A blue and white rectangular frame

Description automatically generated

**TRƯỜNG ĐẠI HỌC BÌNH DƯƠNG**

**KHOA CÔNG NGHỆ THÔNG TIN, ROBOT**

**VÀ TRÍ TUỆ NHÂN TẠO**

**---o0o---**

**TIỂU LUẬN MÔN HỌC**

**CHUYÊN ĐỀ 1**

**Đề tài:**

**XÂY DỰNG WEBSITE   
SHOP BÁN QUẦN ÁO THỜI TRANG**

**Người hướng dẫn: Ths. Nguyễn Thanh Sơn**

**Sinh viên thực hiện : Vũ Duy Hoàng 22050038**

**Vương Quốc Cường 22050059**

**Nguyễn Hoàng Vũ 22050072**

***Bình Dương, tháng 5 năm 2025***

**LỜI CẢM ƠN**

Trước khi kết thúc báo cáo này, nhóm em xin được dành những dòng chân thành nhất để bày tỏ lòng biết ơn sâu sắc đến thầy Lê Hữu Sơn – người đã trực tiếp hướng dẫn, chỉ dạy và đồng hành cùng nhóm em trong suốt quá trình học tập và thực hiện môn Chuyên đề.

Trong suốt thời gian tham gia học phần, nhóm em đã có cơ hội tiếp cận và học hỏi nhiều kiến thức mới mẻ, thiết thực và đầy bổ ích từ thầy. Từ những kiến thức nền tảng về thiết kế website, quản lý cơ sở dữ liệu (database), cho đến những kỹ năng quan trọng như triển khai và đưa website lên nền tảng cloud, tất cả đều được thầy truyền đạt một cách dễ hiểu, logic và gần gũi. Qua từng buổi học, nhóm em không chỉ tiếp thu kiến thức mà còn học được cách tư duy, phân tích và giải quyết vấn đề một cách khoa học, linh hoạt – những điều mà sách vở khó có thể mang lại nếu không có người thầy tận tâm như thầy Sơn.

Không chỉ dừng lại ở việc giảng dạy, thầy còn luôn sẵn sàng hỗ trợ, giải đáp mọi thắc mắc, truyền cảm hứng học tập và định hướng cho chúng nhóm em trong từng bước thực hành. Những góp ý chân thành và sự tận tụy của thầy đã giúp nhóm em hoàn thiện hơn từng phần trong quá trình học, và đặc biệt là trong việc xây dựng, triển khai một dự án web hoàn chỉnh – điều mà trước đây nhóm em còn cảm thấy rất mơ hồ.

Chính nhờ sự dìu dắt, định hướng đúng đắn và kiến thức thực tiễn quý báu mà thầy truyền đạt, nhóm em đã tích lũy được nhiều kỹ năng quan trọng cho hành trang nghề nghiệp trong tương lai. Nhóm em tin rằng những kiến thức và trải nghiệm có được từ môn học này sẽ là nền tảng vững chắc giúp nhóm em tự tin hơn khi bước vào môi trường làm việc thực tế sau này.

Một lần nữa, nhóm em xin gửi đến thầy Lê Hữu Sơn lời cảm ơn chân thành và sâu sắc nhất. Nhóm em kính chúc thầy luôn dồi dào sức khỏe, nhiều niềm vui trong cuộc sống và ngày càng thành công hơn nữa trong sự nghiệp giảng dạy và nghiên cứu. Mong rằng sẽ còn nhiều thế hệ sinh viên khác tiếp tục được thầy truyền cảm hứng và dẫn dắt, như nhóm em đã từng may mắn có được trong học kỳ này.

Nhóm em xin chân thành cảm ơn.

**NHẬN XÉT VÀ CHẤM ĐIỂM CỦA GIẢNG VIÊN**

1. **Nhận xét**:

***Những kết quả đạt được:***

***Những hạn chế:***

1. **Điểm đánh giá**

*Bình Dương, ngày … tháng … năm 2025*

**Giảng viên chấm thi**

*(Ký và ghi rõ họ tên)*

…………………..

**MỤC LỤC**

[Chương 1. GIỚI THIỆU TỔNG QUAN 1](#_Toc198074539)

[1.2. Mô tả bài toán 1](#_Toc198074540)

[1.3. Mô tả chức năng và phi chức năng 3](#_Toc198074541)

[1.3.1. Chức năng chính (Functional Requirements) 3](#_Toc198074542)

[1.3.2. Phi chức năng. 4](#_Toc198074543)

[1.4. Mô tả các công nghệ 4](#_Toc198074544)

[1.4.1. Giao diện người dùng (Frontend) 4](#_Toc198074545)

[1.4.2. Xử lý phía máy chủ (Backend) 4](#_Toc198074546)

[1.4.3. Cơ sở dữ liệu 4](#_Toc198074547)

[1.4.4. Triển khai và công cụ hỗ trợ 4](#_Toc198074548)

[Chương 2. PHÂN TÍCH 5](#_Toc198074549)

[2.2. Sơ đồ use case tổng quát 5](#_Toc198074550)

[2.2.1. Phân tích sơ đồ Use Case hệ thống quản lý bán hàng 6](#_Toc198074551)

[2.2.1.1. Danh sách actor 6](#_Toc198074552)

[2.2.1.2. Danh sách user 6](#_Toc198074553)

[2.3. Sơ đồ use case chi tiết 8](#_Toc198074554)

[2.3.1. Quản trị viên 8](#_Toc198074555)

[2.3.1.1. Quản lí tài khoản (Admin) 8](#_Toc198074556)

[2.3.1.2. Quản lí sản phẩm (Nhân viên) 9](#_Toc198074557)

[2.3.1.3. Quản lí loại sản phẩm (Nhân viên) 11](#_Toc198074558)

[2.3.1.4. Quản lí đơn đặt hàng (Nhân viên) 12](#_Toc198074559)

[2.3.2. Khách hàng 12](#_Toc198074560)

[2.3.2.1. Xem sản phẩm (Người dùng) 12](#_Toc198074561)

[2.3.2.2. Tìm kiếm thông tin (Người dùng) 13](#_Toc198074562)

[2.3.2.3. Đăng ký (Người dùng) 14](#_Toc198074563)

[2.3.2.4. Đăng nhập (Thành viên) 15](#_Toc198074564)

[2.3.2.5. Đăng xuất (Thành viên) 16](#_Toc198074565)

[2.3.2.6. Lập đơn đặt hàng (Thành viên) 17](#_Toc198074566)

[2.4. Sơ đồ phân tích cơ sở dữ liệu 18](#_Toc198074567)

[2.4.1. Phân tích mối quan hệ 18](#_Toc198074568)

[2.4.1.1. Mối quan hệ giữa người dùng và tài khoản. 18](#_Toc198074569)

[2.4.1.2. Mối quan hệ giữa khách hàng và đơn hàng 19](#_Toc198074570)

[2.4.1.3. Mối quan hệ giữa đơn hàng và sản phẩm 20](#_Toc198074571)

[2.4.1.4. Mối quan hệ giữa sản phẩm và loại sản phẩm 20](#_Toc198074572)

[Chương 3. THIẾT KẾ 21](#_Toc198074573)

[3.2. Sơ đồ tuần tự cho các nghiệp vụ thường xuyên 21](#_Toc198074574)

[3.2.1. Usecase “QLSanPham” 21](#_Toc198074575)

[3.2.1.1. Usecase “ThemSanPham” 21](#_Toc198074576)

[3.2.1.2. Usecase “SuaSanPham” 22](#_Toc198074577)

[3.2.1.3. Usecase “XoaSanPham” 22](#_Toc198074578)

[3.2.2. Usecase “TimKiemSanPham” 23](#_Toc198074579)

[3.2.3. Usecase “QLLoaiSanPham” 23](#_Toc198074580)

[3.2.3.1. Usecase “ThemLoaiSanPham” 23](#_Toc198074581)

[3.2.3.2. Usecase “SuaLoaiSanPham” 24](#_Toc198074582)

[3.2.3.3. Usecase “XoaLoaiSanPham” 24](#_Toc198074583)

[3.2.4. Usecase “DangNhap” 25](#_Toc198074584)

[3.2.5. Usecase “LapDonDatHang” 25](#_Toc198074585)

[3.2.6. Usecase “QLGioHang” 26](#_Toc198074586)

[3.2.6.1. Usecase “ThemSanPhamVaoGioHang” 26](#_Toc198074587)

[3.2.6.2. Usecase “SuaSanPhamTrongGioHang” 27](#_Toc198074588)

[3.2.6.3. Usecase “XoaSanPhamTrongGioHang” 27](#_Toc198074589)

[3.2.7. Usecase “DangKyTaiKhoan” 28](#_Toc198074590)

[3.3. Lược đồ quan hệ 28](#_Toc198074591)

[3.3.1. Lược đồ quan hệ. 28](#_Toc198074592)

[3.3.2. Mô tả chi tiết lược đồ cơ sở dữ liệu quan hệ. 29](#_Toc198074593)

[Chương 4. TRIỂN KHAI HỆ THỐNG TRÊN HẠ TẦNG CLOUD 31](#_Toc198074594)

[4.1. Giới thiệu chung 31](#_Toc198074595)

[4.2. Mô hình triển khai trên cloud. 32](#_Toc198074596)

[4.3. Mô tả mô hình 32](#_Toc198074597)

[4.3.1. Các dịch vụ sử dụng 32](#_Toc198074598)

[4.3.2. Mô tả chi tiết mô hình. 33](#_Toc198074599)

[4.4. Triển khai website trên cloud . 33](#_Toc198074600)

