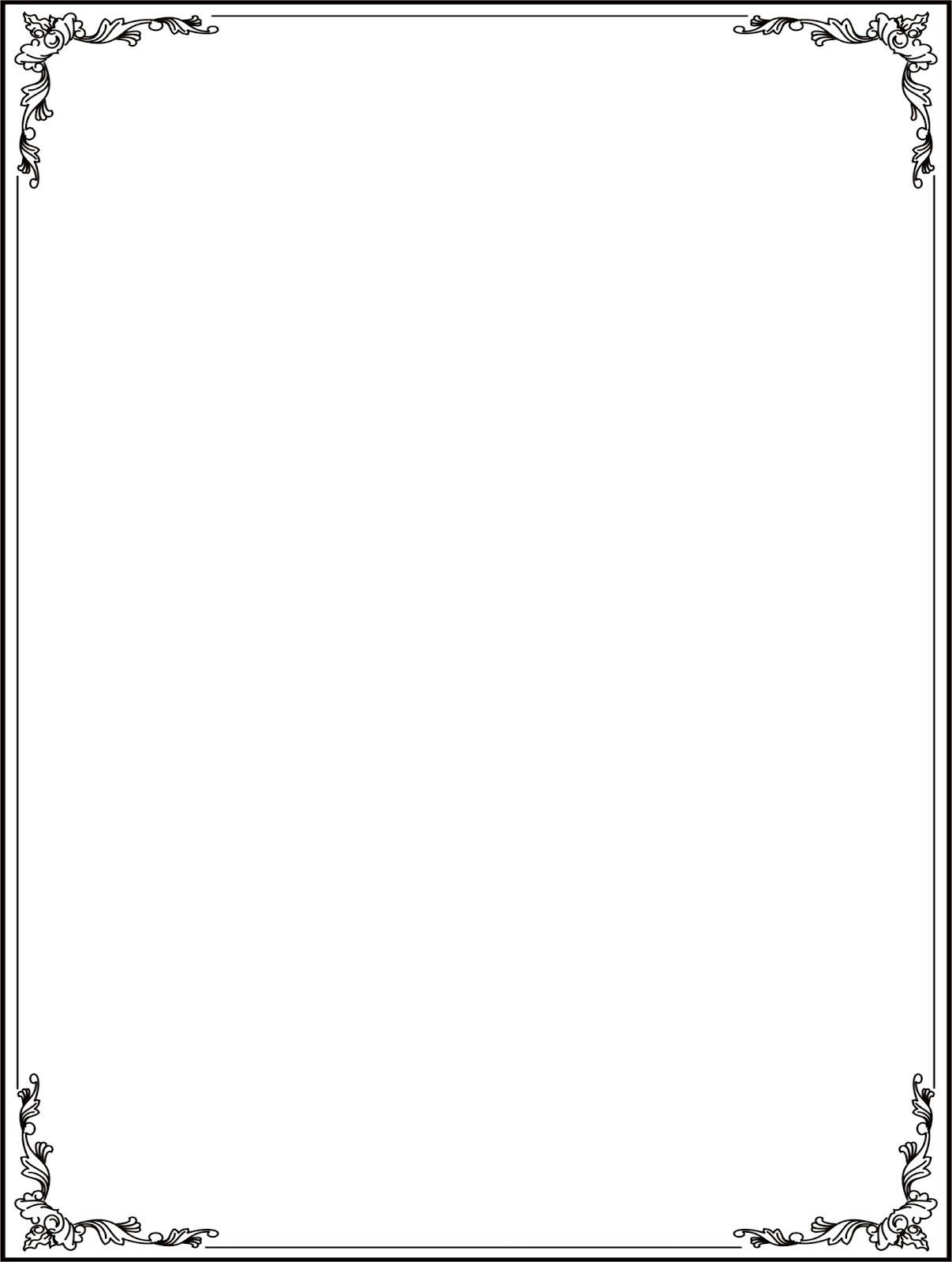
# 



**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN CÔNG NGHỆ PHẦN MỀM**

\*\*\*\*\*\*\*\*



**ĐỒ ÁN MÔN HỌC**

**THIẾT KẾ KIẾN TRÚC PHẦN MỀM**

**Đề tài**

**Quản Lý Kho Hàng**

Nhóm sinh viên thực hiện: Nhóm 15

Danh sách sinh viên: **Vũ Văn Nam-0198366**

**Võ Duy Quý - 0202066**

Lớp học phần: 66PM

Giảng viên hướng dẫn: Phạm Hữu Tùng

**Hà Nội, Ngày 26/5**

**Mục lục**

[**1. Giới thiệu**](#_txehux7pyosa) **4**

[1.1 Lý do chọn đề tài](#_v01vd3p5puwo) 4

[1.2 Mục tiêu đồ án](#_pqhsli2jicy7) 4

[1.3. Phạm vi](#_twew901fxni2) 4

[1.4. Công cụ và Công nghệ sử dụng](#_tx85dwu54uf8) 5

[**2. Phân tích thiết kế hệ thống**](#_mduck6ka3qxf) **5**

[2.1. Mô tả bài toán](#_syhqb9iz9ive) 5

[2.2. Yêu cầu chức năng](#_fq3woknnrtxf) 5

[2.3. Yêu cầu phi chức năng](#_yowt2yqdvmea) 6

[2.4 Biểu đồ tổng quát](#_zfcq811crjmf) 7

[2.4.1 Usecase Quản lý sản phẩm](#_gy1xlnr9j3gt) 8

[2.4.2 Usecase Quản lý thể loại](#_dvdi34o39pxr) 8

[2.4.3 Usecase Quản lý nhà cung cấp](#_yfs685703fca) 9

[2.4.5 Usecase Quản lý nhân viên](#_p3rf5pxqfa10) 9

[2.4.6 Usecase Quản lý khách hàng](#_akci691zmxwy) 10

[2.5) Đặc tả Usecase](#_u8u71ogevcum) 10

[2.5.1 Đăng nhập](#_cczi0svhal8s) 10

[2.5.2 Đăng xuất](#_97fflvmedcu9) 11

[2.5.3 Quản lý thể loại](#_7gs9jjh71m23) 13

[2.5.4 Quản lý nhà cung cấp](#_trbslt1o17d2) 16

[2.5.5 Quản lý khách hàng](#_ewycnzw5fdwk) 20

[2.5.6 Quản lý sản phẩm](#_o7r5gyt7xwmn) 23

[2.6 Biểu đồ hoạt động](#_kj2zysuqp9jd) 28

[2.6.1 Quản lý sản phẩm](#_6t76mbs52hd9) 28

[2.6.2 Quản lý khách hàng](#_1cpvutym776k) 29

[2.6.3 Quản lý nhà cung cấp](#_tyuqvheuipiz) 30

[2.6.4 Quản lý nhân viên](#_wf4pmex0gjij) 31

[2.6.5 Quản lý nhập hàng](#_rfozk19o2kol) 32

[2.6.6 Quản lý xuất hàng](#_j7472r8ibnh2) 33

[2.6.7 Báo cáo, thống kê](#_sriszp5uz30v) 34

[**3: Thiết kế hệ thống**](#_uofgqq2yl85i) **34**

[3.1 Mô hình kiến trúc](#_kcowaombdmc) 34

[3.2 Sơ đồ kiến trúc tổng thể](#_ubpze7l08gdz) 35

[3.3 Biểu đồ tuần tự các chức năng chính](#_ijwwgihrit22) 36

[3.3.1 Đăng nhập](#_7sdegmkob9pi) 36

[3.3.2 Đăng ký](#_sy9guzv96ktr) 36

[3.3.3 Thêm khách hàng](#_tfrr8smfbi5k) 37

[3.3.4 Sửa khách hàng](#_en4tsymj9z2g) 38

[3.3.5 Xóa khách hàng](#_uopu90shs8lk) 38

[3.3.6 Xem chi tiết khách hàng](#_usfeg65i0yr4) 38

[3.3.7 Xem danh sách khách hàng](#_v756e04edlo7) 39

[3.3.8 Thêm sản phẩm](#_rqm8bvjkuvvm) 40

[3.3.9 Sửa sản phẩm](#_2lv5fhtn5ktz) 40

[3.3.10 Xóa sản phẩm](#_k6b3kbuvxiij) 41

[3.3.11 Xem chi tiết sản phẩm](#_iq0z4vhftup1) 41

[3.3.12 Xem danh sách sản phẩm](#_k3n26qvdpinj) 42

[3.3.13 Nhập hàng](#_8pf64zlozr0q) 42

[3.3.13 Hiển thị doanh thu](#_hut7f5uogl17) 43

[3.3.14 Thêm người dùng](#_iyaw6ec5dw7j) 43

[3.3.15 Sửa thông tin người dùng](#_yfes27iwgqy) 44

[3.3.14 Xóa người dùng](#_21d776m1ncce) 44

[3.4 Class Diagram](#_v5b4msagt1do) 45

[3.5 Sơ đồ quan hệ ERD](#_did3wqlj1ac2) 46

[3.6 Thiết kế giao diện người dùng](#_4wstbzsr7m46) 46

[3.6.1 Giao diện sản phẩm](#_69vefbsm1a5f) 46

[3.6.2 Giao diện quản lý khách hàng](#_bzgcfv70avj0) 47

[3.6.3 Giao diện quản lý đơn hàng nhập](#_pns7rdhle0tr) 47

[3.6.4 Giao diện quản lý đơn hàng đã xóa](#_am40s6pap2kd) 48

[3.6.5 Giao diện quản lý xuất hàng](#_wca3iepll4sz) 48

[3.6.6 Giao diện hóa đơn đã xóa](#_925o863a8i7c) 49

[3.6.7 Giao diện quản lý doanh thu](#_c5q3bk438nb5) 49

[3.6.8 Giao diện quản lý người dùng](#_4p5pvaf1djy2) 50

[**4. Cài đặt và triển khai hệ thống**](#_bs70wvoqus43) **50**

[4.1 Môi trường trường triển khai](#_bycxqsoi2hr6) 50

[4.2 Yêu cầu hệ thống](#_19axbnwuuiy5) 50

[4.3 Các bước cài đặt chương trình](#_lopf4d3581gg) 51

[**Chương 5. Kết quả và đánh giá**](#_tjoinks3c9no) **52**

[5.1. Kết quả thử nghiệm hệ thống](#_r0ny6jfnceeu) 52

[5.2. Đánh giá hiệu quả hệ thống](#_wsqfjuhfboq6) 53

[**6. Tài liệu tham khảo**](#_bwx5e5mxz62u) **55**

# 

# **1. Giới thiệu**

## 1.1 Lý do chọn đề tài

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ, việc quản lý kho hàng bằng phương pháp thủ công hoặc sử dụng các công cụ truyền thống như Excel không còn đáp ứng hiệu quả yêu cầu quản lý chính xác, nhanh chóng và đồng bộ. Các doanh nghiệp, đặc biệt là doanh nghiệp vừa và nhỏ, ngày càng cần đến những hệ thống quản lý kho hàng chuyên nghiệp, có khả năng cập nhật số liệu theo thời gian thực, hỗ trợ nhập – xuất hàng, kiểm kê và báo cáo tồn kho một cách dễ dàng và chính xác.

Laravel là một framework PHP phổ biến với cú pháp rõ ràng, bảo mật cao, hỗ trợ xây dựng ứng dụng web nhanh chóng và linh hoạt theo mô hình MVC. Việc sử dụng Laravel để xây dựng hệ thống quản lý kho hàng giúp tận dụng được nhiều ưu điểm như khả năng mở rộng, dễ bảo trì, tích hợp API và hệ thống phân quyền người dùng.

Vì vậy, em chọn đề tài “Xây dựng hệ thống quản lý kho hàng” nhằm tìm hiểu và ứng dụng kiến thức lập trình web, đồng thời giải quyết bài toán thực tiễn trong quản lý kho hàng.

## 1.2 Mục tiêu đồ án

* Xây dựng một hệ thống quản lý kho hàng với các chức năng cơ bản như:  
  + Quản lý sản phẩm và danh mục sản phẩm.  
  + Quản lý tồn kho: nhập kho, xuất kho.  
  + Quản lý nhà cung cấp và khách hàng.  
  + Quản lý người dùng, phân quyền truy cập.  
  + Thống kê, báo cáo tổng quan về hàng hóa và hoạt động kho.
* Ứng dụng framework Laravel để:  
  + Phát triển hệ thống theo mô hình MVC.  
  + Đảm bảo tính bảo mật và hiệu năng của ứng dụng.  
  + Dễ dàng mở rộng, bảo trì và tích hợp các chức năng trong tương lai.  
  + Củng cố và nâng cao kỹ năng lập trình backend với PHP và Laravel.

+ Nâng cao khả năng phân tích, thiết kế cơ sở dữ liệu và tổ chức mã nguồn theo hướng chuẩn hóa.

## 1.3. Phạm vi

Hệ thống quản lý kho hàng được thiết kế để hỗ trợ doanh nghiệp trong việc theo dõi và quản lý hàng hóa một cách hiệu quả. Người dùng chính của hệ thống bao gồm quản trị viên, nhân viên kho và khách hàng nội bộ. Các chức năng chính của hệ thống bao gồm:

* Quản lý danh mục sản phẩm (tạo, sửa, xóa sản phẩm)
* Quản lý thể loại sản phẩm(đồ gia dụng, đồ điện tử, đồ nội thất,…)
* Quản lý nhập kho (thêm hàng, cập nhật số lượng, theo dõi lịch sử nhập hàng).
* Quản lý xuất kho (xuất hàng, kiểm soát số lượng, theo dõi lịch sử xuất hàng).
* Kiểm kê tồn kho và cảnh báo hàng tồn kho thấp.
* Quản lý nhà cung cấp và khách hàng.
* Quản lý tài khoản người dùng.
* Tạo báo cáo tổng quan về hàng tồn kho, lịch sử giao dịch, hiệu suất kho hàng.

