**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**======\*\*\*======**

****

**Báo cáo thực tập tốt nghiệp**

|  |  |
| --- | --- |
| Giáo viên: | Ths. Phạm Thị Kim Phượng |
| Nhóm - Lớp: | CNTT2-K11 |
| Họ tên: | Trần Quang Hoàng |
|  |  |
|  |  |
|  |  |
| Hệ thống: | **Website bán điện thoại hungmobile.vn** |

Hà nội, Năm 2020

LỜI MỞ ĐẦU

Hiện nay, các công nghệ tiên tiến phát triển ngày càng mạnh mẽ và được ứng dụng ngàycàng nhiều vào các lĩnh vực kinh tế, sản xuất cũng như đời sống thường nhật của conngười. Một điểm tiêu biểu trong việc phát triển các công nghệ đó phải kể đến việc ứng dụng công nghệ thông tin vào hầu khắp các hoạt động. Nhờ đó, các công việc được thựchiện nhanh, chính xác và đạt kết quả cao hơn rất nhiều.

Khi xã hội ngày càng phát triển, mức sống của người dân được nâng cao thu nhập kinh tế ngày càng được cải thiện thì chiếc điện thoại di động không còn trở nên xa lạ với mọingười nữa mà ngược lại nó là một vật dụng không thể thiếu đối với người dân hiện nay.Hầu hết mỗi người đều trang bị cho mình một chiếc điện thoại phù hợp với nhu cầu và túi tiền của mình. Tuy nhiên, với cuộc sống ngày càng bận rộn như hiện nay thì việcmuốn mua một chiếc điện thoại mình ưa thích thì người tiêu dùng phải đến tận cửa hàngđể chọn lựa vì thế sẽ mất khá nhiều thời gian và công sức.

Cùng với các lý do nêu trên, qua tìm hiểu nhóm đồ án được biết việc ứng dụng bán điệnthoại di động trực tuyến sẽ giúp cho khách hàng giảm bớt được thời gian và công sứcphải đến tận cửa hàng để mua. Muốn lựa chọn cho mình một chiếc điện thoại ưng ý phùhợp với túi tiền thì khách hàng chỉ cần ngồi bên chiếc máy tính có nối mạng internet làcó thế mua được mặt hàng điện thoại mình cần.Do đó nhóm đề tài chọn thực hiện đề tài *“Xây dựng Website bán điện thoại di động trực tuyến cho cửa hang Hùng Mobile ”*. Phần mềm được xây dựng với định hướng giúp việc mua sắm điện thoại của khách hàngdễ dàng và nhanh gọn hơn, không mất thời gian và công sức. Qua quá trình khảo sát thực tế, tìm hiểu và phân tích thiết kế nhóm đề tài đưa ra bản báocáo gồm các phần sau để giúp người đọc hiểu rõ về chức năng của Website.

MỤC LỤC

[Chương 1 Tổng quan về Java Spring và JPA 5](#_Toc33697491)

[1.1 Giới thiệu Spring Framework 5](#_Toc33697492)

[1.1.1 Spring là gì? 5](#_Toc33697493)

[1.1.2 Kiến trúc của Spring 6](#_Toc33697494)

[1.1.3 Các lợi ích của Spring Framework 7](#_Toc33697495)

[1.1.4 Một số dự án khác của Spring. 8](#_Toc33697496)

[1.2 Tổng quan về JPA (Java Persistence API) 9](#_Toc33697497)

[1.2.1 JPA là gì? 9](#_Toc33697498)

[1.2.2 ORM là gì? 10](#_Toc33697499)

[1.2.3 Tại sao nên dùng JPA? 10](#_Toc33697500)

[Chương 2 Khảo sát hệ thống 11](#_Toc33697501)

[2.1 Khảo sát hệ thống 11](#_Toc33697502)

[2.2 Hoạt động của hệ thống 11](#_Toc33697503)

[2.3 Các yêu cầu chức năng 11](#_Toc33697504)

[2.4 Các yêu cầu phi chức năng 12](#_Toc33697505)

[Chương 3 Phân tích hệ thống 13](#_Toc33697506)

[3.1 Các use case chính 13](#_Toc33697507)

[3.2 Các use case thứ cấp 13](#_Toc33697508)

[3.3 Phân tích use case 14](#_Toc33697509)