[4.4.1. Khởi tạo dịch vụ VPC 34](#_Toc198074601)

[4.4.2. Khởi tạo dịch vụ RDS. 37](#_Toc198074602)

[4.4.3. Khởi tạo dịch vụ EC2 38](#_Toc198074603)

[4.4.4. Đẩy Frontend và Backend lên EC2. 38](#_Toc198074604)

[Chương 5. KẾT LUẬN 39](#_Toc198074605)

[5.1. Kết quả triển khai 39](#_Toc198074606)

[5.2. Ưu điểm 40](#_Toc198074607)

[5.3. Hạn chế 40](#_Toc198074608)

**MỤC LỤC HÌNH ẢNH**

[Hình 2‑1 Sơ đồ use case tổng quát 5](#_Toc198075477)

[Hình 2‑2 Sơ đồ use case Quản Lý Tài Khoản 8](#_Toc198075478)

[Hình 2‑3 Sơ đồ use case Quản Lí Sản Phẩm 9](#_Toc198075479)

[Hình 2‑4 Sơ đồ use case Quản Lí Loại Sản Phẩm 11](#_Toc198075480)

[Hình 2‑5 Sơ đồ use case Quản Lí Đơn Đặt Hàng 12](#_Toc198075481)

[Hình 2‑6 Sơ đồ use case Xem Sản Phẩm 12](#_Toc198075482)

[Hình 2‑7 Sơ đồ use case Tìm Kiếm Thông Tin (người dùng) 13](#_Toc198075483)

[Hình 2‑8 Sơ đồ use case Đăng Ký (người dùng) 14](#_Toc198075484)

[Hình 2‑9 Sơ đồ use case Đăng Nhập (thành viên) 15](#_Toc198075485)

[Hình 2‑10 Sơ đồ use case Đăng Xuất (thành viên) 16](#_Toc198075486)

[Hình 2‑11 Sơ đồ Lập Đơn Đặt Hàng (thành viên) 17](#_Toc198075487)

[Hình 2‑12 Sơ đồ cơ sở dữ liệu ERD 18](#_Toc198075488)

[Hình 2‑13 Mối quan hệ giữa người dùng và tài khoản 18](#_Toc198075489)

[Hình 2‑14 Mối quan hệ giữa khách hàng và đơn hàng 19](#_Toc198075490)

[Hình 2‑15 Mối quan hệ giữa đơn hàng và sản phẩm 20](#_Toc198075491)

[Hình 2‑16 Mối quan hệ giữa sản phẩm và loại sản phẩm 20](#_Toc198075492)

[Hình 3‑1 Sequence thêm sản phẩm thành công 21](#_Toc198075493)

[Hình 3‑2 Sequence sửa sản phẩm thành công 22](#_Toc198075494)

[Hình 3‑3 Sequence xóa sản phẩm thành công. 22](#_Toc198075495)

[Hình 3‑4 Sequence tìm kiếm sản phẩm thành công. 23](#_Toc198075496)

[Hình 3‑5 Sequence thêm loại sản phẩm thành công 23](#_Toc198075497)

[Hình 3‑6 Sequence sửa loại sản phẩm thành công 24](#_Toc198075498)

[Hình 3‑7 Sequence xóa loại sản phẩm thành công. 24](#_Toc198075499)

[Hình 3‑8 Sequence đăng nhập thành công. 25](#_Toc198075500)

[Hình 3‑9 Sequence lập đơn đặt hàng thành công 25](#_Toc198075501)

[Hình 3‑10 Sequence thêm sản phẩm vào giỏ hàng thành công 26](#_Toc198075502)

[Hình 3‑11 Sequence sửa sản phẩm trong giỏ hàng 27](#_Toc198075503)

[Hình 3‑12 Sequence xóa sản phẩm trong giỏ hàng thành công. 27](#_Toc198075504)

[Hình 3‑13 Sequence đăng kí tài khoản thành công 28](#_Toc198075505)

[Hình 3‑14 Lược đồ quan hệ 28](#_Toc198075506)

[Hình 4‑1 Sơ đồ triển khai trên cloud 32](#_Toc198075507)

[Hình 4‑2 Cấu trúc VPC dashboard để đẩy web lên cloud AWS 34](#_Toc198075508)

[Hình 4‑3 Cấu hình VPC 34](#_Toc198075509)

[Hình 4‑4 Cấu hình Subnets 35](#_Toc198075510)

[Hình 4‑5 Cấu hình Internet Gateway (IGW) 36](#_Toc198075511)

[Hình 4‑6 Cấu hình Route Table 36](#_Toc198075512)

[Hình 4‑7 Cấu hình DB Subnet Group 37](#_Toc198075513)

[Hình 4‑8 Cấu hình Database 37](#_Toc198075514)

[Hình 4‑9 Cấu hình Instance 38](#_Toc198075515)

[Hình 4‑10 Khởi chạy Backend 39](#_Toc198075516)

**MỤC LỤC BẢNG**

[Bảng 1‑1 Bảng chức năng đối với khách hàng 3](#_Toc198074609)

[Bảng 1‑2 Chức năng đối với quản trị viên 4](#_Toc198074610)

[Bảng 1‑3 Phi chức năng 4](#_Toc198074611)

[Bảng 2‑1 Bảng Danh Sách Actor 6](#_Toc198074612)

[Bảng 2‑2 Bảng danh sách user 6](#_Toc198074613)

[Bảng 3‑1 Tài khoản 29](#_Toc198074614)

[Bảng 3‑2 Nhân viên 29](#_Toc198074615)

[Bảng 3‑3 Khách hàng 29](#_Toc198074616)

[Bảng 3‑4 Đơn hàng 30](#_Toc198074617)

[Bảng 3‑5 Sản phẩm 30](#_Toc198074618)

[Bảng 3‑6 Chi tiết đơn hàng 30](#_Toc198074619)

[Bảng 3‑7 Loại sản phẩm 31](#_Toc198074620)

[Bảng 4‑1 Vai trò từng subnet 35](#_Toc198074621)

# GIỚI THIỆU TỔNG QUAN

## Mô tả bài toán

* Trong bối cảnh nền kinh tế số phát triển mạnh mẽ và cuộc cách mạng công nghiệp 4.0 đang diễn ra trên toàn cầu, ngành bán lẻ nói chung và ngành thời trang nói riêng đang đối mặt với những thách thức lớn. Đặc biệt đối với các cửa hàng thời trang vừa và nhỏ, việc quản lý bán hàng và kho hàng vẫn chủ yếu dựa vào phương pháp thủ công hoặc sử dụng các công cụ không đồng bộ, khiến quá trình này trở nên kém hiệu quả. Các cửa hàng thường sử dụng Excel hoặc phần mềm không chuyên biệt để theo dõi thông tin đơn hàng, sản phẩm và khách hàng. Điều này không chỉ tạo ra sự thiếu chính xác trong việc quản lý thông tin mà còn làm giảm hiệu quả công việc, gây khó khăn trong việc mở rộng quy mô.
* Vấn đề thường gặp: Các cửa hàng thời trang vừa và nhỏ gặp phải nhiều vấn đề trong việc quản lý bán hàng, bao gồm:
* Sai sót trong nhập liệu: Hầu hết các cửa hàng vẫn phải nhập liệu thủ công vào bảng tính Excel hoặc ghi chép tay, điều này dễ dẫn đến sai sót trong thông tin sản phẩm, đơn hàng và khách hàng. Những sai sót này làm tăng khối lượng công việc và gây khó khăn khi cần truy xuất thông tin nhanh chóng, đặc biệt trong các tình huống khẩn cấp hoặc khi có khiếu nại từ khách hàng.
* Khó khăn trong việc kiểm soát tồn kho: Việc không có hệ thống quản lý tồn kho tự động khiến các cửa hàng dễ bị thiếu hụt sản phẩm trong khi nhu cầu tăng cao, hoặc dư thừa hàng hóa khi không có cách theo dõi chính xác tình trạng kho. Điều này ảnh hưởng trực tiếp đến doanh thu và sự hài lòng của khách hàng.
* Quản lý đơn hàng không hiệu quả: Quy trình tiếp nhận và xử lý đơn hàng vẫn diễn ra qua các kênh thủ công như tin nhắn, điện thoại hoặc email. Điều này dẫn đến việc xử lý đơn hàng chậm chạp và dễ sai sót, đặc biệt trong những đợt khuyến mãi hoặc mùa cao điểm khi số lượng đơn hàng tăng mạnh.
* Thiếu trải nghiệm khách hàng chuyên nghiệp: Khách hàng không thể dễ dàng tìm kiếm và đặt hàng sản phẩm qua các nền tảng trực tuyến, thiếu các thông tin chi tiết về sản phẩm, thiếu hệ thống thanh toán trực tuyến và theo dõi đơn hàng. Điều này làm giảm sự hài lòng và lòng tin của khách hàng, ảnh hưởng đến khả năng cạnh tranh của cửa hàng.
* Giải pháp cần thiết: Để giải quyết các vấn đề trên, nhóm em đã xây dựng một hệ thống bán hàng trực tuyến cho các cửa hàng thời trang vừa và nhỏ, với mục tiêu tự động hóa quy trình bán hàng, quản lý kho, đơn hàng và khách hàng, từ đó nâng cao hiệu quả công việc và trải nghiệm khách hàng. Hệ thống này sẽ bao gồm các tính năng chính như:
* Quản lý sản phẩm, đơn hàng và khách hàng đồng bộ: Tất cả các thông tin liên quan đến sản phẩm, đơn hàng và khách hàng được đồng bộ và cập nhật theo thời gian thực. Việc này giúp giảm thiểu sai sót, tiết kiệm thời gian và nâng cao hiệu quả công việc.
  + Thông tin sản phẩm sẽ được quản lý chi tiết với mô tả, kích thước, màu sắc, giá cả, và hình ảnh, giúp khách hàng dễ dàng lựa chọn sản phẩm.
  + Hệ thống sẽ hỗ trợ chủ cửa hàng theo dõi trạng thái đơn hàng từ khi đặt hàng đến khi được xác nhận.
* Tự động hóa quy trình bán hàng: Hệ thống cho phép khách hàng dễ dàng thêm sản phẩm vào giỏ hàng .
* Quản lý kho tự động: Hệ thống giúp quản lý kho hàng hiệu quả, theo dõi số lượng sản phẩm còn trong kho và cảnh báo khi sản phẩm đạt mức tồn kho thấp
* Nâng cao trải nghiệm khách hàng: Website được thiết kế với giao diện thân thiện, dễ sử dụng, giúp khách hàng dễ dàng tìm kiếm, lọc và đặt hàng sản phẩm. Ngoài ra, hệ thống còn hỗ trợ khách hàng tạo tài khoản để theo dõi lịch sử đơn hang.
* Mục tiêu của bài toán: Mục tiêu của bài toán là xây dựng một hệ thống phần mềm bán hàng trực tuyến tối ưu cho các cửa hàng thời trang vừa và nhỏ, giúp:
* Tự động hóa toàn bộ quy trình bán hàng, từ xem sản phẩm đến khi hoàn tất đơn hàng.
* Quản lý kho, đơn hàng và khách hàng một cách hiệu quả.
* Tích hợp bán hàng đa kênh, mở rộng khả năng tiếp cận khách hàng.
* Nâng cao trải nghiệm khách hàng qua các tính năng như tìm kiếm nhanh, lọc sản phẩm và chương trình khuyến mãi.