## 1.4. Công cụ và Công nghệ sử dụng

Để thực hiện đồ án, nhóm em áp dụng các phương pháp sau:

* Phân tích yêu cầu: Thu thập và phân tích yêu cầu từ thực tế quản lý kho hàng để xác định các chức năng chính cần triển khai.
* Thiết kế hệ thống:  
  + Thiết kế sơ đồ cơ sở dữ liệu (ERD).  
  + Thiết kế giao diện người dùng (UI) và kiến trúc tổng thể của hệ thống theo mô hình MVC.
* Lập trình và triển khai:  
  + Sử dụng Laravel để xây dựng hệ thống backend.  
  + Sử dụng Blade để xây dựng giao diện frontend đơn giản, thân thiện.  
  + Áp dụng nguyên lý SOLID và các design pattern phù hợp để tổ chức mã nguồn.
* Kiểm thử và đánh giá:  
  + Thực hiện kiểm thử chức năng và hiệu năng của hệ thống.  
  + Thử nghiệm trên dữ liệu thực tế để đánh giá tính ứng dụng.

# **2. Phân tích thiết kế hệ thống**

## 2.1. Mô tả bài toán

Trong quá trình hoạt động, các doanh nghiệp luôn cần theo dõi tình trạng hàng hóa trong kho để đảm bảo hoạt động sản xuất – kinh doanh diễn ra hiệu quả. Tuy nhiên, việc quản lý thủ công dễ dẫn đến tình trạng sai sót số liệu, thất thoát hàng hóa hoặc chậm trễ trong việc xử lý đơn hàng.

Bài toán đặt ra là xây dựng một hệ thống quản lý kho hàng có khả năng:

* Lưu trữ và quản lý thông tin sản phẩm.
* Ghi nhận các hoạt động nhập, xuất, kiểm kê hàng hóa.
* Quản lý mối quan hệ với nhà cung cấp và khách hàng.
* Hỗ trợ báo cáo số lượng tồn kho, lịch sử giao dịch, và cảnh báo số lượng hàng thấp.

Mục tiêu là tạo ra một hệ thống hoạt động ổn định, giao diện dễ sử dụng và có khả năng mở rộng trong tương lai.

### 

## 2.2. Yêu cầu chức năng

Hệ thống cần đảm bảo các chức năng chính sau:

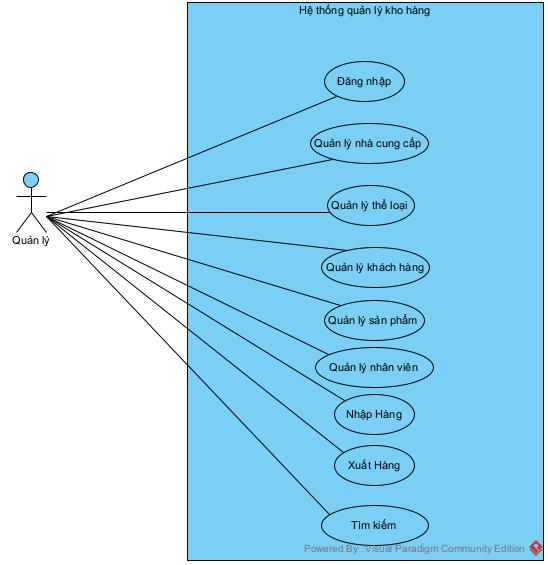
* Quản lý danh mục sản phẩm
* Thêm, sửa, xóa danh mục sản phẩm.
* Quản lý sản phẩm
* Thêm, sửa, xóa sản phẩm.
* Mỗi sản phẩm có thông tin: tên, mô tả, đơn vị, số lượng, giá, ảnh, danh mục,...
* Quản lý kho hàng
* Nhập kho: Ghi nhận thông tin hàng nhập, nhà cung cấp, ngày nhập, số lượng,...
* Xuất kho: Ghi nhận thông tin xuất hàng, khách hàng, ngày xuất,...
* Lịch sử giao dịch: Theo dõi chi tiết các lần nhập/xuất của từng sản phẩm.
* Quản lý nhà cung cấp và khách hàng
* Lưu thông tin liên hệ, địa chỉ, lịch sử giao dịch.
* Báo cáo – Thống kê
* Báo cáo tồn kho theo từng sản phẩm.
* Thống kê số lượng hàng nhập/xuất theo thời gian.

## 2.3. Yêu cầu phi chức năng

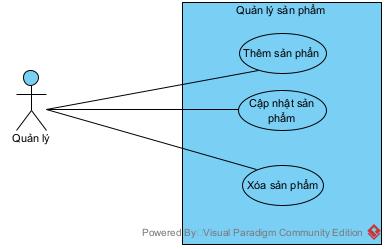
Bên cạnh các chức năng chính, hệ thống cần đảm bảo các yêu cầu phi chức năng sau:

* Tính bảo mật: Hệ thống phải bảo vệ dữ liệu người dùng, phân quyền truy cập hợp lý, tránh rò rỉ dữ liệu.
* Tính dễ sử dụng: Giao diện trực quan, thao tác dễ hiểu, hướng đến người dùng phổ thông.
* Tính ổn định và hiệu năng: Hệ thống xử lý nhanh, hoạt động ổn định kể cả khi có nhiều dữ liệu.
* Khả năng mở rộng: Có thể dễ dàng thêm các module mới như tích hợp mã vạch, hệ thống cảnh báo, hoặc kết nối với phần mềm kế toán.
* Tính tương thích: Hệ thống hoạt động tốt trên trình duyệt phổ biến, không phụ thuộc vào thiết bị.

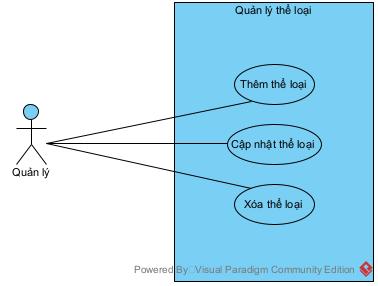
## 2.4 Biểu đồ tổng quát

****

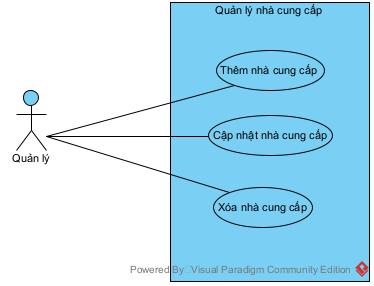
### 2.4.1 Usecase Quản lý sản phẩm

****

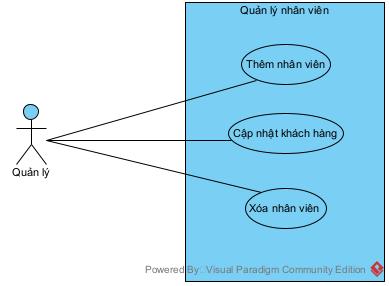
### 2.4.2 Usecase Quản lý thể loại

****

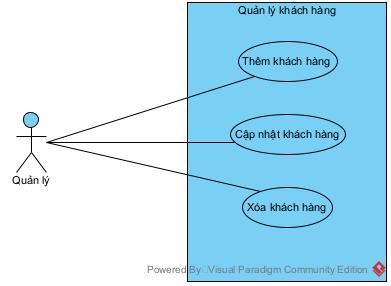
### 2.4.3 Usecase Quản lý nhà cung cấp

****

### 2.4.5 Usecase Quản lý nhân viên

****

### 2.4.6 Usecase Quản lý khách hàng

****

## 2.5) Đặc tả Usecase

### 2.5.1 Đăng nhập

| ID | UC01 |
| --- | --- |
| Name | Đăng nhập |
| Brief description | Chức năng cho phép người dùng đã có tài khoản quản lý đăng nhập vào hệ thống |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã có tài khoản quản lý |
| Basic flow | 1. Người dùng vào trang đăng nhập  2. Người dùng điền thông tin  3. Người dùng ấn gửi  4. Hệ thống sẽ kiểm tra xem tài khoản có tồn tại trong hệ thống hay không, nếu có tồn tại thì chuyển hướng đến trang chủ còn nếu không tồn tại sẽ thông báo đăng nhập sai |
| Alternative flow |  |
| Result | Người dùng đăng nhập được vào hệ thống |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

### 2.5.2 Đăng xuất

| ID | UC02 |
| --- | --- |
| Name | Đăng xuất |
| Brief description | Chức năng cho phép người dùng đăng xuất khỏi hệ thống |
| Actors | Tình nguyện viên, tổ chức, admin |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng bấm vào Avatar góc trái màn hình  2. Người dùng chọn đăng xuất  3. Hệ thống hỏi lại xem người dùng có muốn đăng xuất hay không  4. Người dùng đồng ý đăng xuất  5. Hệ thống đăng xuất tài khoản và chuyển hướng đến trang đăng nhập |
| Alternative flow |  |
| Result | Người dùng đăng xuất thành công |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

### 2.5.3 Quản lý thể loại

2.5.3.1 Thêm thể loại

| ID | UC331 |
| --- | --- |
| Name | Thêm thể loại |
| Brief description | Chức năng cho phép admin thêm thể loại mới vào hệ thống |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn nút thêm thể loại  2. Hệ thống hiển thị form nhập dữ liệu  3. Người dùng nhập dữ liệu và ấn gửi  4. Hệ thống sẽ kiểm tra dữ liệu đầu vào nếu đúng thì sẽ thêm thể loại mới nếu không đúng sẽ thông báo sai định dạng |
| Alternative flow |  |
| Result | Quản lý thêm được thể loại mới |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

2.5.3.2 Sửa thể loại

| ID | UC332 |
| --- | --- |
| Name | Sửa thể loại |
| Brief description | Chức năng cho phép quản lý sửa được thể loại cũ |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn thể loại muốn sửa  2. Hệ thống hiển thị dữ liệu thể loại cũ để người dùng sửa  3. Người dùng sửa dữ liệu và ấn gửi  4. Hệ thống sẽ kiểm tra dữ liệu đầu vào nếu đúng thì sửa thể loại nếu sai sẽ thông báo |
| Alternative flow |  |
| Result | Người dùng sửa được thể loại |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

2.5.3.3 Xóa thể loại

| ID | UC333 |
| --- | --- |
| Name | Xóa thể loại |
| Brief description | Chức năng cho phép người dùng xóa thể loại đã tồn tại |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn thể loại muốn xóa  2. Người dùng ấn nút xóa  3. Hệ thống hỏi lại có muốn xóa không  4. Người dùng ấn đồng ý  5. Hệ thống xóa thể loại |
| Alternative flow |  |
| Result | Người dùng xóa được thể loại |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

### 2.5.4 Quản lý nhà cung cấp

2.5.4.1 Thêm nhà cung cấp

| ID | UC341 |
| --- | --- |
| Name | Thêm nhà cung cấp |
| Brief description | Chức năng cho phép người dùng thêm nhà cung cấp mới |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn thêm mới nhà cung cấp  2. Hệ thống hiển thị form thêm mới  3. Người dùng nhập thông tin  4. Hệ thống sẽ kiểm tra xem dữ liệu nhập vào có đúng định dạng không nếu đúng thì thêm mới nếu sai thì thông báo |
| Alternative flow |  |
| Result | Người dùng thêm được nhà cung cấp mới |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

2.5.4.2 Sửa nhà cung cấp

| ID | UC342 |
| --- | --- |
| Name | Sửa nhà cung cấp |
| Brief description | Chức năng cho phép người dùng sửa nhà cung cấp |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn nhà cung cấp muốn sửa và chọn sửa  2. Hệ thống hiển thị form sửa  3. Người dùng sửa  4. Hệ thống sẽ kiểm tra xem dữ liệu có đúng không nếu đúng thì sửa nếu sai thì báo lỗi |
| Alternative flow |  |
| Result | Người dùng sửa được nhà cung cấp |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

2.5.4.3 Xóa nhà cung cấp

| ID | UC343 |
| --- | --- |
| Name | Xóa nhà cung cấp |
| Brief description | Chức năng cho phép người dùng xóa nhà cung cấp |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn nhà cung cấp muốn xóa  2. Người dùng chọn nút xóa  3. Hệ thống hỏi lại có muốn xóa không  4. Người dùng xác nhận xóa  5. Hệ thống xóa nhà cung cấp |
| Alternative flow |  |
| Result | Người dùng xóa được nhà cung cấp |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

### 2.5.5 Quản lý khách hàng

2.5.5.1 Thêm khách hàng

| ID | UC351 |
| --- | --- |
| Name | Thêm khách hàng |
| Brief description | Chức năng cho phép người dùng thêm khách hàng mới |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn thêm khách hàng  2. Hệ thống hiển thị form thêm mới  3. Người dùng nhập thông tin khách hàng  4. Hệ thống sẽ kiểm tra xem dữ liệu nếu đúng thì thêm mới nếu sai thì thông báo |
| Alternative flow |  |
| Result | Người dùng thêm mới được khách hàng |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