[3.3.1 Phân tích use case “Xem hàng” 14](#_Toc33697510)

[3.3.2 Phân tích use case “Tìm kiếm” 15](#_Toc33697511)

[3.3.3 Phân tích use case “Xem tin tức” 16](#_Toc33697512)

[3.3.4 Phân tích use case “Tạo giỏ hàng” 17](#_Toc33697513)

[3.3.5 Phân tích use case “Hỏi thông tin” 18](#_Toc33697514)

[3.3.6 Phân tích use case “Mua hàng” 19](#_Toc33697515)

[3.3.7 Phân tích use case “Nhận đơn hàng” 20](#_Toc33697516)

[3.3.8 Phân tích use case “Trả lời thắc mắc” 21](#_Toc33697517)

[Chương 4 Thiết kế cơ sở dữ liệu 22](#_Toc33697518)

[4.1 Mô tả hệ thống 22](#_Toc33697519)

[4.2 Xác định các lớp thực thể 22](#_Toc33697520)

[Chương 5 Thiết kế giao diện 26](#_Toc33697521)

[5.1 Thiết kế giao diện use case “Xem hàng” 26](#_Toc33697522)

[5.1.1 Hình dùng màn hình 26](#_Toc33697523)

[5.2 Thiết kế giao diện use case “Tạo giỏ hàng” 27](#_Toc33697524)

[5.2.1 Hình dung màn hình 27](#_Toc33697525)

[5.3 Thiết kế giao diện use case “mua hàng” 27](#_Toc33697526)

[5.3.1 Hình dung màn hình 27](#_Toc33697527)

[5.4 Thiết kế giao diện use case “Xem tình trạng đơn hàng” 28](#_Toc33697528)

[5.5 Thiết kế giao diện use case “Tìm kiếm” 28](#_Toc33697529)

[5.5.1 Hình dung màn hình 28](#_Toc33697530)

[Chương 6 Cài đặt chương trình 29](#_Toc33697531)

[6.1 Cài đặt MySQL 29](#_Toc33697532)

[6.1.1 Tải MySQL 29](#_Toc33697533)

[6.1.2 Cài đặt 29](#_Toc33697534)

[6.2 Cài đặt chương trình 40](#_Toc33697535)

[6.2.1 Giao diện phần người dùng 40](#_Toc33697536)

[6.2.2 Giao diện phần quản lý 44](#_Toc33697537)

[Tài liệu tham khảo 51](#_Toc33697538)

# Tổng quan về Java Spring và JPA

## Giới thiệu Spring Framework

### Spring là gì?

Spring là một Framework phát triển các ứng dụng Java được sử dụng bởi hàng triệu lập trình viên. Nó giúp tạo các ứng dụng có hiệu năng cao, dễ kiểm thử, sử dụng lại code…

Spring nhẹ và trong suốt (nhẹ: kích thước nhỏ, version cơ bản chỉ khoảng 2MB; trong suốt: hoạt động một cách trong suốt với lập trình viên)

Spring là một phần mềm mã nguồn mở, được phát triển, chia sẻ và có một cộng đồng người dùng rất lớn.

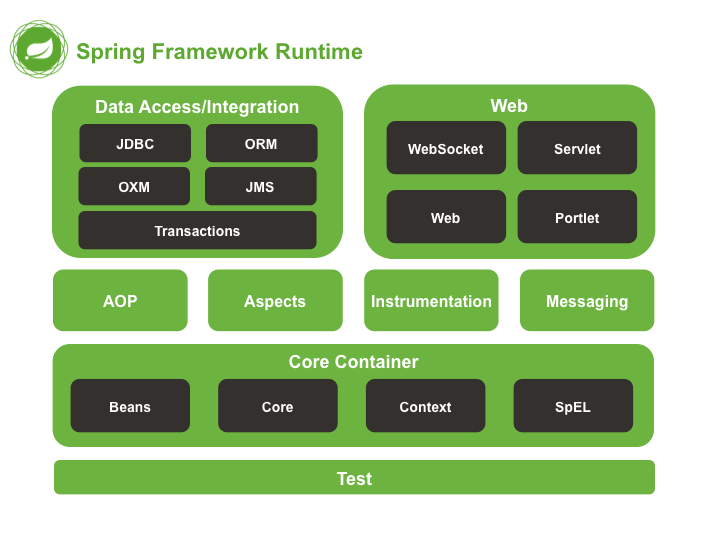
Spring Framework được xây dựng dựa trên 2 nguyên tắc design chính là: Dependency Injection và Aspect Oriented Programming.