## Mô tả chức năng và phi chức năng

* Trong quá trình phát triển hệ thống website bán hàng cho cửa hàng thời trang, việc xác định rõ các chức năng (functional requirements) và phi chức năng (non-functional requirements) là yếu tố then chốt để đảm bảo hệ thống đáp ứng đúng nhu cầu thực tế của cả người dùng cuối và quản trị viên. Mục này trình bày chi tiết các yêu cầu về chức năng chính mà hệ thống cần triển khai, cũng như các tiêu chí về hiệu năng, độ tin cậy, khả dụng và tính tương thích để đảm bảo hệ thống hoạt động ổn định, hiệu quả và thân thiện với người sử dụng.

### Chức năng chính (Functional Requirements)

* Hệ thống website thời trang sẽ phục vụ hai nhóm người dùng chính: Khách hàng (User) và Quản trị viên (Admin). Mỗi nhóm người dùng có tập hợp chức năng riêng biệt, tương ứng với vai trò của họ trong hệ thống.
* Đối với Khách hàng (User):

**Bảng 1‑1 Bảng chức năng đối với khách hàng**

|  |  |
| --- | --- |
| **Chức năng** | **Mô tả chi tiết** |
| Xem danh mục sản phẩm | Cho phép khách truy cập xem tất cả sản phẩm theo danh mục như: áo, quần, váy. Có thể xem theo nhóm phân loại như màu sắc, giá bán. |
| Tìm kiếm | Hệ thống hỗ trợ tìm kiếm theo từ khóa tên sản phẩm. |
| Quản lý giỏ hàng | Người dùng có thể thêm sản phẩm vào giỏ hàng, cập nhật số lượng, xóa hoặc chỉnh sửa trước khi tiến hành đặt hàng. Giỏ hàng được lưu trữ theo tài khoản đăng nhập. |
| Đặt hàng | Sau khi hoàn tất giỏ hàng, khách hàng có thể đặt hàng và chờ xác nhận đơn hàng. |
| Xem lịch sử đơn hàng và trạng thái | Sau khi đặt hàng, người dùng có thể đăng nhập để theo dõi tình trạng đơn hàng cũng như xem lại lịch sử mua sắm trước đó. |

* Đối với Quản trị viên (Admin):

**Bảng 1‑2 Chức năng đối với quản trị viên**

|  |  |
| --- | --- |
| **Chức năng** | **Mô tả chi tiết** |
| Quản lý người dùng và vai trò | Cho phép đăng nhập hệ thống quản trị. |
| Quản lý sản phẩm | Giao diện để thêm mới sản phẩm, chỉnh sửa thông tin sản phẩm (tên, mô tả, giá, số lượng tồn kho, size, màu sắc), cập nhật hình ảnh minh họa. |
| Quản lý danh mục và thuộc tính | Hệ thống cho phép tạo và sửa các danh mục (áo, quần, váy...), cũng như các thuộc tính cố định như kích thước, màu sắc để phục vụ bộ lọc tìm kiếm. |
| Quản lý đơn hàng | Xem danh sách đơn hàng, xác nhận đơn hàng và tìm |
| Quản lý khách hàng | Quản trị viên có thể xem thông tin khách hàng. |

### Phi chức năng.

* Bên cạnh các chức năng cốt lõi, hệ thống website cũng cần đảm bảo các yêu cầu phi chức năng nhằm tăng cường trải nghiệm người dùng, đảm bảo hiệu năng ổn định, bảo mật, mở rộng dễ dàng trong tương lai.

**Bảng 1‑3 Phi chức năng**

|  |  |
| --- | --- |
| **Tiêu chí** | **Yêu cầu cụ thể** |
| Tính tiến hóa | Cho phép thay đổi các quy định của hệ thống website. |
| Tính tiện dụng | Giao diện trực quan sinh động, người dùng dễ đọc, dễ sử dụng và phù hợp với thực tế. |
| Tính hiệu quả | Website load dữ liệu dễ dàng và nhanh chóng. |
| Tính tương thích | Tương thích các hệ điều hành Windows trở lên.  Hệ quản trị CSDL trên MySQL.  Ngôn ngữ lập trình HTML. |

## Mô tả các công nghệ

* Việc lựa chọn công nghệ phù hợp là một yếu tố then chốt giúp dự án website thời trang đạt được hiệu quả về mặt kỹ thuật lẫn trải nghiệm người dùng. Dưới đây là các công nghệ được sử dụng trong hệ thống, được phân loại theo từng lớp trong mô hình phát triển phần mềm: giao diện người dùng (frontend), xử lý nghiệp vụ (backend), cơ sở dữ liệu, và triển khai hệ thống.

### Giao diện người dùng (Frontend)

* Giao diện người dùng là phần người dùng trực tiếp tương tác, đòi hỏi thiết kế trực quan, dễ sử dụng và phản hồi nhanh chóng. Các công nghệ được sử dụng gồm:
* HTML & CSS:
* Sử dụng HTML5 để định nghĩa cấu trúc trang web, kết hợp với CSS3 để định dạng và trình bày giao diện.
* JavaScript:
* Được sử dụng để tạo các tương tác động trên trang như cập nhật giỏ hàng, kiểm tra form đầu vào, hiển thị popup, xử lý sự kiện người dùng...
* Hỗ trợ gọi các API từ backend để hiển thị dữ liệu động mà không cần reload trang.

### Xử lý phía máy chủ (Backend)

* Phần backend chịu trách nhiệm xử lý logic nghiệp vụ, giao tiếp với cơ sở dữ liệu và cung cấp dữ liệu cho frontend thông qua API.
* Node.js:
* Là nền tảng JavaScript chạy phía máy chủ, nổi bật nhờ hiệu năng cao, khả năng xử lý đồng thời mạnh mẽ.
* Được sử dụng để xây dựng máy chủ, xử lý các route (đường dẫn API), xử lý logic đơn hàng, xác thực người dùng...
* API:
* Là kiến trúc giao tiếp chuẩn giúp frontend (trang web) và các ứng dụng mobile (nếu có) có thể tương tác thống nhất với backend.
* API được thiết kế theo các phương thức HTTP như GET, POST, PUT, DELETE để tương ứng với các thao tác lấy, thêm, cập nhật, xóa dữ liệu.

### Cơ sở dữ liệu

* Dữ liệu là thành phần cốt lõi của hệ thống, bao gồm thông tin sản phẩm, đơn hàng, khách hàng và lịch sử giao dịch.
* MySQL:
* Là hệ quản trị cơ sở dữ liệu quan hệ, hỗ trợ ngôn ngữ SQL tiêu chuẩn.
* Đảm bảo tính toàn vẹn và nhất quán dữ liệu theo chuẩn ACID, phù hợp với mô hình dữ liệu phức tạp như sản phẩm có nhiều thuộc tính (size, màu), đơn hàng có nhiều dòng sản phẩm...

### Triển khai và công cụ hỗ trợ

* Để đảm bảo quy trình phát triển mượt mà, dễ triển khai và bảo trì, nhóm sử dụng các công cụ hiện đại hỗ trợ DevOps và CI/CD.
* Docker:
* Giúp container hóa toàn bộ ứng dụng (frontend, backend, database), đảm bảo môi trường nhất quán giữa máy phát triển và máy chủ thật.
* Hỗ trợ triển khai trên các nền tảng cloud hoặc local, dễ dàng tái sử dụng và mở rộng.
* Git & GitHub:
* Sử dụng Git để quản lý mã nguồn theo từng nhánh (branch), hỗ trợ làm việc nhóm hiệu quả và tránh xung đột mã nguồn.
* GitHub được dùng như một kho lưu trữ trung tâm, kết hợp tính năng Pull Request, Code Review và Issue Tracker.
* Tích hợp GitHub Actions để tự động hoá quy trình CI/CD: kiểm tra lỗi cú pháp, test mã nguồn, và tự động build-deploy ứng dụng khi có thay đổi.