2.5.5.2 Sửa khách hàng

| ID | UC352 |
| --- | --- |
| Name | Sửa khách hàng |
| Brief description | Chức năng cho phép người dùng sửa khách hàng đã tồn tại |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn khách hàng muốn sửa  2. Hệ thống hiển thị form và thông tin khách hàng  3. Người dùng sửa dữ liệu  4. Hệ thống sẽ kiểm tra xem dữ liệu có đúng định dạng không nếu đúng thì sửa thông tin khách hàng nếu sai thì thông báo lỗi |
| Alternative flow |  |
| Result | Người dùng sửa được thông tin khách hàng |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

2.5.5.3 Xóa khách hàng

| ID | UC353 |
| --- | --- |
| Name | Xóa khách hàng |
| Brief description | Chức năng cho phép người dùng xóa khách hàng |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn khách hàng muốn xóa  2.Hệ thống hỏi lại có muốn xóa không  3. Người dùng ấn xác nhận  4. Hệ thống xóa khách hàng |
| Alternative flow |  |
| Result | Người dùng xóa được khách hàng |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

### 2.5.6 Quản lý sản phẩm

2.5.6.1 Thêm sản phẩm

| ID | UC061 |
| --- | --- |
| Name | Thêm sản phẩm |
| Brief description | Chức năng cho phép người dùng thêm sản phẩm vào hệ thống |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn thêm sản phẩm  2. Hệ thống hiển thị form thêm mới  3. Người dùng nhập thông tin sản phẩm  4. Hệ thống sẽ kiểm tra xem dữ liệu nếu đúng thì thêm mới nếu sai thì thông báo |
| Alternative flow |  |
| Result | Người dùng thêm sản phẩm thành công |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

2.5.6.2 Cập nhật sản phẩm

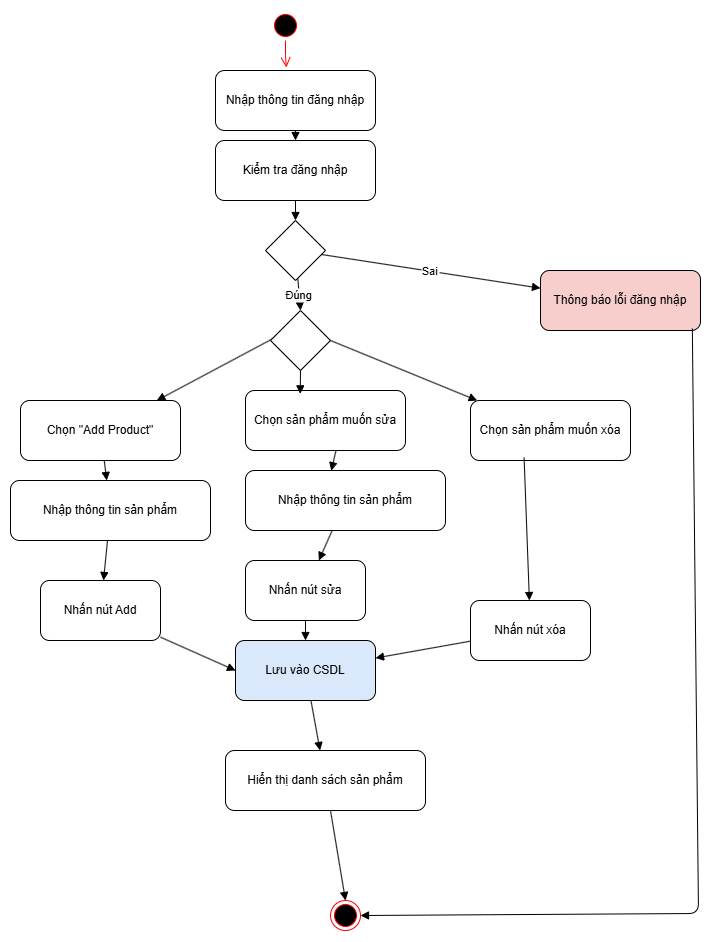
| ID | UC062 |
| --- | --- |
| Name | Cập nhật sản phẩm |
| Brief description | Chức năng cho phép người dùng cập nhật lại thông tin sản phẩm |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn sản phẩm muốn sửa  2. Hệ thống hiển thị form và thông tin khách hàng  3. Người dùng sửa dữ liệu  4. Hệ thống sẽ kiểm tra xem dữ liệu có đúng định dạng không nếu đúng thì sửa thông tin khách hàng nếu sai thì thông báo lỗi |
| Alternative flow |  |
| Result | Người dùng sửa được thông tin khách hàng |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

2.5.6.3 Xóa sản phẩm

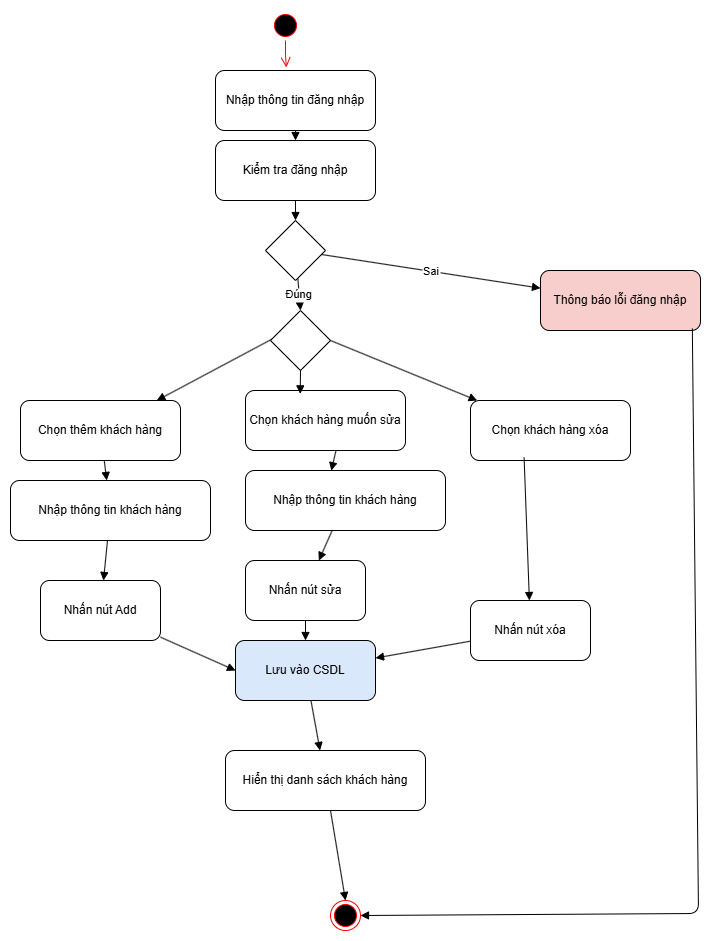
| ID | UC063 |
| --- | --- |
| Name | Xóa sản phẩm |
| Brief description | Chức năng cho phép người dùng xóa sản phẩm |
| Actors | Quản lý |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng chọn sản phẩm muốn xóa  2.Hệ thống hỏi lại có muốn xóa không  3. Người dùng ấn xác nhận  4. Hệ thống xóa sản phẩm |
| Alternative flow |  |
| Result | Người dùng xóa được khách hàng |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