Những tính năng cốt lõi của Spring có thể được sử dụng để phát triển Java Desktop, ứng dụng mobile, Java Web. Mục tiêu chính của Spring là giúp phát triển các ứng dụng J2EE một cách dễ dàng hơn dựa trên mô hình sử dụng POJO (Plain Old Java Object).

Ở đồ án thực tập tốt nghiệp này, chúng ta sử dụng Spring Framework để một ứng dụng web.

### Kiến trúc của Spring

Spring được chia làm nhiều module khác nhau, tùy theo mục đích phát triển ứng dụng mà ta dùng 1 trong các module đó.



*Hình 1.1: kiến trúc tổng thể của Spring Framework.*

* **Test**

Tầng này cung cấp khả năng hỗ trợ kiểm thử với JUnit và TestNG.

* **Spring Core Container**

Bao gồm các module spring core, beans, context và expression languate (EL)

* Spring core, bean cung cấp tính năng IOC và Dependency Injection.
* Spring Context hỗ trợ đa ngôn ngữ (internationalization), các tính năng Java EE như EJB, JMX.
* Expression Language được mở rộng từ Expresion Language trong JSP. Nó cung cấp hỗ trợ việc setting/getting giá trị, các method cải tiến cho phép truy cập collections, index, các toán tử logic…
* **AOP, Aspects and Instrumentation**

Những module này hỗ trợ cài đặt lập trình hướng khía cạnh (Aspect Oriented Programming), hỗ trợ tích hợp với AspectJ.

* **Data Access / Integration**

Nhóm này bao gồm JDBC, ORM, OXM, JMS và module Transaction. Những module này cung cấp khả năng giao tiếp với database

* **Web**

Hay còn gọi là Spring MVC Nhóm này gồm Web, Web-Servlet… hỗ trợ việc tạo ứng dụng web.

### Các lợi ích của Spring Framework

* Spring cho phép lập trình viên sử dụng POJOs. Việc sử dụng POJOs giúp bạn không phải làm việc với EJB, ứng dụng, các luồng chạy, cấu hình… đơn giản hơn rất nhiều.
* Spring được tổ chức theo kiểu mô đun. Số lượng các gói và các lớp khá nhiều, nhưng bạn chỉ cần quan tâm đến những gì bạn cần và không cần quan tâm đến phần còn lại.
* Spring hỗ trợ sử dụng khá nhiều công nghệ như ORM Framework, các logging framework, JEE, các thư viện tạo lịch trình (Quartz và JDK timer)…
* Module Web của Spring được thiết kế theo mô hình MVC nên nó cung cấp đầy đủ các tính năng giúp thay thế các web framework khác như Struts.

### Một số dự án khác của Spring.

Cũng dựa trên các nguyên tắc thiết kế cơ bản của spring core. Spring còn phát triển nhiều project con như:

* **Spring MVC**

Spring MVC được thiết kế dành cho việc xây dựng các ứng dụng nền tảng web.

* **Spring Security**

Cung cấp các cơ chế xác thực (authentication) và phân quyền (authorization) cho ứng dụng của bạn.

* **Spring Boot**

Spring Boot là một framework giúp chúng ta phát triển cũng như chạy ứng dụng một cách nhanh chóng.

* **Spring Batch**

Dự án này giúp chúng ta dễ dàng tạo các lịch trình (scheduling) và tiến trình (processing) cho các công việc xử lý theo mẻ (batch job).

* **Spring Social**

Dự án này sẽ kết nối ứng dụng của bạn với các API bên thứ ba của Facebook, Twitter, Linkedin … (ví dụ đăng nhập bằng facebook, google+ …)

**Spring IO, Spring Cloud, Spring Mobile, Spring for Android, Spring Session…**

## Tổng quan về JPA (Java Persistence API)

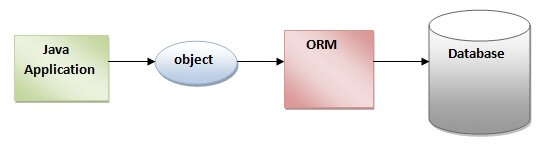
Bất kỳ một ứng dụng nào cũng đều cần phải thực hiện các tháo tác (CRUD) đến database bằng việc lưu trữ, truy vấn dữ liệu. Chúng ta cần phải viết nhiều code để thực hiện các việc này.