# 

# PHÂN TÍCH

## Sơ đồ use case tổng quát

A diagram of a company

AI-generated content may be incorrect.

Hình 2‑1 **Sơ đồ use case tổng quát**

### Phân tích sơ đồ Use Case hệ thống quản lý bán hàng

#### Danh sách actor

Bảng 2‑1 **Bảng Danh Sách Actor**

|  |  |  |
| --- | --- | --- |
| **Mã Actor** | **Actor** | **Ý nghĩa** |
| A1 | Admin | Là người trực tiếp quản lý và sử dụng tất cả chức năng của website, admin cũng là một nhân viên. |
| A2 | Nhân viên | Là người sử dụng các chức năng quản lý của website và chịu sự quản lý của Admin, nhân viên cũng là một khách hàng. |
| A3 | Người dùng | Là người truy cập vào website, không cần đăng nhập và chỉ sử dụng được một số chức năng cơ bản trong website. |
| A4 | Thành viên | Cũng là một người dùng nhưng là người đã đăng nhập và sử dụng tất cả chức năng có trong website. |

#### Danh sách user

**Bảng 2‑2 Bảng danh sách user**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên Use-case** | **Actor** | **Ý nghĩa** |
| 1 | DangNhap | A3 | Đăng nhập vào website |
| 2 | DangKi | A3 | Đăng ký tài khoản |
| 3 | TimKiemSanPham | A3 | Tìm kiếm các sản phẩm trên website |
| 4 | QLGioHang | A3 | Quản lý giỏ hàng |
| 5 | ThemSPGioHang | A3 | Thêm sản phẩm vào giỏ hàng |
| 6 | CapNhatGioHang | A3 | Cập nhật giỏ hàng |
| 7 | XoaSPGioHang | A3 | Xóa sản phẩm trong giỏ hàng |
| 8 | XemSanPham | A3 | Xem sản phẩm |
| 9 | ĐangNhap | A3 | Đăng nhập |
| 10 | DangXuat | A4 | Đăng xuất |
| 11 | LapDonHang | A4 | Lập đơn hàng |
| 12 | QLDonHang | A4 | Quản lý đơn hàng |
| 13 | ThemDonHang | A4 | Thêm đơn hàng |
| 14 | SuaDonHang | A4 | Sửa đơn hàng |
| 15 | XoaDonHang | A4 | Xóa đơn hàng |
| 16 | ThanhToanDonHang | A4 | Thanh toán đơn hàng |
| 17 | QLSanPham | A2 | Quản lý sản phẩm |
| 18 | ThemSanPham | A2 | Thêm sản phẩm |
| 19 | SuaSanPham | A2 | Sửa sản phẩm |
| 20 | XoaSanPham | A2 | Xóa sản phẩm |
| 21 | QLLoaiSanPham | A2 | Quản lý loại sản phẩm |
| 22 | ThemLoaiSanPham | A2 | Thêm loại sản phẩm |
| 23 | SuaLoaiSanPham | A2 | Sửa loại sản phẩm |
| 24 | XoaLoaiSanPham | A2 | Xóa loại sản phẩm |
| 25 | QuanLyDonHang | A2 | Xác nhận đơn hàng |
| 26 | QuanLyTaiKhoan | A1 | Khóa mở tài khoản |

Các mối quan hệ giữa các Use Case

* <<include>>: Mối quan hệ này được sử dụng để tái sử dụng một use case phổ biến, ví dụ DangNhap là tiền đề bắt buộc của nhiều chức năng khác như quản lý đơn hàng, sản phẩm,...
* Không có mối quan hệ <<extend>>, điều này cho thấy hệ thống hoạt động theo các nhánh rõ ràng, không có hành vi mở rộng tùy theo điều kiện (ví dụ như xác minh 2 bước hoặc tùy chọn thanh toán)

## Sơ đồ use case chi tiết

### Quản trị viên

#### Quản lí tài khoản (Admin)

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2‑2 **Sơ đồ use case Quản Lý Tài Khoản**

* Tóm tắt:
* Use case "Quản lý Tài khoản" mô tả cách Admin quản lý các tài khoản trong hệ thống. Qua đó, Admin có thể thực hiện chức năng thay đổi trạng thái tài khoản .
* Diễn giải sơ đồ Use Case:
* Actor: Admin
* Use case chính: Quản lý Tài khoản
* Dòng sự kiện:Admin đăng nhập hệ thống và truy cập chức năng quản lý tài khoản.
* Hệ thống hiển thị danh sách tài khoản hiện có.
* Admin thực hiện chức năng thay đổi tài khoản:
  + Admin muốn thực hiện thay đổi trạng thái của tài khoản, Admin tìm kiếm tài khoản cần thay đổi trạng thái, chuyển từ hoạt động sang đã khóa và ngược lại.

#### Quản lí sản phẩm (Nhân viên)

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2‑3 **Sơ đồ use case Quản Lí Sản Phẩm**

* Tóm Tắt
* Use case mô tả cách nhân viên quản lý sản phẩm, nó cho thấy cách nhân viên thêm, sửa hoặc xóa một sản phẩm.
* Dòng sự kiện
  + - * Use case này bắt đầu khi một nhân viên muốn thêm, sửa hoặc xóa một sản phẩm.

Nhân viên vào màn hình Quản lý.

Nhân viên chọn menu Sản phẩm.

Hệ thống load danh sách sản phẩm lên màn hình.

* Nếu nhân viên muốn thêm sản phẩm nhấn chọn Thêm mới trên giao diện danh sách sản phẩm
* Hệ thống hiển thị màn hình Thêm mới sản phẩm với các thông tin liên quan với sản phẩm.
* Nhân viên nhập đầy đủ thông tin :

- Tên sản phẩm

- Giá tiền

- Hình ảnh

- Màu sắc

- Số lượng

- Loại sản phẩm

- Mô tả

* Nhấn nút Thêm.
* Nếu các thông tin nhập vào hợp lệ thì hệ thống sẽ thêm thông tin vào cơ sở dữ liệu, hiển thị danh sách sản phẩm lên màn hình danh sách Sản phẩm.
* Nếu nhân viên muốn sửa thông tin Sản phẩm:
* Nhân viên nhấn chọn link Sửa tương ứng với sản phẩm muốn sửa.
* Hệ thống hiển thị màn hình cập nhật thông tin sản phẩm đó, nhân viên thực hiện sửa thông tin và nhấn nút Cập nhật.
* Nếu thông tin sửa hợp lệ hệ thống sẽ cập nhật thông tin xuống cơ sở dữ liệu, công và hiển thị danh sách sản phẩm lên màn hình danh sách Sản phẩm.
* Nếu nhân viên muốn xóa sản phẩm:
* Nhân viên nhấn chọn link Xóa tương ứng với sản phẩm muốn xóa thì hệ thống sẽ đưa thông báo bạn muốn xóa sản phẩm này, nhân viên nhấn đồng ý thì hệ thống sẽ thực hiện xóa.
* Hê thống cập nhật lại danh sách các sản phẩm và hiển thị lên màn hình danh sách sản phẩm.
* Nếu nhân viên muốn tìm kiếm một sản phẩm:
* Nhân viên nhập thông tin sản phẩm muốn tìm vào thanh Tìm kiếm.
* Nếu thông tin đó tồn tại thì hệ thống sẽ hiển thị màn hình chứa đầy đủ các thông tin liên quan.
* Kết thúc usecase.

#### Quản lí loại sản phẩm (Nhân viên)

A diagram of a company

AI-generated content may be incorrect.

Hình 2‑4 **Sơ đồ use case Quản Lí Loại Sản Phẩm**

* Tóm Tắt
* Use case mô tả cách nhân viên quản lý loại sản phẩm, nó cho thấy cách nhân viên thêm, sửa hoặc xóa một loại sản phẩm.
* Dòng sự kiện
* Use case này bắt đầu khi một nhân viên muốn thêm, sửa hoặc xóa một loại sản phẩm.
* Nhân viên vào màn hình Quản lý.
* Nhân viên chọn menu Loại sản phẩm.
* Nếu nhân viên muốn thêm loại sản phẩm, nhập tên loại sản phẩm trên giao diện danh sách loại sản phẩm
* Nhấn nút Thêm.
* Thông tin nhập sẽ thêm vào cơ sở dữ liệu và hiển thị màn hình danh sách Loại sản phẩm.
* Nếu nhân viên muốn sửa thông tin Loại sản phẩm:
* Nhân viên tìm kiếm loại sản phẩm muốn sửa nhấn chọn sửa tương ứng với sản phẩm muốn sửa.
* Hệ thống hiển thị màn hình Cập nhật hiển thị các thông tin sản phẩm đó, nhân viên thực hiện sửa thông tin và nhấn nút Cập nhật.
* Nếu nhân viên muốn xóa Loại sản phẩm:
* Nhân viên tìm kiếm loại sản phẩm muốn xóa nhấn chọn xóa tương ứng với loại sản phẩm muốn xóa thì hệ thống đưa ra thông báo bạn chắc chắn muốn xóa lại sản phẩm này không, nhân viên nhấn đồng ý thì hệ thống sẽ thực hiện xóa.
* Hê thống cập nhật lại danh sách các loại sản phẩm và hiển thị lên màn hình danh sách loại sản phẩm.