## 2.6 Biểu đồ hoạt động

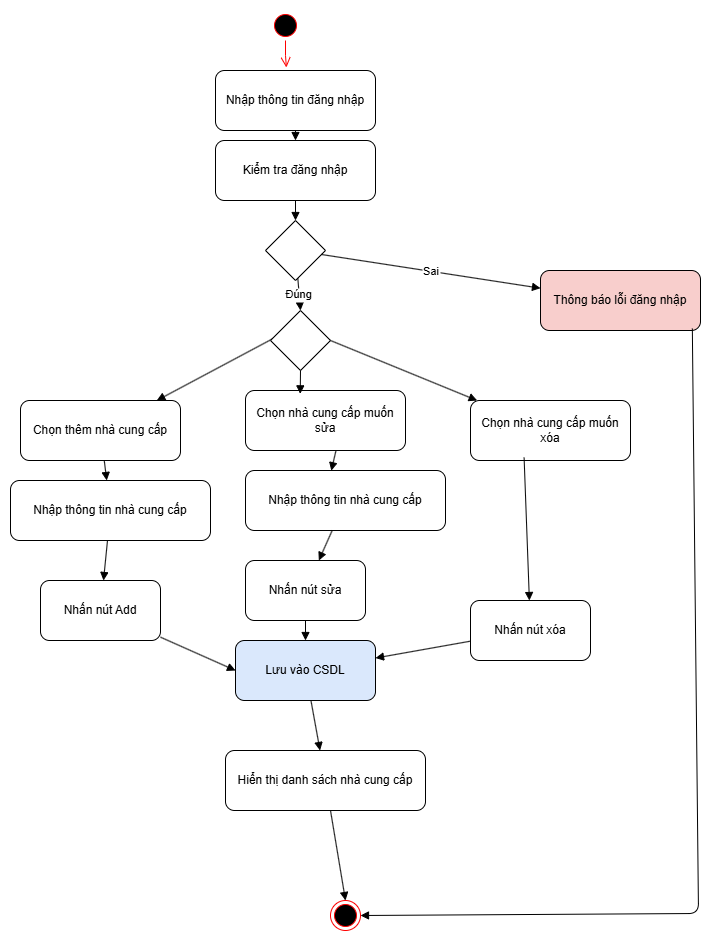
### 2.6.1 Quản lý sản phẩm

****

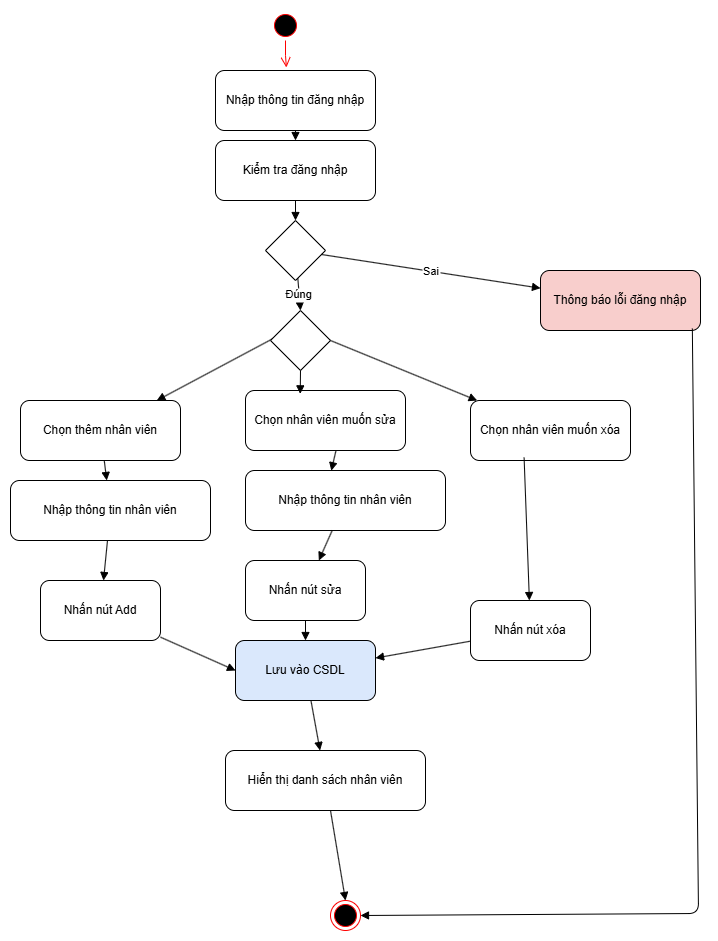
### 2.6.2 Quản lý khách hàng

****

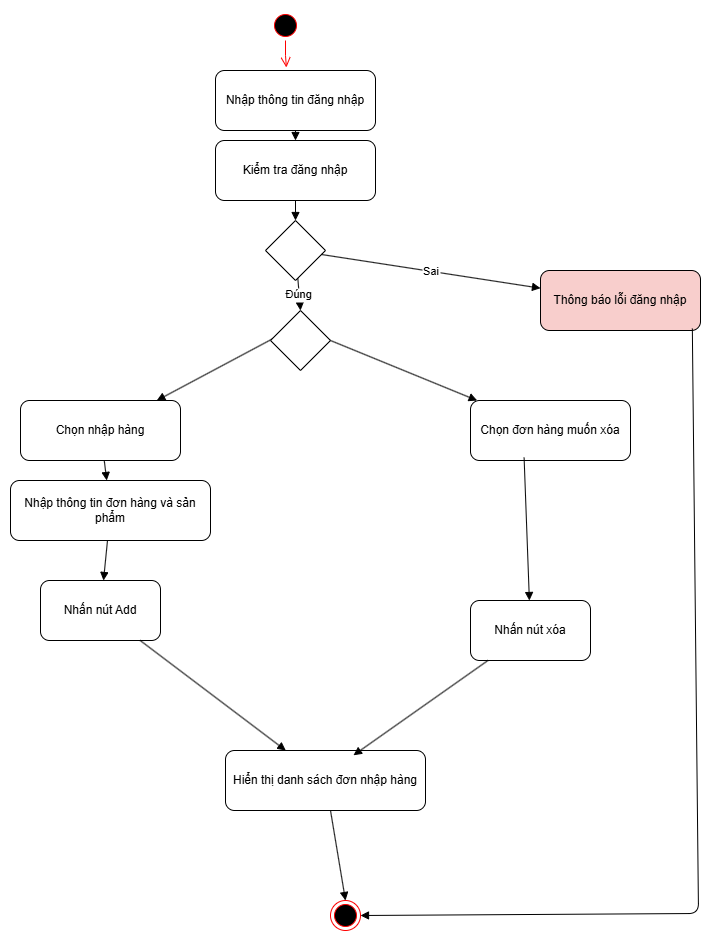
### 2.6.3 Quản lý nhà cung cấp

****

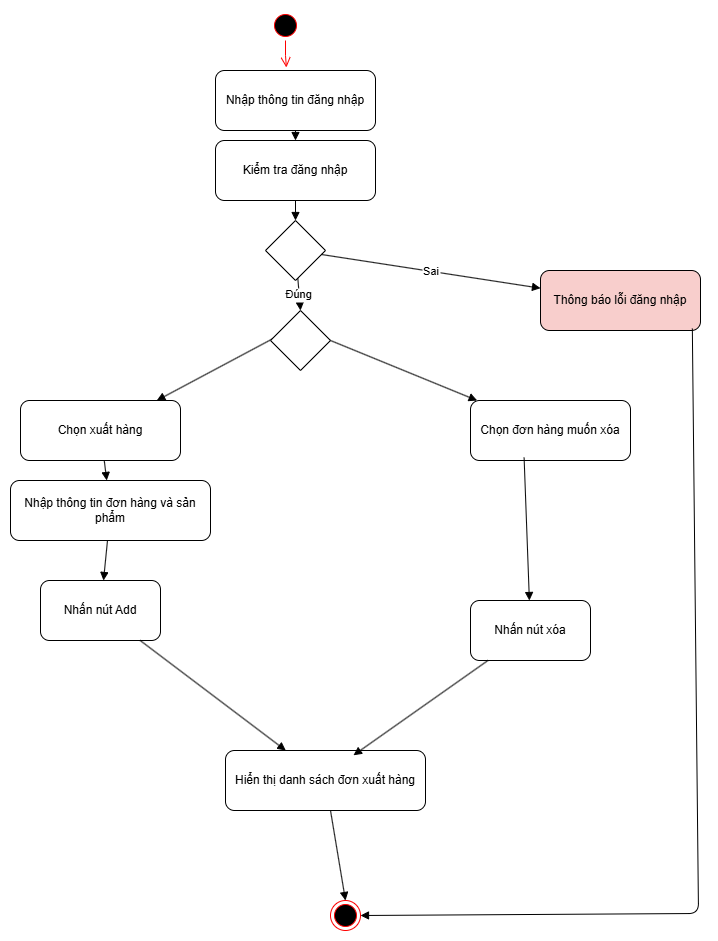
### 2.6.4 Quản lý nhân viên

****

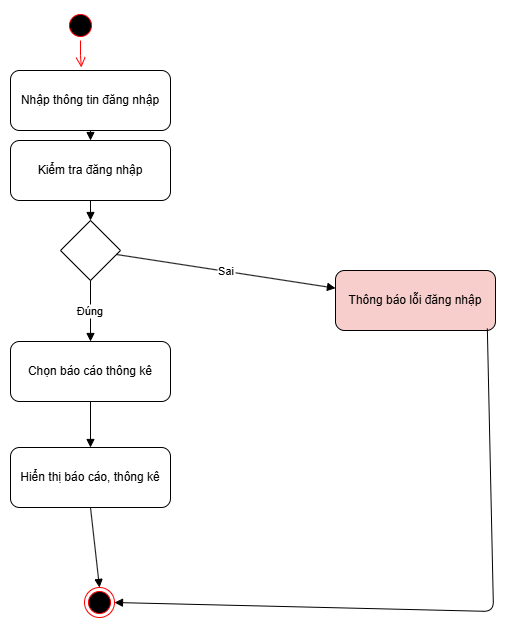
### 2.6.5 Quản lý nhập hàng

****

### 2.6.6 Quản lý xuất hàng

****

### 2.6.7 Báo cáo, thống kê

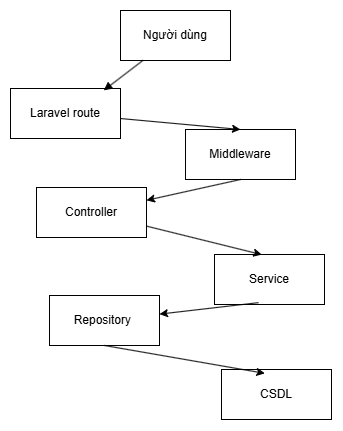
****

# **3: Thiết kế hệ thống**

## 3.1 Mô hình kiến trúc

Trong quá trình xây dựng hệ thống quản lý kho hàng, nhóm đã lựa chọn mô hình kiến trúc MVC (Model - View - Controller) làm kiến trúc chính cho hệ thống. Đây là một mô hình kiến trúc phổ biến và được hỗ trợ mạnh mẽ trong framework Laravel, giúp tách biệt rõ ràng giữa các thành phần của ứng dụng.

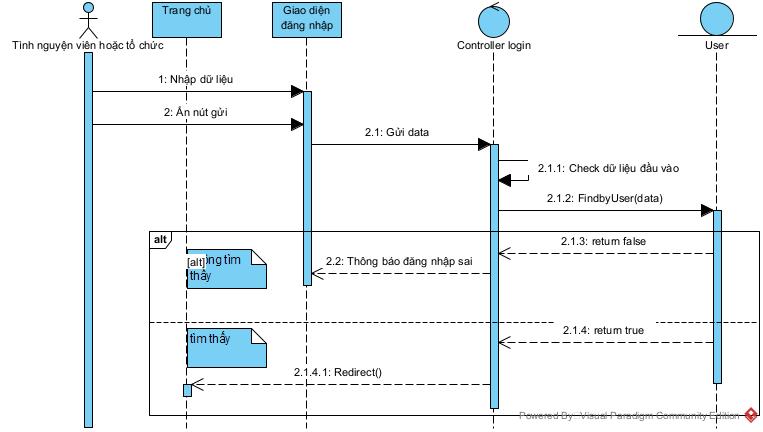
## 3.2 Sơ đồ kiến trúc tổng thể

****

* Người dùng: gửi yêu cầu đến hệ thống
* Laravel route: định tuyến, xác định yêu cầu nào sẽ được gửi tới controller nào
* Middleware: kiểm tra xác thực người dùng
* Controller: nơi tiếp nhận request từ người dùng sau khi được Middleware và Route xử lý. Nó hoạt động như một cầu nối giữa tầng giao diện và logic xử lý nghiệp vụ
* Service: tầng chứa các logic nghiệp vụ
* Repository: tầng truy xuất dữ liệu

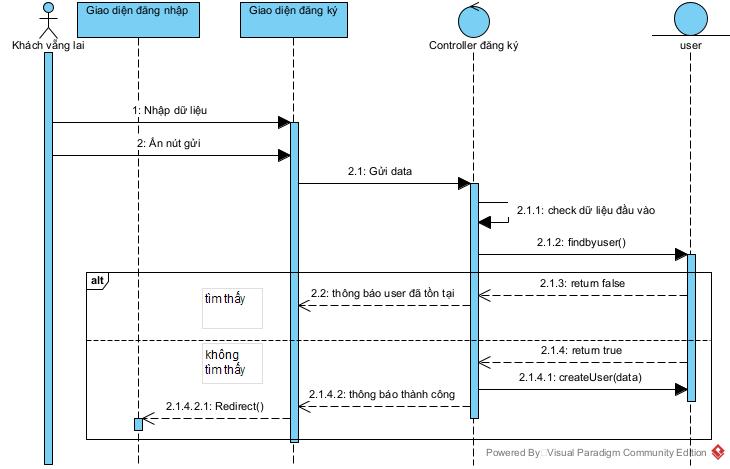
## 3.3 Biểu đồ tuần tự các chức năng chính