Với công nghệ JDBC trước đây, chúng ta phải thực hiện các việc: mở kết nối vào Database, tạo các Statement, ResultSet, … và sau cùng phải đóng tất cả các thứ đó lại. Dẫn đến code sẽ trở nên cồng kềnh, khó mở rộng và bảo trì. Chúng ta cũng cần phải làm việc với cả Java code và SQL. Thực tế, từng Database khác nhau thì câu SQL có một số phần khác nhau, nên đòi hỏi chúng ta phải nắm được sự khác biệt này để viết code cho phù hợp. Khi ứng dụng muốn chuyển từ database sang database khác (ví dụ từ MySQL sang Oracle) thì chắc chắn sẽ có một số phần của câu SQL cần phải đổi. Công việc sửa code lại đòi hỏi phải test lại ứng dụng. Điều này sẽ tốn thời gian, chi phí phát triển và cũng gặp không ít rủi ro. Để khắc phục nhược điểm này, đã có rất nhiều Framework ra đời với mục đích giúp xóa đi vấn để về tương thích giữa các Database, giúp tập trung vào phần xử lý nghiệp vụ.

Trong dự án này, tôi sẽ sử dụng JPA - một chuẩn đặc tả cho các ORM Framework để giải quyết vấn đề trên.

### JPA là gì?

JPA là viết tắt của Java Persistence API, nó là một đặc tả Java cho việc ánh xạ giữa các đối tượng Java với cơ sở dữ liệu quan hệ sử dụng công nghệ phổ biến là ORM (Object Relational Mapping).



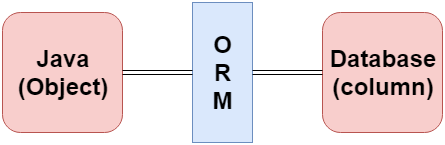
*Hình 1.2: Cách thức truy xuất dữ liệu của JPA.*

JPA hoạt động như một cầu nối giữa các table/ các mối quan hệ giữa các table trong database và các class/ mối quan hệ giữa các object. Ví dụ: table USER với các column (Id, username, password) sẽ tương ứng với class User.java với các field Id, username, password. Từ đó mỗi khi truy vấn table hay các column ta sẽ gọi trực tiếp các phương thức trên các class, các field của class mà không cần quan tâm tới việc đang dùng loại database nào, kiểu dữ liệu database ra sao, …

### ORM là gì?

ORM là viết tắt của Object Relational Mapping, là một công nghệ/ khái niệm/ quá trình chuyển đổi dữ liệu từ ngôn ngữ hướng đối tượng sang Database quan hệ và ngược lại. Ví dụ, trong Java nó được thực hiện với sự trợ giúp của Reflection và JDBC.

ORM có khả năng xử lý các thao tác của nhiều loại cơ sở dữ liệu khác nhau một cách dễ dàng mà không quan tâm đến loại database sử dụng (SQL Server, MySQL, PostgreSQL,…) hay loại thao tác sử dụng (INSERT, UPDATE, DELETE, SELECT, …).



*Hình 1.3: Chuyển đổi giữa Java Object và Database column sử dụng ORM.*

### Tại sao nên dùng JPA?

* Viết ít code hơn.
* Performance tốt.
* Độc lập về database.
* Không phải làm việc với SQL.
* Hỗ trợ cấu hình triển khai bằng annotation và xml.
* Có nhiều framework ORM miễn phí hỗ trợ có thể dùng để phát triển nhiều loại ứng dụng khác nhau.
* JPA là một đặc tả đã được chuẩn hóa và là một thành phần trong đặc tả EJB 3.
* Dễ dàng chuyển từ một ORM này sang một ORM khác. Ví dụ từ iBatis sang Hibernate.

# Khảo sát hệ thống

## Khảo sát hệ thống

- Phương pháp: Quan sát.