#### Quản lí đơn đặt hàng (Nhân viên)

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2‑5 **Sơ đồ use case Quản Lí Đơn Đặt Hàng**

* Tóm Tắt
* Use case mô tả cách nhân viên quản lý đơn hàng, nó cho thấy các nhân viên có thể xác nhận đơn hàng .
* Dòng sự kiện
* Use case này bắt đầu khi một nhân viên muốn xác nhận đơn hàng.
* Nhân viên vào màn hình Quản lý.
* Nhân viên chọn menu Đơn hàng.
* Hệ thống load danh sách đơn hàng lên màn hình.
* Nếu nhân viên nhấn xác nhận đơn hàng thì phải tìm kiếm đơn hàng cần xác nhận, chuyển đơn hàng từ chờ xác nhận sang đã xác nhận.

### Khách hàng

#### Xem sản phẩm (Người dùng)

A diagram of a graph

AI-generated content may be incorrect.

Hình 2‑6  **Sơ đồ use case Xem Sản Phẩm**

* Tóm tắt:
* Use case “Xem Sản phẩm” mô tả chức năng cho phép người dùng hệ thống truy cập để xem danh sách các sản phẩm hiện có trong hệ thống.
* Diễn giải sơ đồ Use Case:
* Actor: NguoiDung
* Use Case chính: XemSanPham
* Dòng sự kiện:
* Người dùng truy cập hệ thống (không cần đăng nhập).
* Truy cập vào màn hình xem sản phẩm.
* Hệ thống tải và hiển thị danh sách sản phẩm đang có.
* Hệ thống hiển thị chi tiết thông tin sản phẩm: tên, mô tả, giá tiền, hình ảnh,...
* Use case kết thúc.

#### Tìm kiếm thông tin (Người dùng)

A graph with a red arrow pointing to a line

AI-generated content may be incorrect.

Hình 2‑7 **Sơ đồ use case Tìm Kiếm Thông Tin (người dùng)**

* Tóm tắt:
* Use case “Tìm kiếm thông tin” mô tả chức năng cho phép người dùng tìm kiếm dữ liệu cụ thể trong hệ thống thông qua từ khóa hoặc tiêu chí tìm kiếm.
* Diễn giải sơ đồ Use Case:
* Actor: NguoiDung
* Use Case chính: TimKiemThongTin
* Dòng sự kiện:
* Người dùng truy cập hệ thống.
* Người dùng nhập tên sản phẩm cần tìm vào ô tìm kiếm.
* Hệ thống kiểm tra và truy vấn dữ liệu từ cơ sở dữ liệu.
* Nếu tìm thấy dữ liệu khớp với từ khóa: Hệ thống hiển thị kết quả tìm kiếm với thông tin liên quan.
* Nếu không tìm thấy: Hệ thống thông báo “Không có sản phẩm phù hợp”.
* Use case kết thúc.

#### Đăng ký (Người dùng)

A diagram with a red arrow pointing to a circle

AI-generated content may be incorrect.

Hình 2‑8 **Sơ đồ use case Đăng Ký (người dùng)**

* Tóm tắt Use Case "Đăng Ký"
* Use case “Đăng Ký” mô tả quy trình người dùng tạo tài khoản mới trên hệ thống bằng cách cung cấp thông tin cá nhân như tên đăng nhập, mật khẩu, email, số điện thoại,... Việc đăng ký giúp người dùng truy cập các chức năng của hệ thống yêu cầu xác thực.
* Diễn giải sơ đồ Use Case
* Actor: NguoiDung – đại diện cho người dùng bên ngoài hệ thống.
* Use Case: DangKi – hành động đăng ký tài khoản.
* Mối quan hệ: Quan hệ trực tiếp, người dùng thực hiện hành động đăng ký thông qua giao diện hệ thống.
* Dòng sự kiện
* Người dùng truy cập vào hệ thống.
* Người dùng chọn chức năng “Đăng ký”.
* Hệ thống hiển thị biểu mẫu đăng ký.
* Người dùng nhập thông tin: tên đăng nhập, mật khẩu, email,...
* Nhấn nút “Đăng ký”.
* Hệ thống kiểm tra tính hợp lệ của thông tin:
  + Kiểm tra trường bắt buộc có được nhập hay không.
  + Kiểm tra định dạng email, độ mạnh mật khẩu.
  + Kiểm tra tên đăng nhập đã tồn tại chưa.
* Nếu hợp lệ:
  + Hệ thống lưu thông tin vào cơ sở dữ liệu.
  + Thông báo “Đăng ký thành công”.
* Use Case kết thúc.

#### Đăng nhập (Thành viên)

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2‑9 **Sơ đồ use case Đăng Nhập (thành viên)**

* Tóm tắt
* Use case “Đăng Ký” mô tả quy trình người dùng (ThanhVien) tạo tài khoản mới trên hệ thống. Sau khi đăng ký thành công, hệ thống sẽ tự động thực hiện chức năng “Đăng Nhập” để giúp người dùng truy cập vào hệ thống mà không cần nhập lại thông tin.
* Diễn giải sơ đồ Use Case
* Actors:
* ThanhVien: Đại diện cho người dùng có nhu cầu tạo tài khoản và sử dụng hệ thống.
* Use Cases:
* DangKi: Chức năng cho phép người dùng đăng ký tài khoản.
* DangNhap: Chức năng cho phép người dùng đăng nhập vào hệ thống.
* Mối quan hệ <<include>> giữa DangKi và DangNhap nghĩa là: quá trình đăng ký sẽ bao gồm luôn hành động đăng nhập – ví dụ, sau khi đăng ký thành công, hệ thống sẽ tự động đăng nhập tài khoản vừa tạo.
* Tóm tắt Use Case "Đăng Ký" (bao gồm Đăng Nhập)
* Mục đích: Cho phép thành viên tạo tài khoản mới và được tự động đăng nhập sau khi đăng ký thành công.
* Tác nhân chính: ThanhVien
* Kịch bản chính:
* Thành viên chọn chức năng “Đăng ký”.
* Nhập thông tin (tên, email, mật khẩu...).
* Hệ thống kiểm tra và lưu dữ liệu.
* Sau khi đăng ký thành công, hệ thống tự động gọi Use Case “Đăng nhập”.
* Thành viên được đưa vào giao diện sử dụng chính.
* Ý nghĩa kiến trúc
* Tái sử dụng logic "Đăng nhập": Thay vì viết lại hành động đăng nhập trong use case đăng ký, bạn chỉ định rằng "Đăng ký" bao gồm luôn "Đăng nhập" bằng <<include>>.
* Điều này thể hiện rõ ràng rằng "Đăng nhập" là một phần không thể thiếu trong quá trình "Đăng ký" (với hệ thống cho phép đăng nhập tự động sau khi đăng ký).

#### Đăng xuất (Thành viên)

A diagram with a red line and a yellow oval

AI-generated content may be incorrect.

Hình 2‑10 **Sơ đồ use case Đăng Xuất (thành viên)**

* Tóm tắt Use Case
* Use Case “Đăng Nhập” mô tả quá trình thành viên (đã có tài khoản) truy cập vào hệ thống bằng cách nhập tên đăng nhập và mật khẩu. Sau khi xác thực thành công, người dùng có thể sử dụng các chức năng của hệ thống. Ngoài ra, chức năng “Đăng Xuất” được bao gồm để đảm bảo người dùng có thể kết thúc phiên làm việc một cách an toàn.
* Diễn giải sơ đồ Use Case
* Actor: ThanhVien – đại diện cho người dùng đã đăng ký tài khoản và sử dụng hệ thống.
* Use Case chính:
  + DangNhap – hành động xác thực người dùng để truy cập hệ thống.
  + DangXuat – hành động kết thúc phiên làm việc.
* Mối quan hệ:
  + ThanhVien có quan hệ trực tiếp với DangNhap.
  + DangNhap include DangXuat: sau khi đăng nhập, hệ thống luôn bao gồm chức năng đăng xuất như một phần của phiên làm việc.
* Dòng sự kiện
* ThanhVien truy cập hệ thống.
* Chọn chức năng “Đăng nhập”.
* Hệ thống hiển thị giao diện nhập thông tin.
* Người dùng nhập tên đăng nhập và mật khẩu.
* Nhấn nút “Đăng nhập”.
* Sau khi sử dụng hệ thống, người dùng có thể thực hiện “Đăng xuất” để kết thúc phiên.
* Hệ thống hủy phiên làm việc và trở về giao diện ban đầu.
* Use Case kết thúc.

#### Lập đơn đặt hàng (Thành viên)

A graph with a red arrow pointing at a red line

AI-generated content may be incorrect.

Hình 2‑11 **Sơ đồ Lập Đơn Đặt Hàng (thành viên)**

* Tóm tắt Use Case “Lập Đơn Đặt Hàng”
* Use Case “Lập Đơn Đặt Hàng” mô tả quá trình thành viên đã đăng nhập chọn mua sản phẩm và tiến hành tạo đơn đặt hàng trên hệ thống. Quá trình này bao gồm việc lựa chọn sản phẩm, số lượng, nhấn mua hàng.
* Diễn giải sơ đồ Use Case
* Actor: ThanhVien – người dùng đã có tài khoản và đăng nhập hệ thống.
* Use Case chính: LapDonDatHang – thể hiện hành vi đặt hàng của người dùng trong hệ thống.
* Mối quan hệ: Mối quan hệ trực tiếp giữa ThanhVien và LapDonDatHang, thể hiện việc thành viên chủ động thao tác lập đơn hàng.
* Dòng sự kiện
* ThanhVien truy cập hệ thống (đã đăng nhập).
* Duyệt danh sách sản phẩm và chọn sản phẩm muốn mua.
* Nhập số lượng và thêm vào giỏ hàng.
* Chọn chức năng “Lập đơn đặt hàng”.
* Kết thúc Use Case.