#### **3.3.1 Đăng nhập**

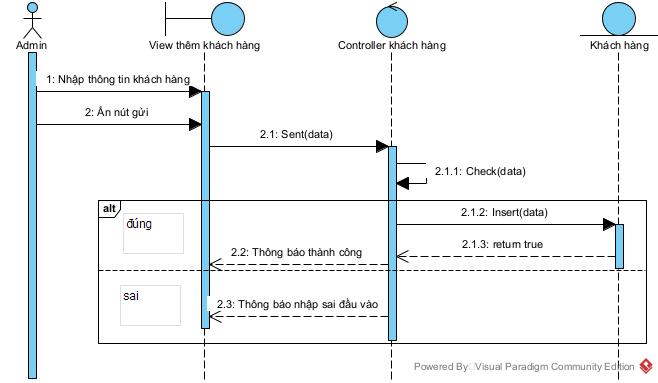


#### 

#### 3.3.2 Đăng ký



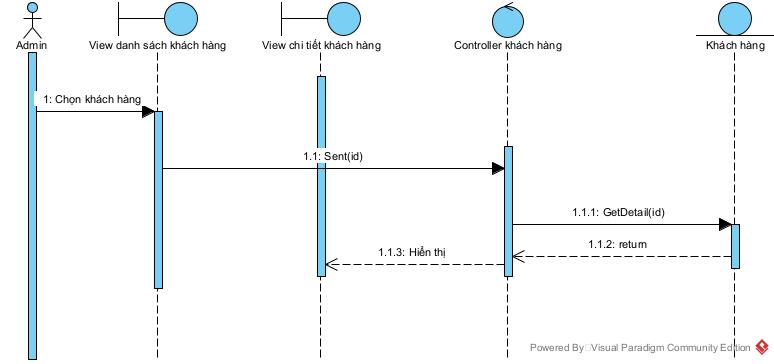
#### 3.3.3 Thêm khách hàng



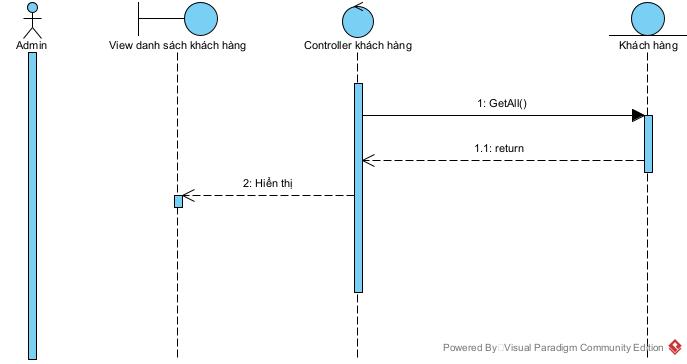
#### 3.3.4 Sửa khách hàng

#### 3.3.5 Xóa khách hàng

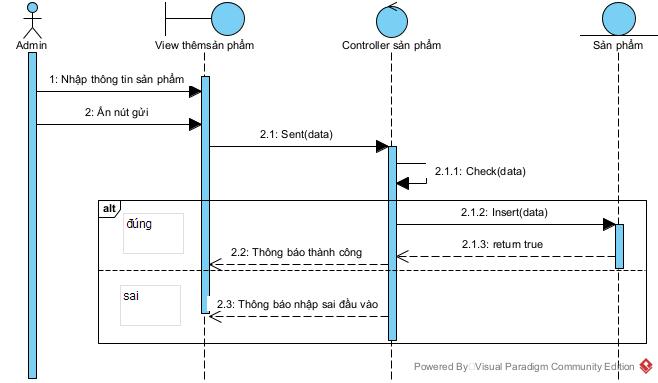
#### 3.3.6 Xem chi tiết khách hàng



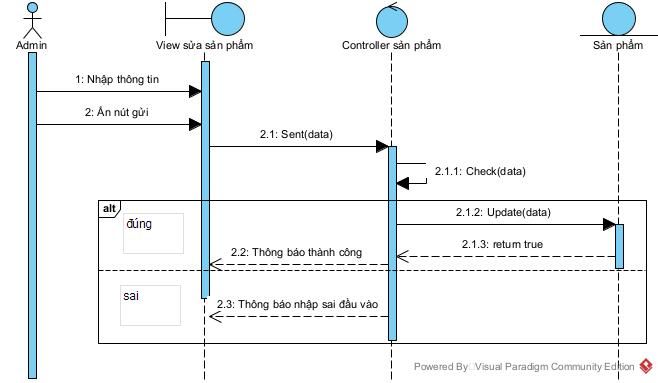
#### 3.3.7 Xem danh sách khách hàng



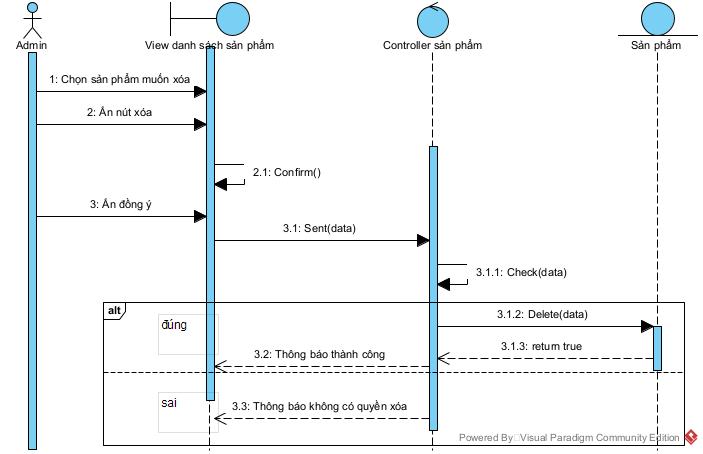
#### 3.3.8 Thêm sản phẩm



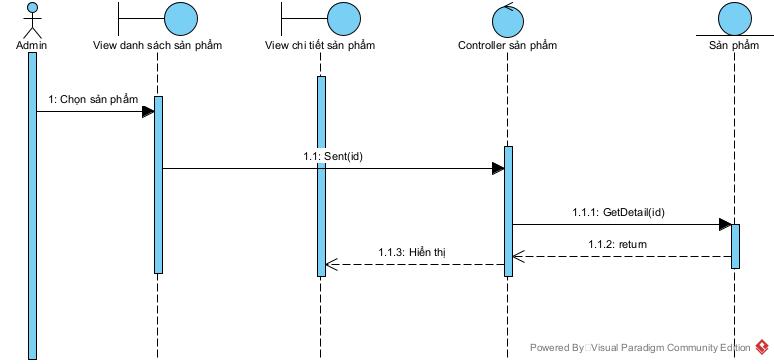
#### 3.3.9 Sửa sản phẩm



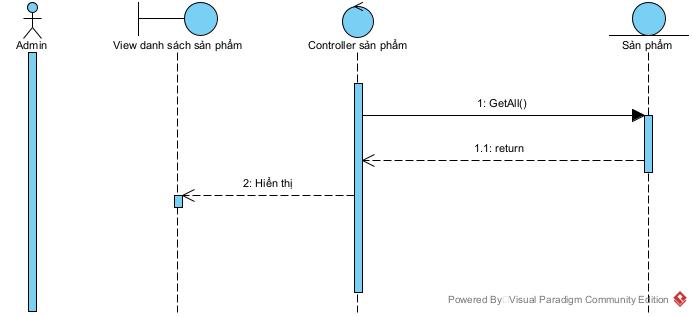
#### 3.3.10 Xóa sản phẩm



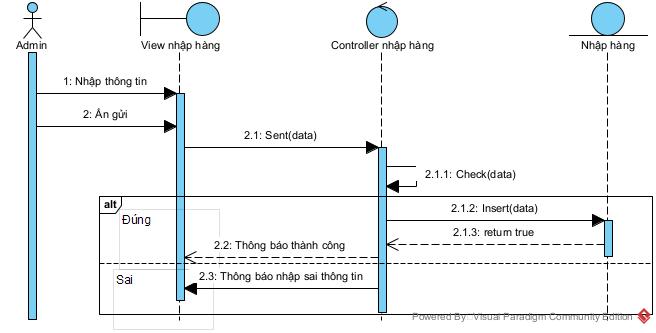
