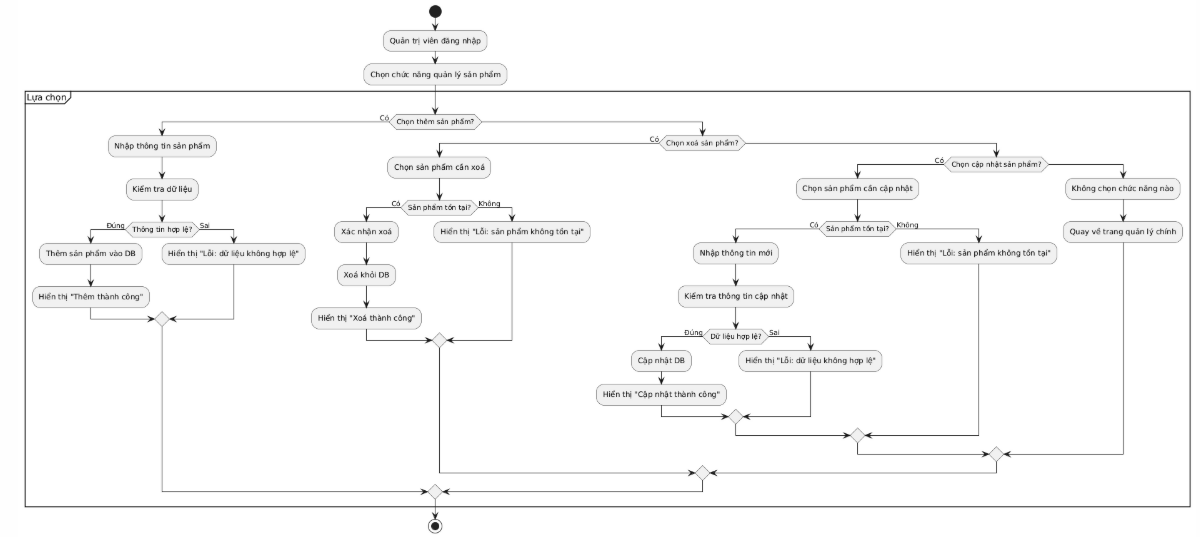
#### **1.5.3.8. Chức năng cập nhật thông tin cá nhân**

****

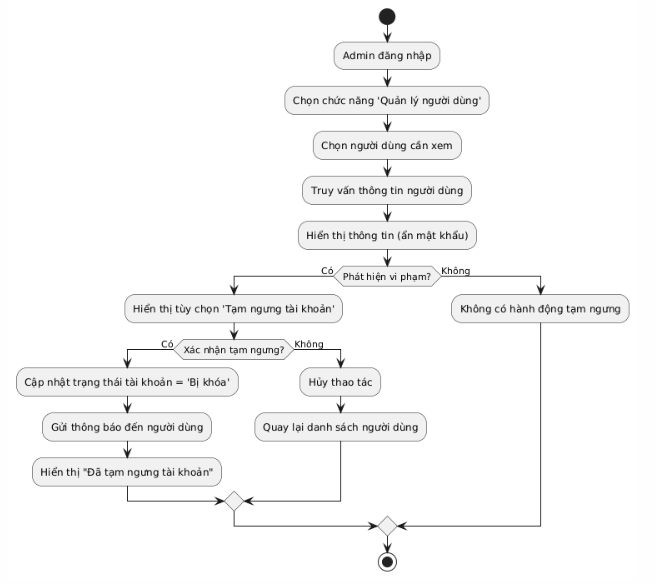
#### **1.5.3.9. Chức năng xem lịch sử đơn hàng của khách hàng**

****

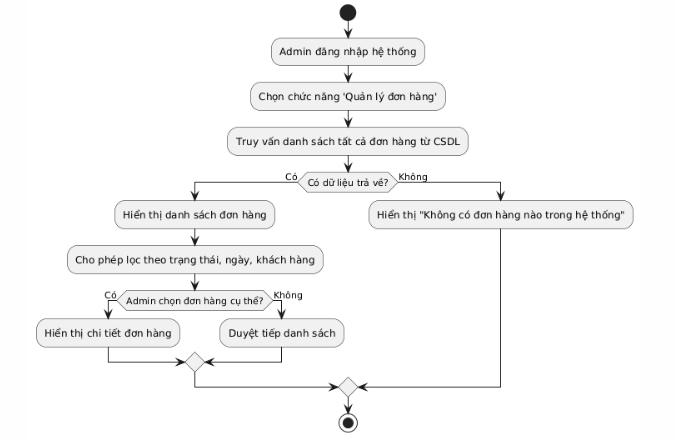
#### **1.5.3.10. Thêm, xoá, cập nhật các thông tin của sản phẩm**

****

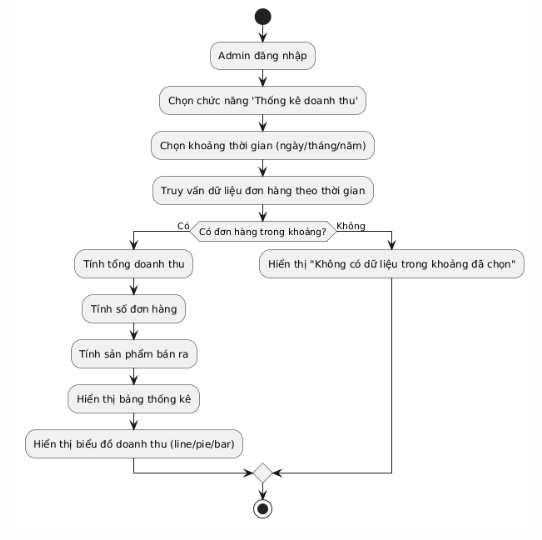
#### **1.5.3.11. Xem chi tiết (ẩn mật khẩu người dùng), tạm ngưng tài khoản khi có vi phạm**