- Đối tượng: website chính thức của Hùng Mobile *(*[*https://hungmobile.vn/*](https://hungmobile.vn/)*)*

- Thông tin nhận được:

+) Các chức năng của hệ thống:

* Về phía khách hàng:
  + Xem sanh sách sác sản phẩm điện thoại, tìm kiếm các sản phẩm theo các tiêu chí như tên, hãng, giá tiền, sắp tang dần, giảm dần theo giá, số lượng đặt mua.
  + Xem chi tiết sản phẩm, đặt hang, chat với nhân viên hỗ trợ.
  + Xem tin tức về sản phẩm công nghệ.
* Về phía quản lí:
  + Đăng nhập tài khoản.
  + Phân quyền truy cập.
  + Quản lí nhập xuất kho.
  + Quản lí nhân viên.
  + Quản lí bán hàng, đơn hàng.
  + Quản lí giá cả.
  + Quản lí khuyễn mãi, tin tức.

## Hoạt động của hệ thống

* Hệ thống có 2 Actor chính là Khách Hàng và Nhân Viên cửa hàng
* Khi khách hàng sử dụng chức năng đặt hàng trên website. Nhân viên cửa hàng sẽ nhận được thông báo và xác nhận đơn hang bằng cách gọi điện cho khách hàng. Sau đấy gửi hàng qua các đơn vị chuyển phát đến tay khách hàng.
* Nhân viên quản lý sẽ quản lí nhập, xuất hàng, quản lí lịch sử đơn hàng.
* Trường hợp khách hang yêu cầu bảo hành, khách hàng lên website để nhập mã đơn hàng, nhân viên kĩ thuật sẽ xác nhận và nhận hoặc từ chối bảo hành.

## Các yêu cầu chức năng

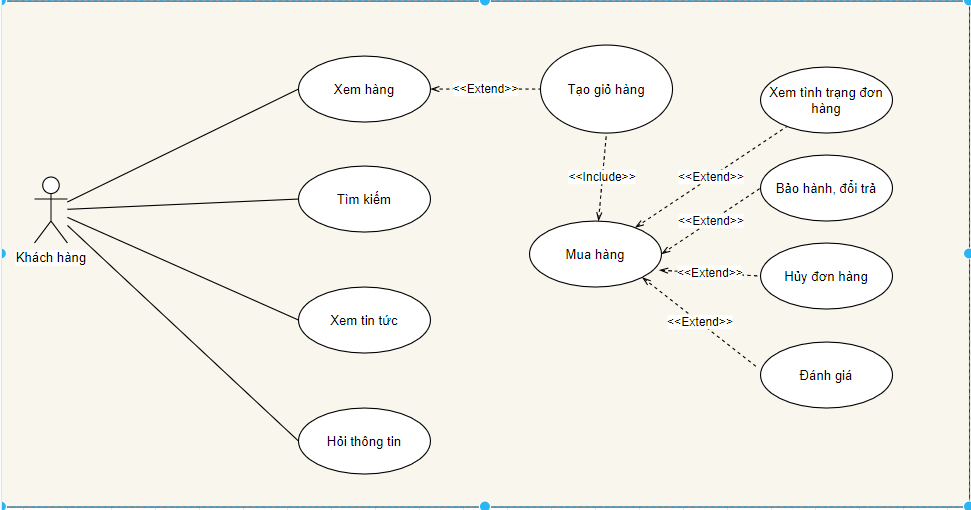
* Chức năng của khách hàng: Tìm kiếm sản phẩm, xem chi tiết sản phẩm, đặt hàng, tạo giỏ hang, đặt hàng, chat với nhân viên, gửi bảo hành.

## Các yêu cầu phi chức năng

* Hoạt động: nhanh và chính xác, an toàn bảo mật thông tin cá nhân người dùng.
* Giao diện: dễ nhìn, đơn giản, người truy cập lần đầu cũng có thể dễ dàng đặt hang hoặc theo dõi các sản phẩm