#### 3.3.11 Xem chi tiết sản phẩm



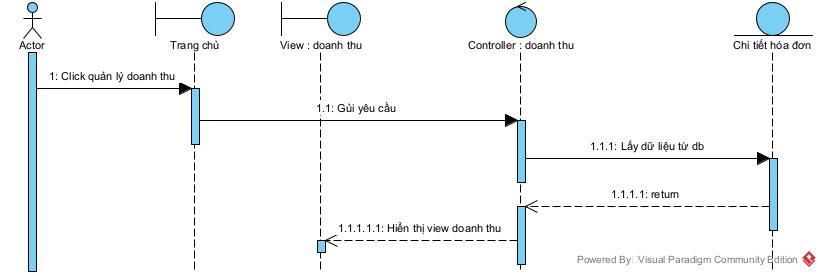
#### 3.3.12 Xem danh sách sản phẩm



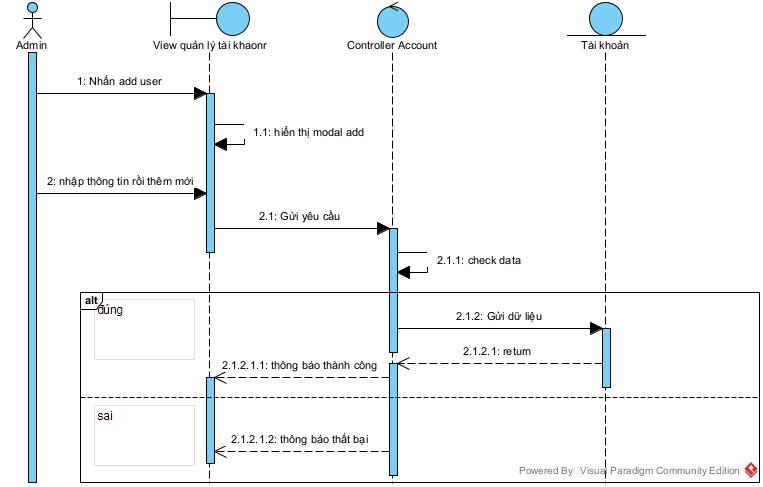
#### 3.3.13 Nhập hàng



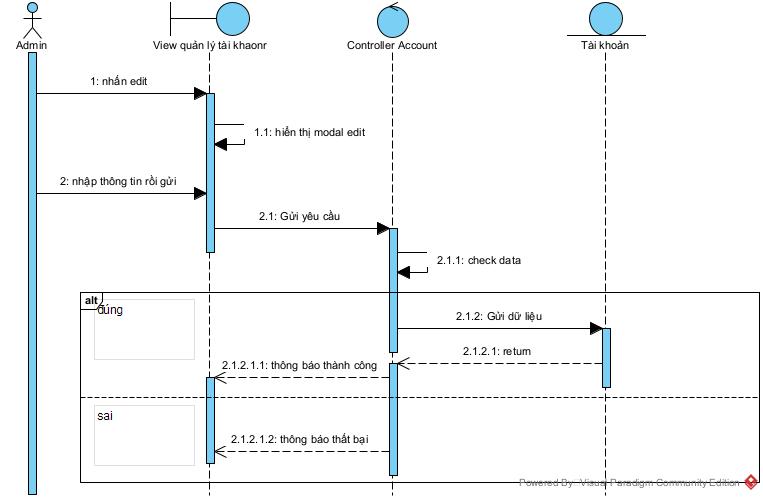
#### 3.3.13 Hiển thị doanh thu



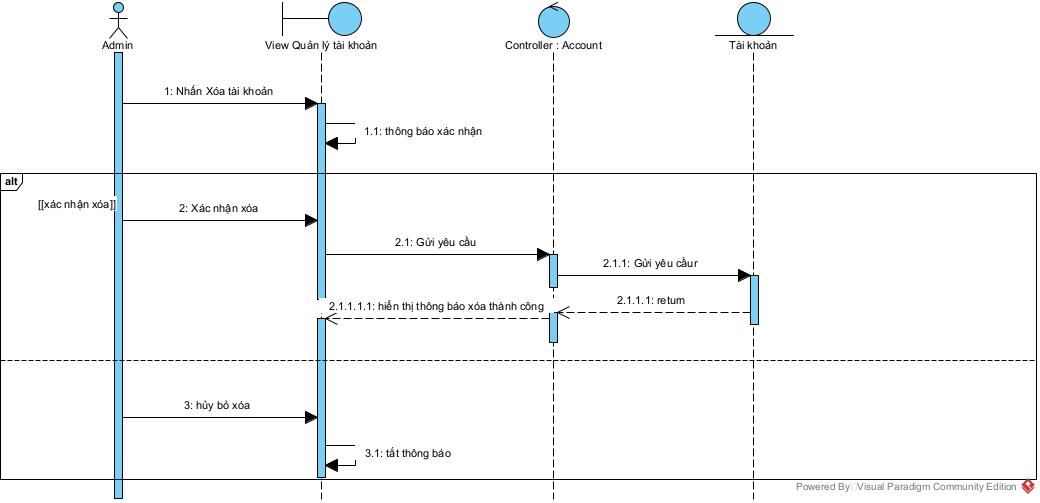
#### 3.3.14 Thêm người dùng



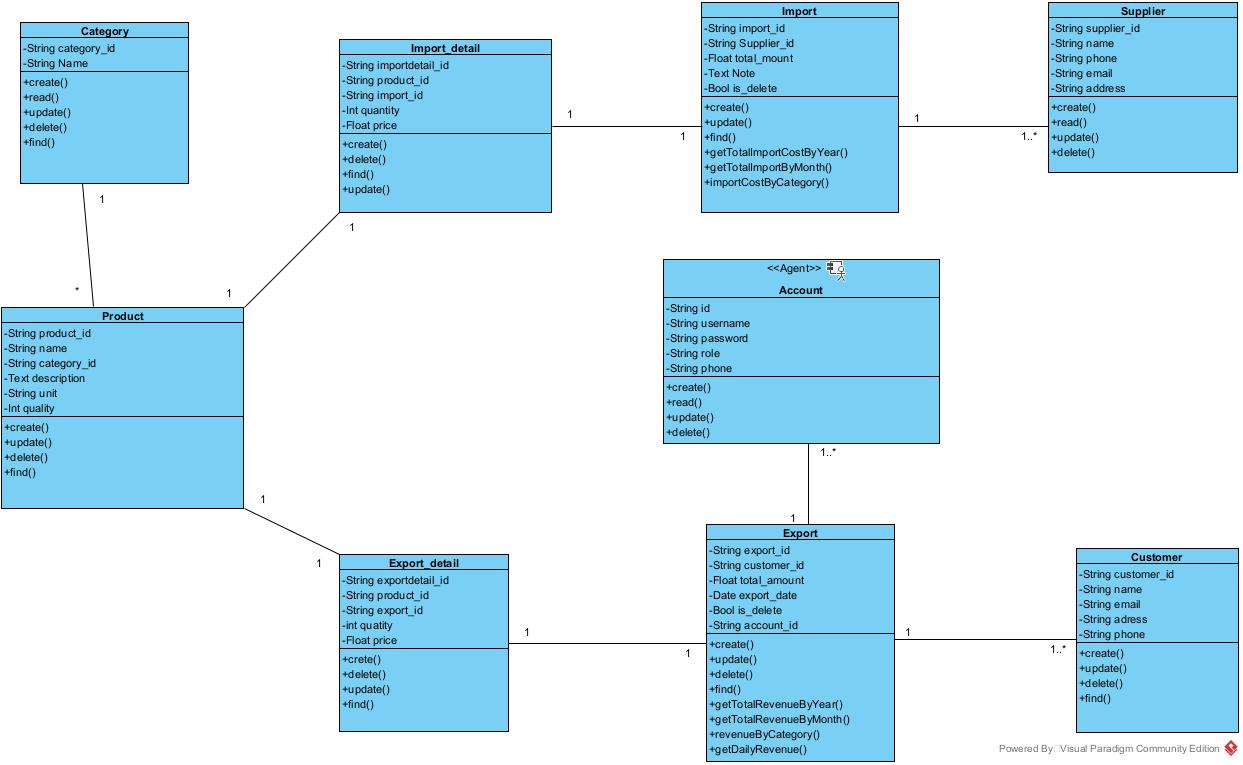
#### 3.3.15 Sửa thông tin người dùng



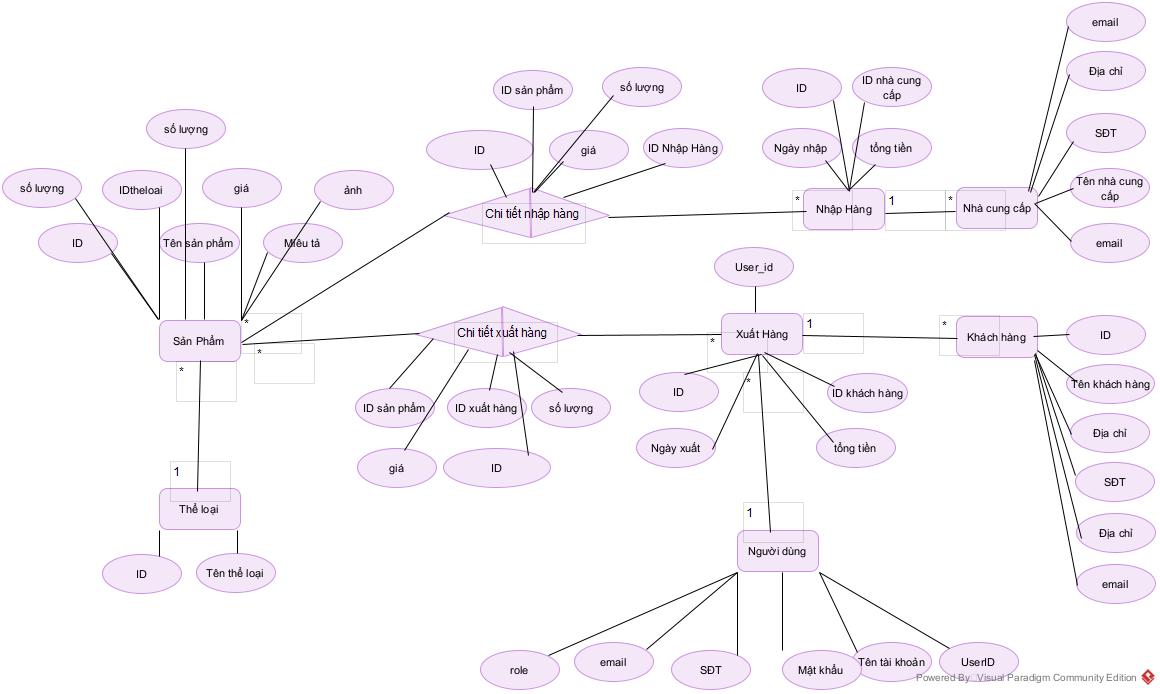
#### 3.3.14 Xóa người dùng



## 3.4 Class Diagram

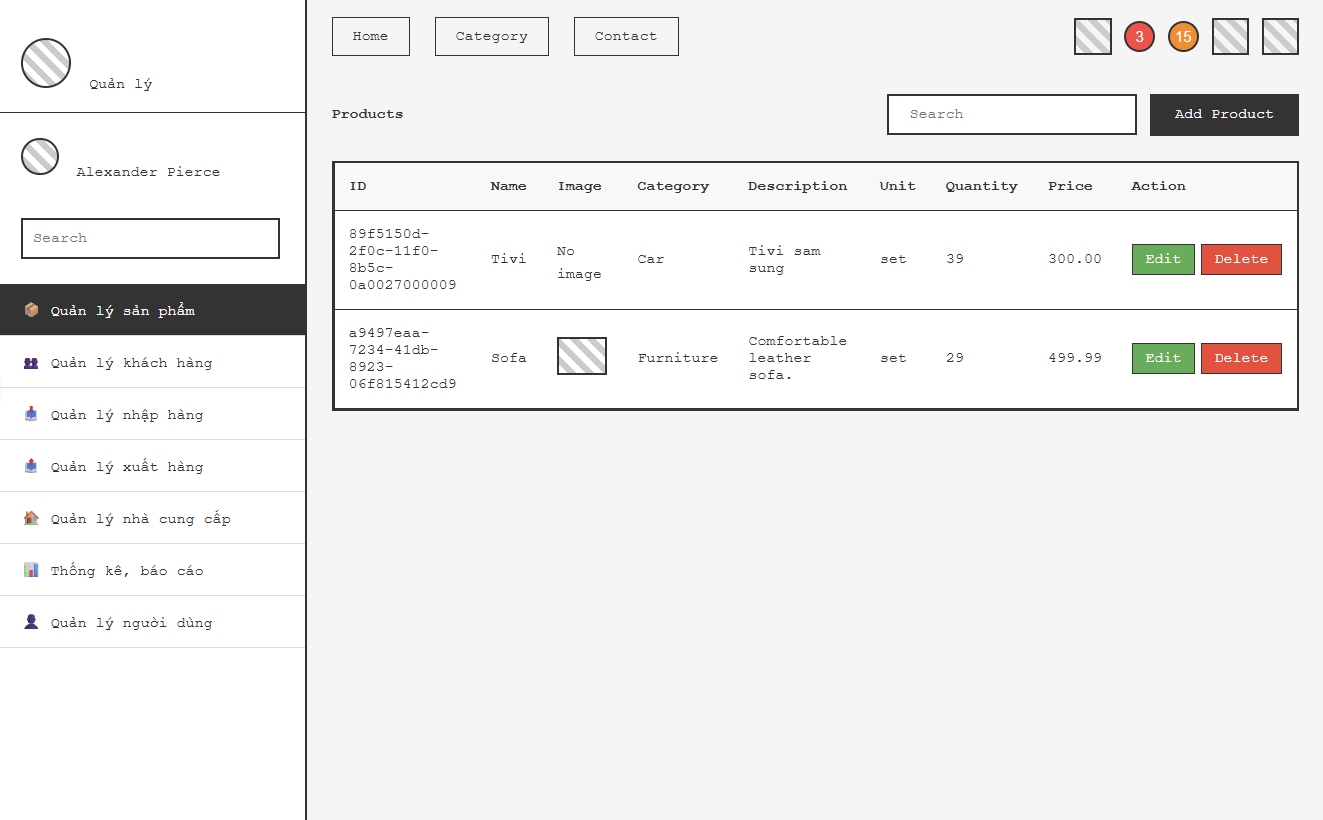
****

## 3.5 Sơ đồ quan hệ ERD

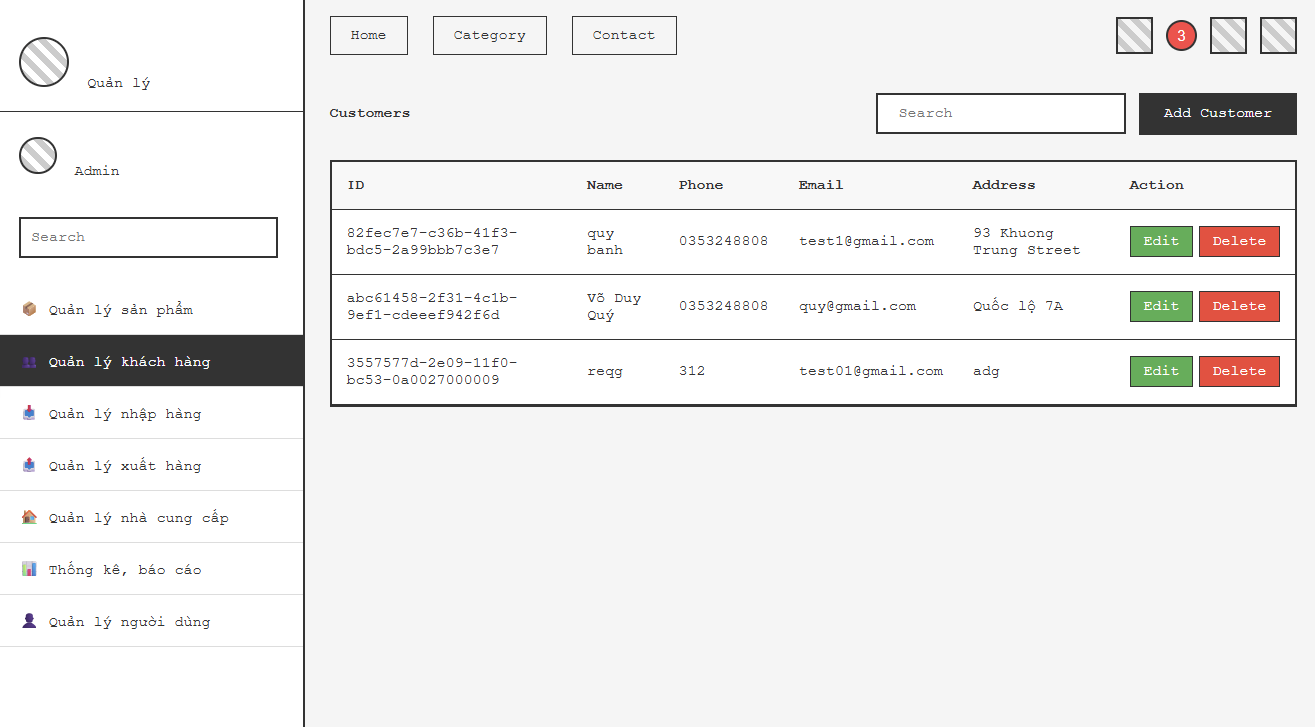