## Sơ đồ phân tích cơ sở dữ liệu

A computer screen shot of a computer screen

AI-generated content may be incorrect.

Hình 2‑12 **Sơ đồ cơ sở dữ liệu ERD**

**Phân tích mối quan hệ**

### Mối quan hệ giữa người dùng và tài khoản.

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, hàng

Nội dung do AI tạo ra có thể không chính xác.

Hình 2‑13 **Mối quan hệ giữa người dùng và tài khoản**

* Một tài khoản thuộc về một nhân viên hoặc một khách hàng, 1 nhân viên hoặc 1 khách hàng chỉ có 1 tài khoản.
* Nhân viên lưu trữ các thông tin như **mã nhân viên**, họ tên, email, chức vụ, tài khoản
* Khách hàng lưu trữ các thông tin như **mã khách hàng**, họ tên, email, địa chỉ, tài khoản
* Tài khoản lưu trữ các thông tin như **tài khoản** mật khẩu, số điện thoại, trạng thái, ngày tạo, loại tài khoản

### Mối quan hệ giữa khách hàng và đơn hàng

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Nội dung do AI tạo ra có thể không chính xác.

Hình 2‑14 **Mối quan hệ giữa khách hàng và đơn hàng**

* Một khách hàng có thể có một hoặc nhiều đơn hàng, một đơn hàng chỉ thuộc về một khách hàng.
* Đơn hàng lưu trữ các thông tin như **mã đơn hàng**, ngày lập đơn, tổng tiền, trạng thái, tài khoản nhân viên(FK), tài khoản khách hàng(FK).

### Mối quan hệ giữa đơn hàng và sản phẩm

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Nội dung do AI tạo ra có thể không chính xác.

Hình 2‑15 **Mối quan hệ giữa đơn hàng và sản phẩm**

* Một đơn hàng có thể có nhiều sản phẩm, 1 sản phẩm có thể có nhiều đơn hàng( mối quan hệ nhiều-nhiều) tạo ra tập thực thể phát sinh là Chi tiết đơn hàng.
* Chi tiết đơn hàng lưu trữ các thông tin như **mã chi tiết đơn hàng**, số lượng, mã đơn hàng(FK), mã sản phẩm(FK).
* Sản phẩm lưu trữ các thông tin như **mã sản phẩm**, tên sản phẩm, ảnh sản phẩm, kích thước, màu sắc, số lượng, giá tiền, mô tả, mã loại sản phẩm(FK).

### Mối quan hệ giữa sản phẩm và loại sản phẩm

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, hàng

Nội dung do AI tạo ra có thể không chính xác.

Hình 2‑16 **Mối quan hệ giữa sản phẩm và loại sản phẩm**

* Một sản phẩm thuộc về 1 loại sản phẩm nhưng 1 loại sản phẩm có nhiều sản phẩm
* Sản phẩm lưu thông tin **mã sản phẩm**, tên sản phẩm, ảnh sản phẩm, kích thước, màu sắc, giá tiền, số lượng, mô tả, mã loại sản phẩm(FK).
* Loại sản phẩm bao gồm: **mã loại sản phẩm** và tên loại sản phẩm.

# THIẾT KẾ

## Sơ đồ tuần tự cho các nghiệp vụ thường xuyên

### Usecase “QLSanPham”

#### Usecase “ThemSanPham”

A diagram of a diagram

AI-generated content may be incorrect.

Hình 3‑1 **Sequence thêm sản phẩm thành công**

#### 

#### Usecase “SuaSanPham”

A diagram of a diagram

AI-generated content may be incorrect.

Hình 3‑2 **Sequence sửa sản phẩm thành công**

#### Usecase “XoaSanPham”

A diagram of a diagram

AI-generated content may be incorrect.

Hình 3‑3 **Sequence xóa sản phẩm thành công.**

### Usecase “TimKiemSanPham”

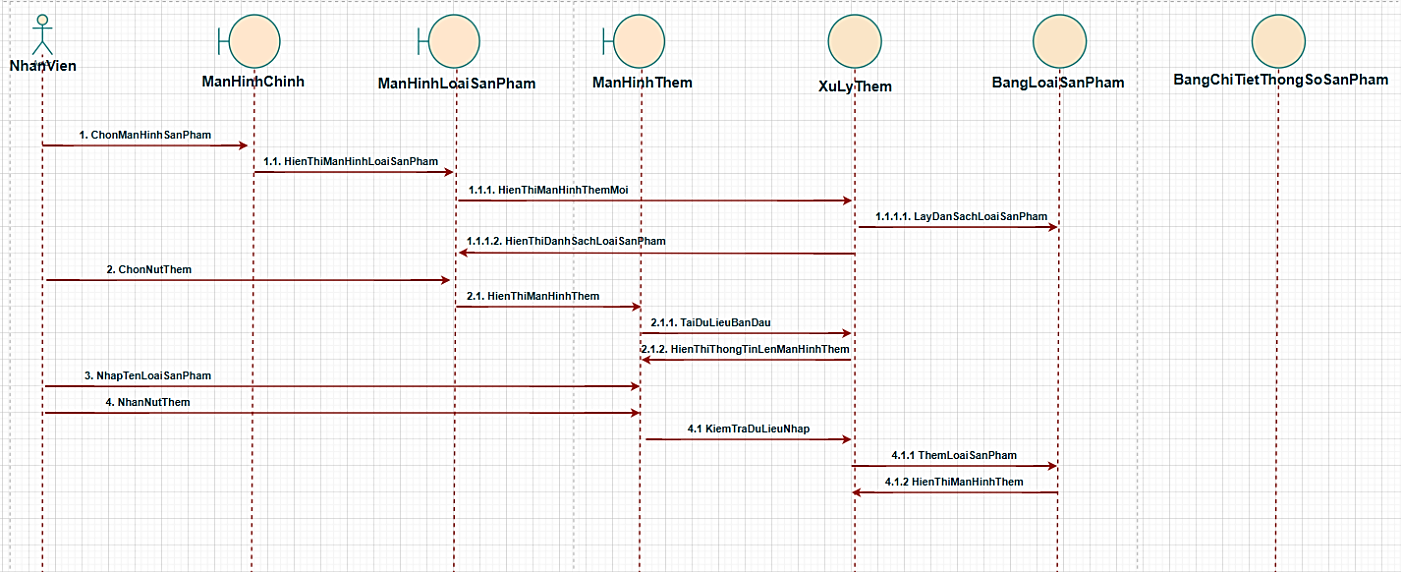
A diagram of a diagram

AI-generated content may be incorrect.

Hình 3‑4 **Sequence tìm kiếm sản phẩm thành công.**

### Usecase “QLLoaiSanPham”

#### Usecase “ThemLoaiSanPham”



Hình 3‑5 **Sequence thêm loại sản phẩm thành công**

#### Usecase “SuaLoaiSanPham”

A diagram of a project

AI-generated content may be incorrect.

Hình 3‑6 **Sequence sửa loại sản phẩm thành công**

#### Usecase “XoaLoaiSanPham”

A diagram of a chemical reaction

AI-generated content may be incorrect.

Hình 3‑7 **Sequence xóa loại sản phẩm thành công.**

### Usecase “DangNhap”

A diagram of a diagram

AI-generated content may be incorrect.

Hình 3‑8 **Sequence đăng nhập thành công.**

### Usecase “LapDonDatHang”

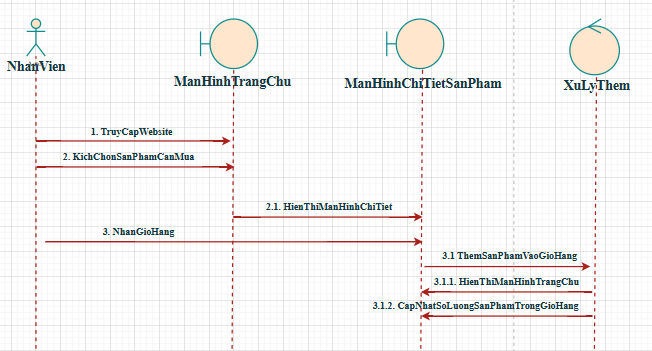
A diagram of a project

AI-generated content may be incorrect.

Hình 3‑9 **Sequence lập đơn đặt hàng thành công**

### Usecase “QLGioHang”

#### Usecase “ThemSanPhamVaoGioHang”



Hình 3‑10 **Sequence thêm sản phẩm vào giỏ hàng thành công**

#### Usecase “SuaSanPhamTrongGioHang”

A diagram of a diagram

AI-generated content may be incorrect.

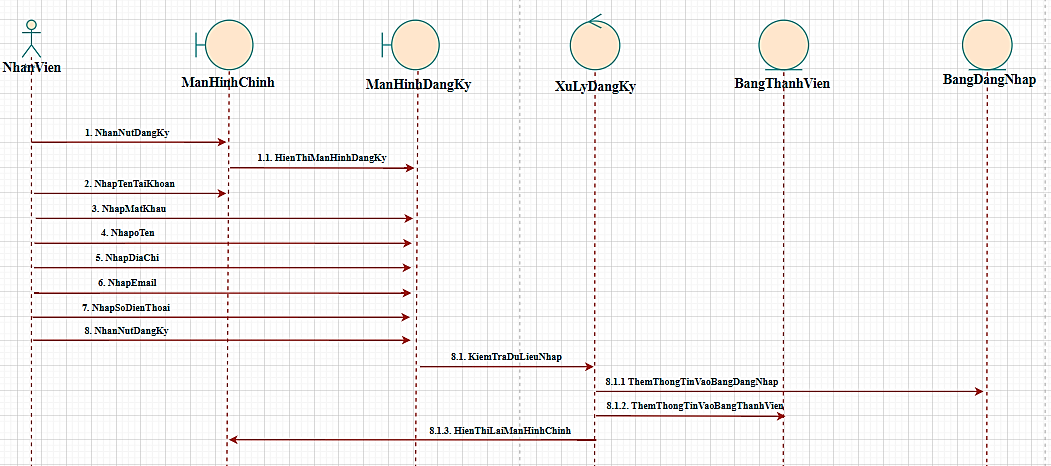
Hình 3‑11 **Sequence sửa sản phẩm trong giỏ hàng**

#### Usecase “XoaSanPhamTrongGioHang”



Hình 3‑12 **Sequence xóa sản phẩm trong giỏ hàng thành công.**

### Usecase “DangKyTaiKhoan”



Hình 3‑13 **Sequence đăng kí tài khoản thành công**

## Lược đồ quan hệ

### Lược đồ quan hệ.

A screenshot of a computer program

AI-generated content may be incorrect.