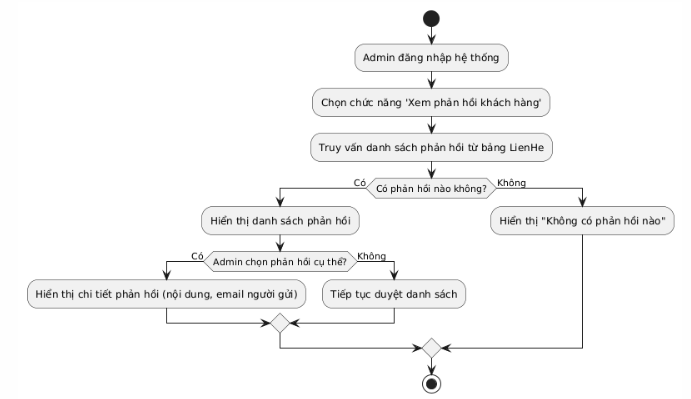
#### **1.5.3.12. Kiểm tra danh sách đơn hàng của toàn bộ cửa hàng**



#### **1.5.3.13. Biểu đồ thống kê doanh thu**

****

#### **1.5.3.14. Kiểm tra danh sách các phản hồi**



# CHƯƠNG 2. TỔNG KẾT VÀ HƯỚNG PHÁT TRIỂN

## 2.1. Tổng kết nội dung tài liệu

Tài liệu SDS đã triển khai giải pháp thiết kế kỹ thuật, bao gồm:

* Kiến trúc hệ thống phân tầng (Presentation, Business Logic, Data Access, Integration), mô hình MVC với ASP.NET MVC và Entity Framework.
* Thiết kế cơ sở dữ liệu quan hệ với các bảng chính như KhachHang, SanPham, DonHang, ChiTietDonHang, PhanHoi, v.v.
* Cấu trúc chương trình: các lớp Model, các Controller xử lý nghiệp vụ và các View Razor cho giao diện người dùng và quản trị.
* Các sơ đồ UML (ERD, Sequence, Use Case, Activity) và wireframe làm rõ luồng xử lý và tương tác giữa thành phần.
* Quy tắc và mẫu thiết kế giao diện responsive theo chuẩn Bootstrap, đảm bảo trải nghiệm nhất quán trên nhiều thiết bị.

## 2.2. Đánh giá chất lượng tài liệu

* **Tính đầy đủ**: Toàn bộ yêu cầu chức năng và phi chức năng đã được xác định và mô tả rõ ràng, không bỏ sót các trường hợp nghiệp vụ quan trọng.
* **Tính nhất quán**: Các định nghĩa thuật ngữ, mã hiệu yêu cầu (FR-xxx, NFR-xxx) và các sơ đồ đều đồng bộ, dễ tham chiếu và không mâu thuẫn.
* **Tính khả thi**: Giải pháp thiết kế trong SDS có cơ sở kỹ thuật rõ ràng, sử dụng công nghệ phổ biến (ASP.NET MVC, EF), phù hợp với hạ tầng đề xuất.
* **Tính dễ bảo trì**: Cấu trúc phân tầng, quy ước đặt tên và diagram chi tiết giúp nhóm phát triển và kiểm thử dễ dàng theo dõi, kiểm soát và mở rộng.

## 2.3. Hướng phát triển và triển khai

* **Triển khai thực tế**: Bước tiếp theo là chuyển đổi SDS thành mã nguồn, thực hiện cài đặt môi trường (IIS, SQL Server), cấu hình CI/CD và triển khai trên staging.
* **Kiểm thử toàn diện**: Thiết kế test case dựa trên SRS và sơ đồ luồng nghiệp vụ, triển khai kiểm thử đơn vị, tích hợp, hệ thống và kiểm thử bảo mật.
* **Tối ưu hiệu năng**: Áp dụng caching (Redis/MemoryCache), tối ưu truy vấn SQL, phân trang dữ liệu lớn và đo đạc thời gian phản hồi.
* **Bảo mật nâng cao**: Mở rộng xác thực hai lớp (2FA), giám sát truy cập, phòng chống XSS, CSRF, SQL Injection và mã hóa dữ liệu nhạy cảm.
* **Mở rộng tính năng**: Tích hợp thêm cổng thanh toán đa dạng, mobile app, kết nối ERP/CRM, phân tích dữ liệu người dùng bằng ML, chatbot nâng cao.

## 2.4. Kết luận

Qua quá trình thiết kế giải pháp (SDS), tài liệu đã xây dựng nền tảng vững chắc để nhóm phát triển tiến hành triển khai, kiểm thử và bàn giao hệ thống website bán máy tính và thiết bị điện tử. Việc tuân thủ chặt chẽ các chuẩn mực và quy ước trong hai tài liệu này sẽ đảm bảo dự án hoàn thành đúng tiến độ, chất lượng và đáp ứng đầy đủ kỳ vọng của khách hàng cũng như người quản trị hệ thống.

# TÀI LIỆU THAM KHẢO

**[1].** <https://qndev.github.io/resources/SRS.pdf>