****

## 3.6 Thiết kế giao diện người dùng

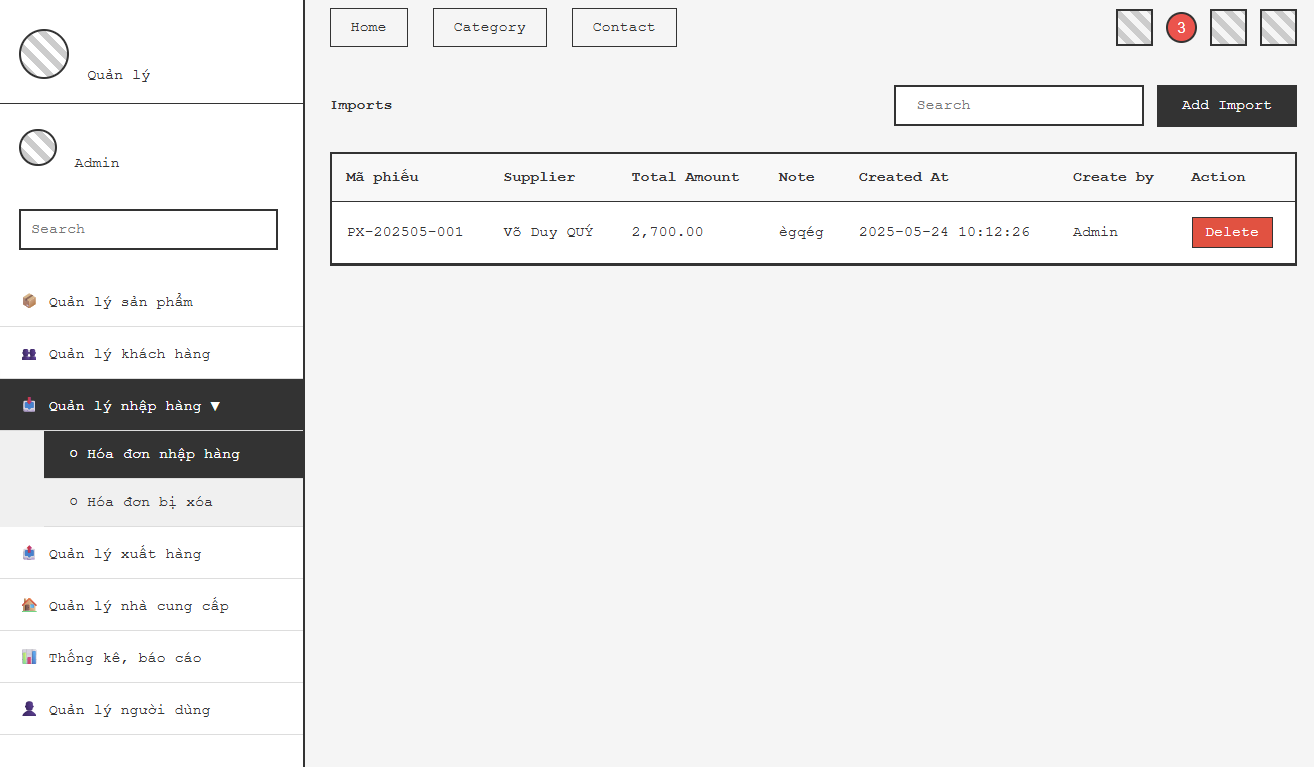
### 3.6.1 Giao diện sản phẩm



### 3.6.2 Giao diện quản lý khách hàng

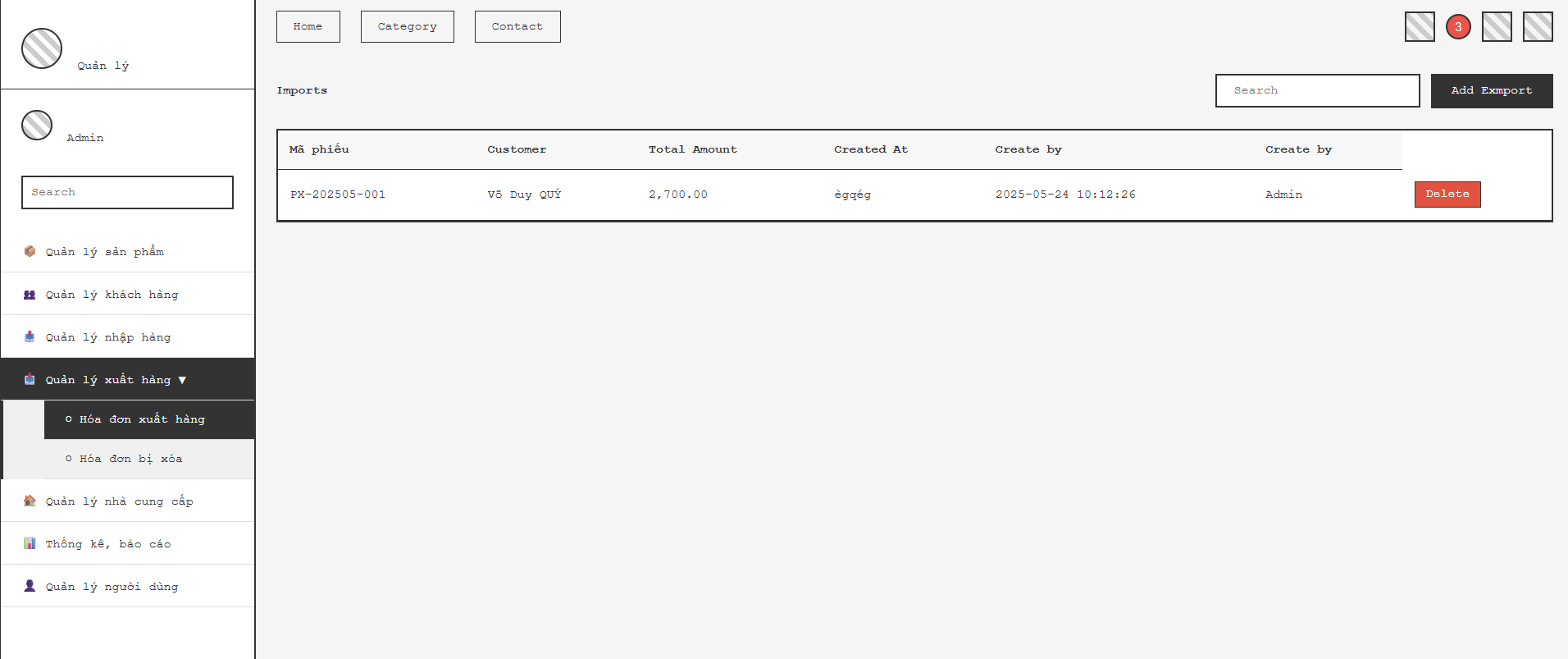


### 3.6.3 Giao diện quản lý đơn hàng nhập



### 3.6.4 Giao diện quản lý đơn hàng đã xóa

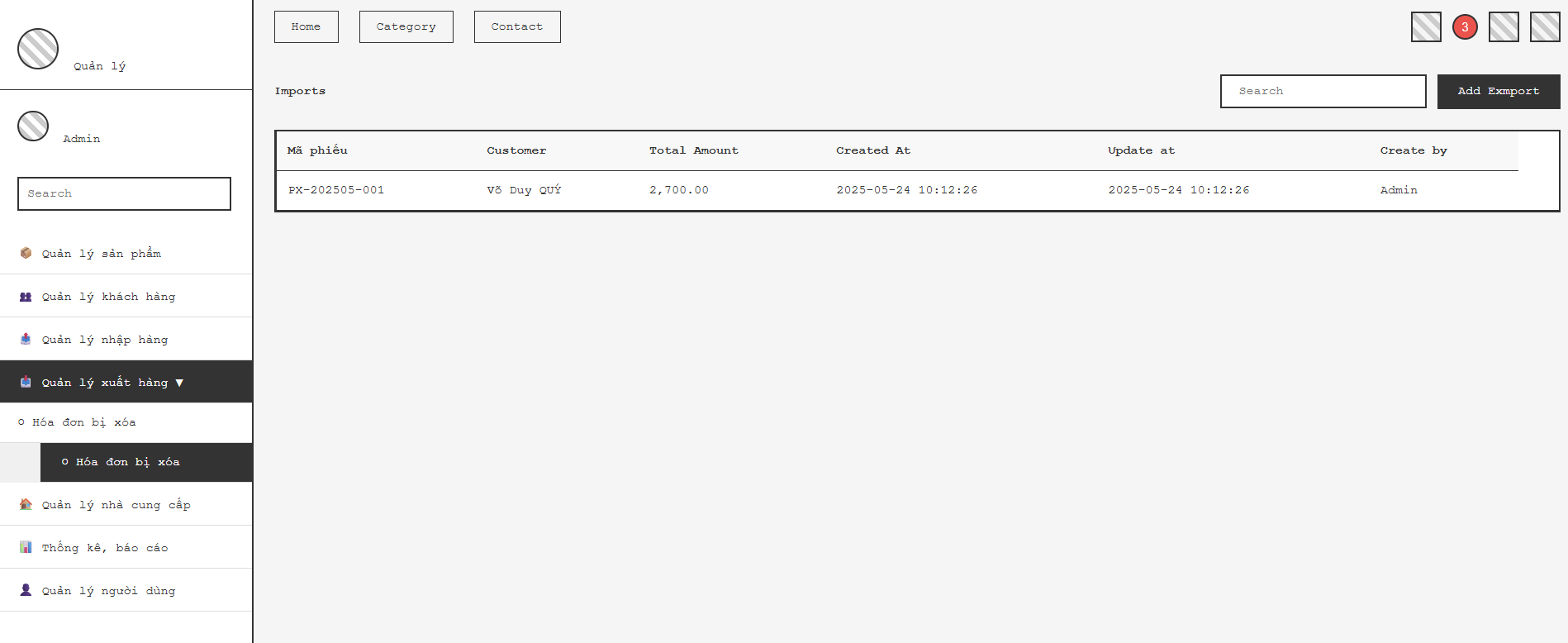
### 3.6.5 Giao diện quản lý xuất hàng



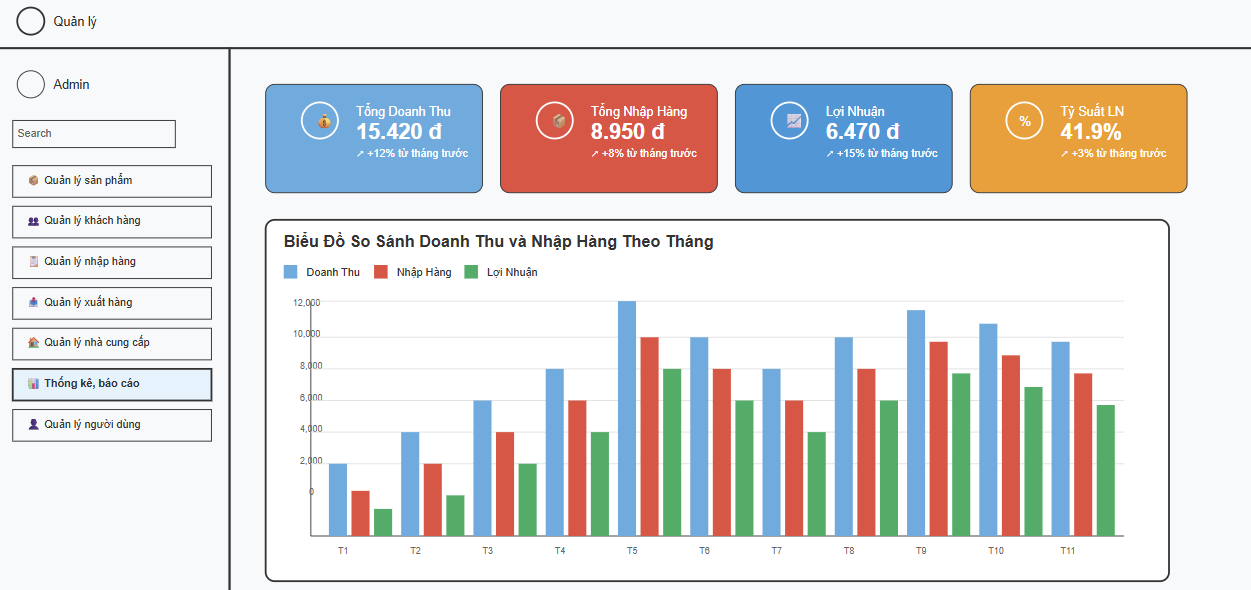
### 

### 

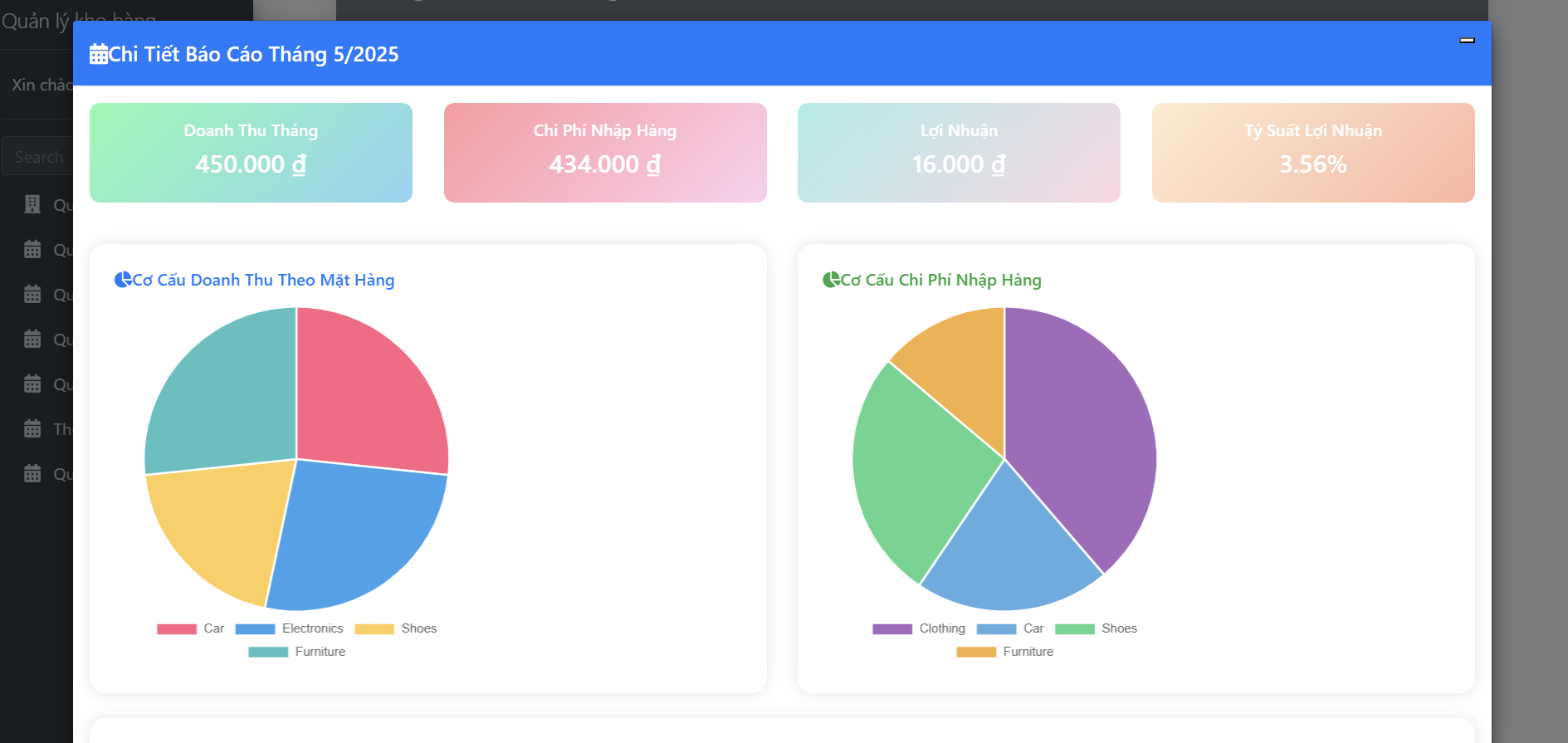
### 3.6.6 Giao diện hóa đơn đã xóa



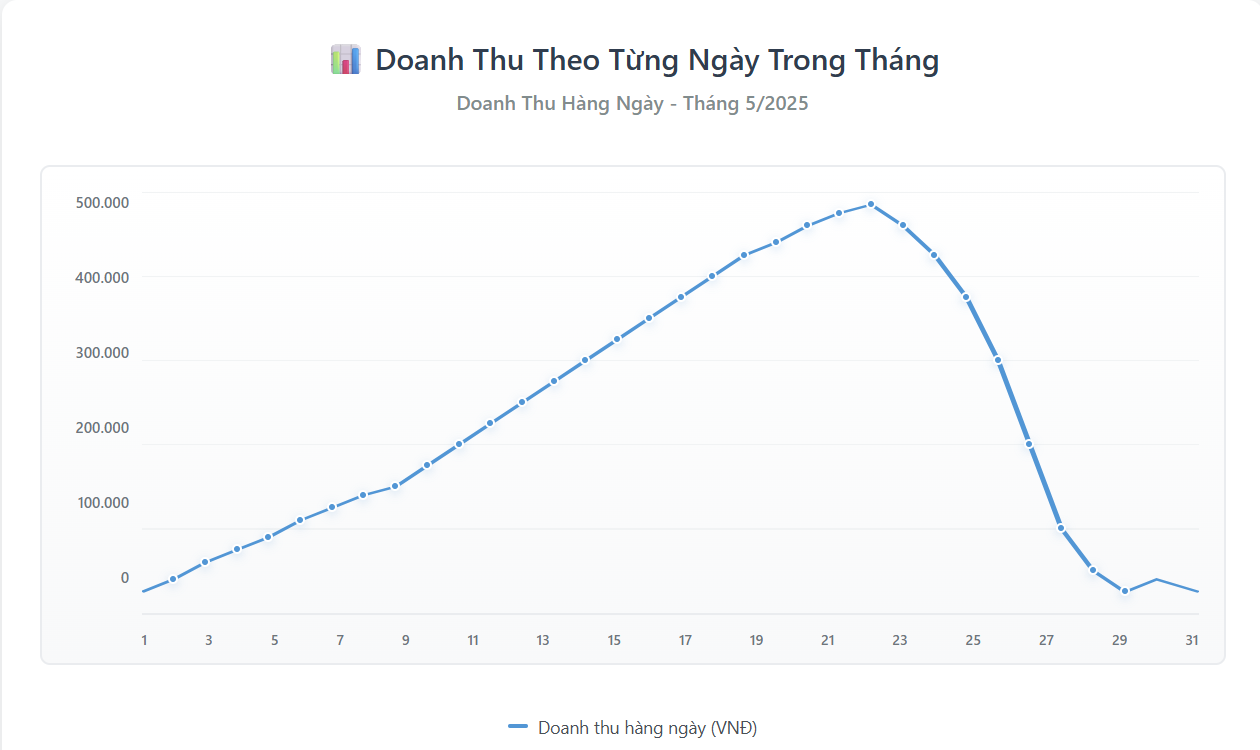
### 3.6.7 Giao diện quản lý doanh thu



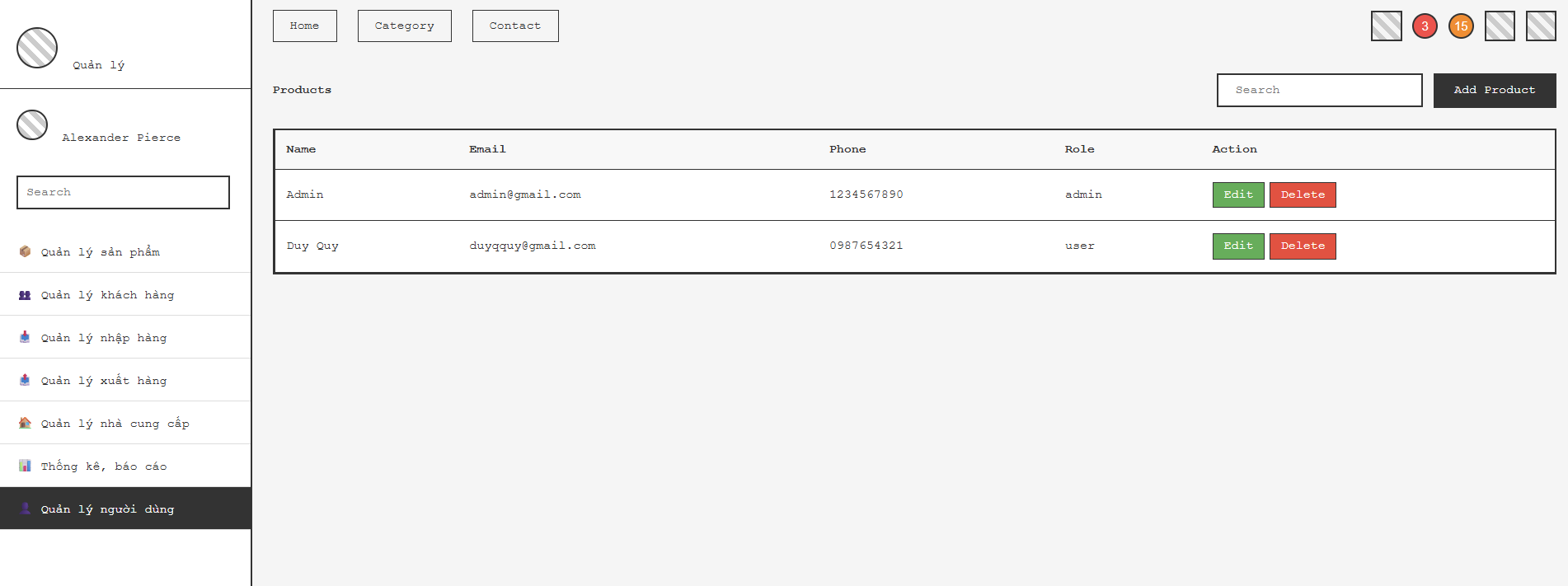
### 3.6.8 Giao diện chi tiết doanh thu theo sản phẩm