Hình 3‑14 **Lược đồ quan hệ**

* TaiKhoan(**taiKhoan**, matKhau, soDienThoai, trangThai, ngayTao, loaiTaiKhoan).
* NhanVien(**maNhanVien**, hoTen, email, chucVu, taiKhoan(FK)).
* KhachHang(**maKhachHang**, hoTen, email, diaChi, taiKhoan(FK)).
* DonHang(**maDonHan**g, ngayLapDon, tongTien, trangThai, taiKhoanNhanVien(FK), taiKhoanKhachHang(FK)).
* SanPham(**maSanPham**, tenSanPham, anhSanPham, kichThuoc, mauSac, giaTien, soLuong, moTa, maLoaiSanPham(FK)).
* ChiTietDonHang(**maChiTietDonHang**, soLuong, tongTien, maDonHang(FK), maSanPham(FK)).
* LoaiSanPham(**maLoaiSanPham**, tenLoaiSanPham)

### Mô tả chi tiết lược đồ cơ sở dữ liệu quan hệ.

**Bảng 3‑1 Tài khoản**

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| **taiKhoan** | Int | Tài khoản (Primary Key). |
| matKhau | Varchar(255) | Mật khẩu tài khoản. |
| soDienThoai | Varchar(20) | Số điện thoại. |
| trangThai | TinyInt | Trạng thái. |
| ngayTao | Date | Ngày tạo tài khoản. |
| loaiTaiKhoan | Enum | Loại tài khoản. |

**Bảng 3‑2 Nhân viên**

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| maNhanVien | Int | Mã nhân viên (Primary Key). |
| hoTen | Varchar(100) | Họ tên nhân viên. |
| email | Varchar(100) | Email nhân viên. |
| chucVu | Varchar(50) | Chức vụ. |
| taiKhoan | Int | Tài khoản (Foreign Key). |

**Bảng 3‑3 Khách hàng**

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| **maKhachHang** | Int | |  | | --- | |  |   Mã khách hàng (Primary Key) |
| hoTen | Varchar(100) | Họ tên khách hàng |
| Email | Varchar(100) | Email khách hàng |
| diaChi | Varchar(255) | Địa chỉ khách hàng |
| taiKhoan | Int | Tài khoản (Foreign Key) |

**Bảng 3‑4 Đơn hàng**

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| **maDonHang** | Int | Mã đơn hàng(Primary Key) |
| ngayLapDon | Date | Ngày lập đơn |
| tongTien | Decimal(15,2) | Tổng tiền |
| trangThai | Varchar(50) | Trạng thái |
| taiKhoanNhanVien | Int | Tài khoản nhân viên(Foreign key) |
| taiKhoanKhachHang | Int | Tài khoản khách hàng(Foreign key) |

**Bảng 3‑5 Sản phẩm**

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| maSanPham | Int | Mã sản phẩm (Primary Key) |
| tenSanPham | Varchar(100) | Tên sản phẩm |
| anhSanPham | Varchar(255) | Ảnh sản phẩm |
| kichThuoc | Varchar(50) | Kích thước |
| mauSac | Varchar(50) | Màu sắc |
| soLuong | Int | Số lượng |
| giaTien | Decimal(15,0) | Giá tiền |
| moTa | Text | Mô tả |
| maLoaiSanPham | Int | Mã loại sản phẩm |

**Bảng 3‑6 Chi tiết đơn hàng**

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| **maChiTietDonHang** | Int | Mã chi tiết đơn hàng(Primary Key) |
| soLuong | Int | Số lượng |
| tongTien | Decimal(15,2) | Tổng tiền |
| maDonHang | Int | Mã đơn hàng(Foreign key) |
| maSanPham | Int | Mã sản phẩm(Foreign key) |

**Bảng 3‑7 Loại sản phẩm**

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| **maLoaiSanPham** | Int | Mã loại sản phẩm(Primary Key) |
| tenLoaiSanPham | Varchar(100) | Tên loại sản phẩm |

# TRIỂN KHAI HỆ THỐNG TRÊN HẠ TẦNG CLOUD

## Giới thiệu chung

* Sau khi hoàn tất quá trình xây dựng và kiểm thử hệ thống website bán quần áo thời trang, nhóm em tiến hành triển khai hệ thống lên môi trường thực tế nhằm phục vụ người dùng một cách ổn định và hiệu quả. Để đáp ứng các yêu cầu về khả năng mở rộng linh hoạt, độ tin cậy cao cũng như dễ dàng trong việc quản lý và bảo trì, nhóm đã lựa chọn nền tảng Amazon Web Services (AWS) làm hạ tầng triển khai chính.
* Kiến trúc hệ thống bao gồm ba thành phần cốt lõi: frontend (giao diện người dùng), backend (xử lý logic nghiệp vụ), và cơ sở dữ liệu (lưu trữ thông tin ). Toàn bộ các thành phần được đóng gói trong các container Docker, giúp dễ dàng triển khai, cập nhật và mở rộng khi cần thiết.
* Cơ sở dữ liệu sử dụng hệ quản trị MySQL và được triển khai trên dịch vụ Amazon RDS – một dịch vụ cơ sở dữ liệu được quản lý hoàn toàn bởi AWS. Việc sử dụng RDS không chỉ đảm bảo tính sẵn sàng cao, mà còn giúp bảo vệ dữ liệu một cách an toàn với các tính năng sao lưu tự động, mã hóa và khả năng mở rộng linh hoạt.

## 

## Mô hình triển khai trên cloud.

Ảnh có chứa văn bản, biểu đồ, ảnh chụp màn hình, Kế hoạch

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑1 **Sơ đồ triển khai trên cloud**

## Mô tả mô hình

### Các dịch vụ sử dụng

* Virtual Private Cloud(VPC): là một mạng riêng ảo trên AWS, giúp người dùng tạo ra môi trường mạng ảo hóa, trong đó các tài nguyên AWS có thể giao tiếp với nhau một cách an toàn. VPC cho phép cấu hình các subnet, route table, internet gateway, và các cấu hình bảo mật như security group. Trong mô hình này, VPC giúp kết nối các dịch vụ như EC2, RDS và các container với nhau một cách bảo mật và hiệu quả.
* EC2 Instance (Elastic Compute Cloud): là dịch vụ cung cấp máy chủ ảo (virtual server) trên AWS. Mỗi EC2 instance là một máy chủ chạy trên cơ sở hạ tầng đám mây của AWS. Trong mô hình này, EC2 được sử dụng để chạy các container Docker (Nginx, frontend, backend). Các container này xử lý các yêu cầu từ người dùng, bao gồm việc phục vụ giao diện người dùng, xử lý logic nghiệp vụ và kết nối với cơ sở dữ liệu.
* Amazon RDS (Relational Database Service): là dịch vụ quản lý cơ sở dữ liệu quan hệ trên AWS, giúp dễ dàng thiết lập, vận hành và mở rộng cơ sở dữ liệu trong đám mây. Trong mô hình này, RDS được sử dụng để lưu trữ dữ liệu của hệ thống như thông tin người dùng, đơn hàng, và sản phẩm. Cơ sở dữ liệu RDS sử dụng MySQL và được đặt trong một private subnet để đảm bảo an toàn, chỉ có EC2 mới có thể truy cập vào cơ sở dữ liệu thông qua cổng 3306 (MySQL) và các cấu hình bảo mật được thiết lập trong security group.

### Mô tả chi tiết mô hình.

* Khi người dùng truy cập vào hệ thống thông qua trình duyệt, các yêu cầu truy cập sẽ được gửi qua Internet và đi qua Internet Gateway. Từ Internet Gateway, yêu cầu được định tuyến đến EC2 Instance, nơi hệ thống chính của website đang vận hành. Trên EC2, các thành phần frontend, backend và Nginx được container hóa bằng Docker. Đầu tiên, Nginx Container sẽ tiếp nhận tất cả các yêu cầu. Tùy theo loại yêu cầu, Nginx sẽ chuyển tiếp đến thành phần phù hợp: các yêu cầu hiển thị giao diện như HTML, CSS, JS sẽ được chuyển đến Frontend Container, còn các yêu cầu liên quan đến xử lý logic nghiệp vụ như đăng nhập, đặt hàng hoặc truy xuất dữ liệu sẽ được chuyển đến Backend Container. Tại đây, backend sẽ kết nối đến cơ sở dữ liệu MySQL được triển khai trên Amazon RDS trong một subnet riêng tư để truy vấn hoặc ghi dữ liệu. Sau khi xử lý hoàn tất, phản hồi được gửi lại theo chiều ngược lại: từ Backend đến Nginx, qua Internet Gateway và cuối cùng trả về trình duyệt của người dùng. Toàn bộ quy trình này được tối ưu thông qua kiến trúc microservice và triển khai container, đảm bảo hiệu suất và khả năng mở rộng cao cho hệ thống.