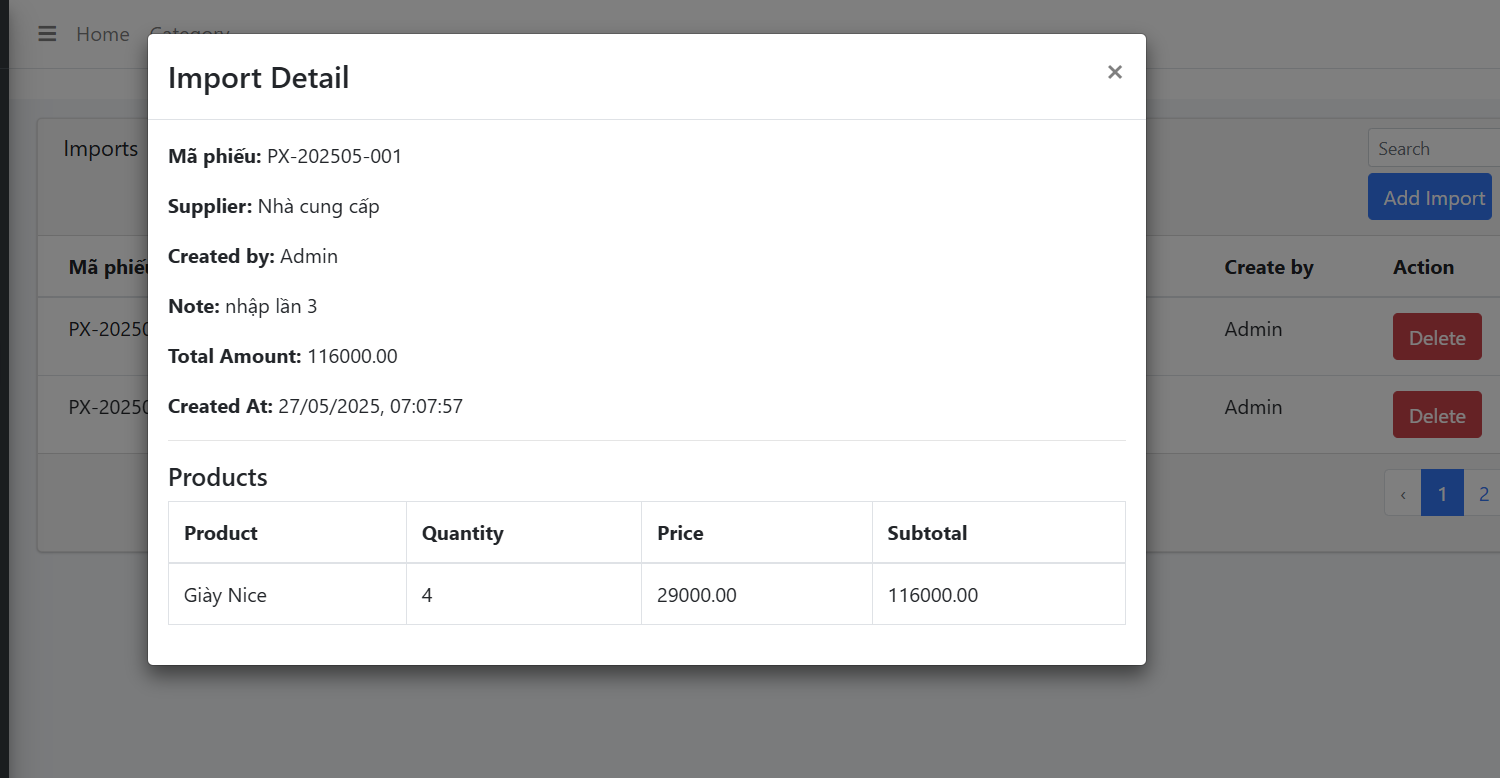
### 3.6.9 Giao diện chi tiết doanh thu theo ngày



### 3.6.10 Giao diện quản lý người dùng



### 3.6.11 Giao diện Modal chi tết đơn hàng



# **4. Cài đặt và triển khai hệ thống**

## 4.1 Môi trường trường triển khai

Hệ thống được xây dựng và triển khai trên môi trường như sau:

* Hệ điều hành: Windows 11
* Trình soạn thảo mã nguồn: Visual Studio Code
* Ngôn ngữ lập trình: PHP 8.x
* Framework: Laravel 10.x
* Hệ quản trị cơ sở dữ liệu: MySQL 8.x
* Trình duyệt hỗ trợ: Chrome, Firefox, Edge
* Công cụ hỗ trợ: Composer, Laravel Artisan CLI, Git

## 4.2 Yêu cầu hệ thống

Để cài đặt và chạy hệ thống, máy tính cần đảm bảo các yêu cầu sau:

* PHP >= 8.1
* Composer (công cụ quản lý thư viện PHP)
* Cài đặt Laravel CLI (laravel installer)
* Phpmyadmin để quản lý cơ sở dữ liệu

## 4.3 Các bước cài đặt chương trình

Các bước tiến hành cài đặt hệ thống Laravel như sau:

**Bước 1**: Cài đặt môi trường Laravel

* Tải và cài đặt Composer từ trang<https://getcomposer.org>
* Cài đặt Laravel bằng Composer

**Bước 2**: Tạo và cấu hình cơ sở dữ liệu

* Tạo CSDL mới trong MySQL (ví dụ: quanlysanpham)
* Mở file .env và cập nhật thông tin kết nối:

DB\_CONNECTION=mysql

DB\_HOST=127.0.0.1

DB\_PORT=3306

DB\_DATABASE=dapttk

DB\_USERNAME=root

DB\_PASSWORD=

**Bước 3**: Chạy migration và seed dữ liệu

**Bước 4**: Chạy ứng dụng Laravel

# **Chương 5. Kết quả và đánh giá**

## 5.1. Kết quả thử nghiệm hệ thống

Sau quá trình xây dựng và triển khai hệ thống, nhóm đã tiến hành kiểm thử các chức năng chính để đảm bảo hệ thống hoạt động đúng với yêu cầu đề ra. Việc thử nghiệm được thực hiện trên môi trường localhost với trình duyệt Google Chrome và cơ sở dữ liệu MySQL. Dưới đây là các kết quả thử nghiệm cụ thể:

* **Quản lý sản phẩm** Người dùng có thể thực hiện các chức năng thêm mới, sửa, xóa và xem danh sách sản phẩm. Hệ thống hỗ trợ hiển thị ảnh sản phẩm, mô tả, đơn vị tính, số lượng và giá bán. Giao diện trực quan, dễ sử dụng và hỗ trợ tìm kiếm sản phẩm theo tên.
* **Quản lý danh mục sản phẩm (Category)** Cho phép thêm các danh mục như Điện tử, Nội thất, Thực phẩm,... Khi thêm sản phẩm, người dùng có thể chọn danh mục tương ứng. Mỗi danh mục có thể liên kết với nhiều sản phẩm, đảm bảo quan hệ 1-nhiều hoạt động đúng.
* **Quản lý khách hàng** Giao diện cho phép quản lý thông tin khách hàng bao gồm tên, số điện thoại, email và địa chỉ. Có thể thực hiện chỉnh sửa hoặc xóa khách hàng dễ dàng.
* **Quản lý nhà cung cấp** Hệ thống cung cấp chức năng quản lý nhà cung cấp bao gồm thông tin liên hệ, tên công ty và mã số thuế. Việc liên kết sản phẩm với nhà cung cấp được triển khai hợp lý.
* **Quản lý nhập hàng, xuất hàng** Tính năng này giúp theo dõi lịch sử nhập – xuất sản phẩm. Khi nhập hàng, số lượng sản phẩm được cập nhật tăng lên. Khi xuất hàng, số lượng giảm tương ứng, đảm bảo tính nhất quán dữ liệu.
* **Thống kê và báo cáo** Hệ thống cung cấp các biểu đồ và bảng thống kê theo thời gian (ngày, tháng, năm) về số lượng sản phẩm đã nhập, đã bán và lợi nhuận thu được. Tính năng này giúp người quản lý có cái nhìn tổng quan về hiệu suất hoạt động của hệ thống.
* **Tính năng bảo mật và phân quyền** Hệ thống có chức năng đăng nhập, đăng ký, và phân quyền người dùng. Chỉ người dùng có vai trò phù hợp mới có thể truy cập các chức năng nhạy cảm như quản lý người dùng, thống kê hoặc xóa dữ liệu.

**Kết luận:** Hầu hết các chức năng đều hoạt động ổn định, đúng yêu cầu và mang lại trải nghiệm người dùng tốt. Một số lỗi nhỏ phát sinh trong quá trình kiểm thử đã được khắc phục kịp thời.

## 5.2. Đánh giá hiệu quả hệ thống

Qua quá trình xây dựng và thử nghiệm, có thể đưa ra một số đánh giá tổng quan về hệ thống như sau:

* **Về tính đầy đủ chức năng:** Hệ thống đã đáp ứng gần như đầy đủ các chức năng được đặt ra trong giai đoạn phân tích yêu cầu. Việc quản lý dữ liệu (sản phẩm, người dùng, khách hàng, nhà cung cấp...) được thực hiện đầy đủ theo mô hình CRUD.
* **Về hiệu suất:** Hệ thống hoạt động ổn định với thời gian phản hồi nhanh trên môi trường máy cá nhân. Các thao tác truy vấn, thêm sửa dữ liệu không gặp tình trạng treo, giật lag.
* **Về khả năng mở rộng:** Do được xây dựng theo mô hình MVC và sử dụng framework Laravel, hệ thống có khả năng mở rộng cao. Việc bổ sung các chức năng mới như in hóa đơn, gửi email, tích hợp API giao vận có thể thực hiện dễ dàng trong tương lai.
* **Về bảo mật:** Laravel hỗ trợ sẵn các cơ chế bảo mật như CSRF token, xác thực người dùng, mã hóa mật khẩu,... giúp hệ thống an toàn trước các lỗ hổng thông thường như SQL Injection, XSS. Tuy nhiên, hệ thống vẫn cần được kiểm thử bảo mật kỹ hơn nếu triển khai thực tế.
* **Về giao diện người dùng:** Giao diện thân thiện, dễ thao tác, đặc biệt phù hợp với người quản trị. Tuy nhiên, một số phần có thể cải tiến thêm về mặt thẩm mỹ và bố cục để thân thiện hơn với người dùng phổ thông.
* **Về khó khăn:** Trong quá trình xây dựng, nhóm gặp một số khó khăn như lỗi khi migrate database, lỗi hiển thị ảnh, lỗi liên quan đến phân quyền người dùng,... Tuy nhiên các vấn đề này đã được giải quyết nhờ tìm hiểu tài liệu chính thức của Laravel và sự hỗ trợ từ cộng đồng.

**Tổng kết:** Hệ thống được triển khai đạt hiệu quả cao, đáp ứng được yêu cầu đề ra trong phạm vi đồ án. Trong tương lai, nhóm có thể nâng cấp hệ thống theo hướng hỗ trợ nhiều ngôn ngữ, tích hợp hệ thống thanh toán, hoặc triển khai trên nền tảng cloud để ứng dụng thực tiễn.

# **6. Tài liệu tham khảo**

* ChatGPT – Công cụ hỗ trợ trả lời câu hỏi và tạo nội dung thông minh. OpenAI.<https://chat.openai.com>
* W3Schools – Trang học lập trình HTML, CSS, JavaScript, PHP cơ bản.<https://www.w3schools.com>
* Laravel Official Documentation – Tài liệu chính thức của Laravel Framework.<https://laravel.com/docs>
* Stack Overflow – Diễn đàn hỏi đáp lập trình viên.<https://stackoverflow.com>
* PHP Manual – Tài liệu chính thức cho ngôn ngữ lập trình PHP.<https://www.php.net/manual/en/>
* Bootstrap Documentation – Tài liệu giao diện và responsive web.<https://getbootstrap.com>
* YouTube – Các video hướng dẫn Laravel và lập trình web thực tế.
* GitHub – Tham khảo mã nguồn mở và tài liệu từ cộng đồng.<https://github.com>
* Font Awesome – Thư viện icon sử dụng trong giao diện web.<https://fontawesome.com>
* FakerPHP – Thư viện tạo dữ liệu giả lập cho Laravel. https://fakerphp.github.io