## Triển khai website trên cloud .

* Source code được đóng gói bằng Docker và được đưa lên GitHub để dễ dàng triển khai và quản lý.
* Liên kết GitHub:
* Frontend: <https://github.com/VuDuyHoang22050038/FE.git>
* Backend: <https://github.com/VuDuyHoang22050038/BE.git>

### Khởi tạo dịch vụ VPC

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑2 **Cấu trúc VPC dashboard để đẩy web lên cloud AWS**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑3 **Cấu hình VPC**

* Để xây dựng một kiến trúc mạng riêng cho hệ thống Clined - Server trên nền tảng AWS, nhóm em tiến hành khởi tạo một VPC có tên là **Nhom4** với dải địa chỉ IP là **10.10.0.0/16**.

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑4 **Cấu hình Subnets**

* Sau khi tạo VPC Nhom4 với dải địa chỉ IP 10.10.0.0/16, hệ thống được chia thành 4 subnet tương ứng với từng tầng chức năng chính.

**Bảng 4‑1 Vai trò từng subnet**

|  |  |  |
| --- | --- | --- |
| **Subnet** | **Địa chỉ IP** | **Vai trò** |
| Subnet-Frontend | 10.10.1.0/24 | Chứa EC2 frontend (giao diện) |
| Subnet-Backend | 10.10.1.0/24 | Chứa EC2 backend (API, xử lý) |
| Subnet-DB-1 | 10.10.1.0/24 | |  | | --- | |  |   Chứa RDS instance |
| Subnet-DB-2 | 10.10.1.0/24 | |  | | --- | |  |   Chứa RDS instance |

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Trang web

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑5 **Cấu hình Internet Gateway (IGW)**

* Để các subnet Frontend và Backend có thể truy cập Internet, ta cần tạo một Internet Gateway (IGW) và gắn nó vào VPC Nhom4.

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑6 **Cấu hình Route Table**

* Tiếp theo, tạo Route Table để thiết lập định tuyến mạng cho các subnet trong VPC Nhom4, nhằm đảm bảo các thành phần trong hệ thống có thể kết nối đúng cách theo kiến trúc thiết kế.

### Khởi tạo dịch vụ RDS.

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑7 **Cấu hình DB Subnet Group**

* Trước khi tạo cơ sở dữ liệu, chúng ta cần tạo một DB Subnet Group.

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑8 **Cấu hình Database**

* Tiếp theo, chúng ta tiến hành tạo một cơ sở dữ liệu MySQL chạy trên hệ điều hành Linux.

### Khởi tạo dịch vụ EC2

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑9 **Cấu hình Instance**

* Trong dịch vụ EC2, chúng ta khởi tạo hai instance, một cho frontend (fe) và một cho backend (be).

### Đẩy Frontend và Backend lên EC2.

* Sau khi khởi tạo thành công hai EC2 instances, chúng ta cần thực hiện cấu hình hai máy EC2: một cho frontend và một cho backend.
* Chạy lệnh sudo yum install docker -y để tiến hành cài đặt Docker trên hệ thống.
* Tiến hành cài đặt Git trên hệ thống bằng cách chạy lệnh sudo yum install git -y.
* Để thiết lập quyền sử dụng Docker cho người dùng hiện tại, hãy chạy lệnh sudo usermod -aG docker $USER.
* Để khởi động Docker, hãy sử dụng lệnh sudo systemctl start docker.
* Tiếp theo chúng ta tiếng hành sao chép dự án từ GitHub.
* Sử dụng lệnh ls để kiểm tra xem thư mục dự án đã được sao chép thành công hay chưa.
* Đối với frontend
* Sử dụng lệnh sudo docker build -t frontend . để build image với tên là frontend từ Dockerfile trong thư mục hiện tại.
* Sau khi build xong, sử dụng lệnh sudo docker run -d -p 8080:80 frontend để khởi chạy container frontend và ánh xạ cổng 80 trong container ra cổng 8080 trên máy chủ.
* Đối với backend
* Dùng lệnh nano .env để mở tệp cấu hình môi trường và chỉnh sửa các thông số cần thiết.

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, màu đen

Nội dung do AI tạo ra có thể không chính xác.

Hình 4‑10 Khởi chạy Backend

* Sau khi kiểm tra xong nhấn tổ hợp phím Ctrl + O để lưu tệp, Enter để xác nhận, và Ctrl + X để thoát khỏi trình chỉnh sửa.
* Sử dụng lệnh sudo docker run -d -p 4000:4000 backend để chạy container backend ở chế độ nền, ánh xạ cổng 4000 trên máy host với container.
* Tiến hành tạo cơ sở dữ liệu và nạp dữ liệu ban đầu từ mã nguồn có sẵn.
* Cấu hình Inbound và Outbound Rules cho các EC2 instances để cho phép các kết nối đến và đi.

# KẾT LUẬN

## Kết quả triển khai

* Sau quá trình nghiên cứu và phát triển, nhóm đã hoàn thành việc xây dựng một hệ thống website bán hàng trực tuyến dành cho các cửa hàng thời trang vừa và nhỏ. Hệ thống đã được triển khai thành công trên nền tảng cloud, đảm bảo khả năng truy cập mọi lúc, mọi nơi. Website đáp ứng các chức năng cốt lõi như:
* Quản lý sản phẩm: thêm/sửa/xóa sản phẩm, hiển thị thông tin chi tiết (hình ảnh, mô tả, giá, size, màu).
* Quản lý đơn hàng: theo dõi trạng thái đơn hàng theo thời gian thực.
* Quản lý khách hàng: lưu trữ thông tin khách hàng và lịch sử mua hàng.
* Tự động hóa quy trình bán hàng: thêm sản phẩm vào giỏ hàng, đặt hàng.
* Quản lý kho hàng: cập nhật tồn kho tự động theo đơn hàng phát sinh.
* Giao diện người dùng thân thiện: hỗ trợ tìm kiếm, lọc sản phẩm, và tạo tài khoản người dùng.

## Ưu điểm

* Tự động hóa toàn diện: Hệ thống đã thay thế hoàn toàn các quy trình thủ công, giảm thiểu sai sót trong nhập liệu và xử lý đơn hàng.
* Quản lý tập trung và đồng bộ: Dữ liệu sản phẩm, khách hàng và đơn hàng được lưu trữ và cập nhật tập trung, dễ dàng truy xuất và xử lý.
* Tăng trải nghiệm người dùng: Giao diện đơn giản, dễ sử dụng, tích hợp nhiều tính năng hỗ trợ mua hàng nhanh chóng và thuận tiện.
* Khả năng mở rộng: Hệ thống triển khai trên nền tảng cloud giúp dễ dàng mở rộng quy mô mà không phụ thuộc vào hạ tầng phần cứng tại chỗ.

## Hạn chế

* Chưa tích hợp thực tế các nền tảng bán hàng lớn như Shopee hoặc Zalo, mới dừng ở mức thiết kế kiến trúc.
* Thiếu chức năng phân tích dữ liệu nâng cao: như báo cáo doanh thu theo thời gian, phân tích hành vi khách hàng, gợi ý sản phẩm.
* Bảo mật và phân quyền còn cơ bản: chưa có cơ chế phân quyền chi tiết cho từng vai trò người dùng như quản trị viên, nhân viên bán hàng, v.v.

## Phương hướng phát triển

* Hoàn thiện tích hợp đa kênh thực tế: Kết nối trực tiếp với API các sàn TMĐT như Shopee, Tiki, Lazada để đồng bộ hóa dữ liệu đơn hàng và tồn kho.
* Nâng cấp hệ thống báo cáo và thống kê: Thêm dashboard quản trị với biểu đồ phân tích doanh số, lượng truy cập, sản phẩm bán chạy...
* Tăng cường bảo mật và hiệu năng: Áp dụng các cơ chế phân quyền nâng cao, bảo vệ dữ liệu người dùng và tối ưu hiệu suất xử lý đơn hàng lớn.
* Ứng dụng trí tuệ nhân tạo (AI): Gợi ý sản phẩm dựa trên hành vi mua hàng, phân loại khách hàng, tối ưu chiến dịch khuyến mãi.
* Phát triển ứng dụng di động: Hỗ trợ người dùng mua hàng mọi lúc, mọi nơi qua ứng dụng Android/iOS.

**TÀI LIỆU KHAM KHẢO**

1. **Tài liệu về kiến trúc 3 lớp (Three-Tier Architecture)   
   Nguồn**: Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach*.
2. **Tài liệu về giao thức HTTP/HTTPS trong giao tiếp Web   
   Nguồn**: Fielding, R., & Reschke, J. (2014). *RFC 7230: Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*.
3. **Tài liệu về hệ quản trị cơ sở dữ liệu quan hệ (MySQL, PostgreSQL)   
   Nguồn**: Elmasri, R., & Navathe, S. B. (2017). *Fundamentals of Database Systems* (7th Edition).
4. **Tài liệu về bảo mật web và xác thực người dùng   
   Nguồn**: Stallings, W. (2017). *Cryptography and Network Security: Principles and Practice*.
5. **Tài liệu về quy trình phát triển phần mềm (SDLC – Software Development Life Cycle)   
   Nguồn**: Sommerville, I. (2016). *Software Engineering* (10th Edition).
6. **Tài liệu về trải nghiệm người dùng (UX/UI Design)   
   Nguồn**: Krug, S. (2014). *Don't Make Me Think: A Common Sense Approach to Web Usability*.