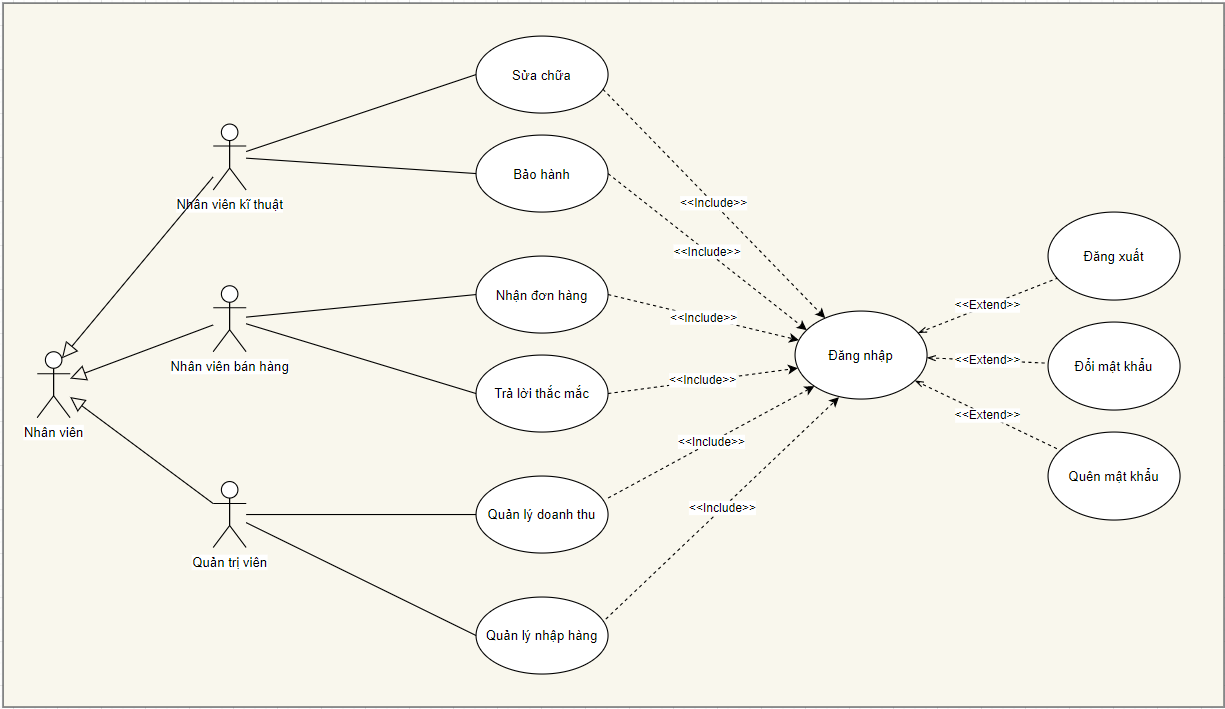
# Phân tích hệ thống

## Các use case chính



*Hình 3.1: Biểu đồ các use case chính.*

## Các use case thứ cấp

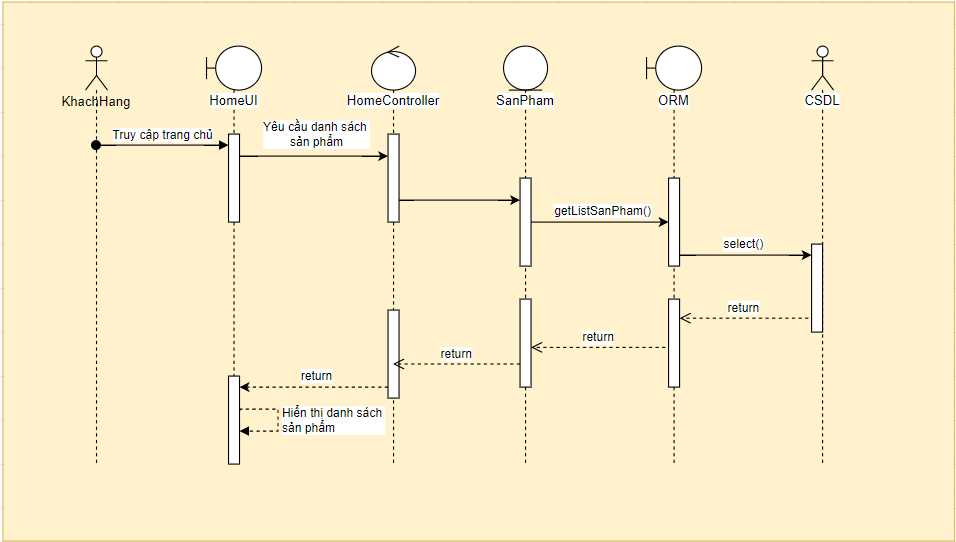


*Hình 3.2: Biểu đồ các use case thứ cấp*

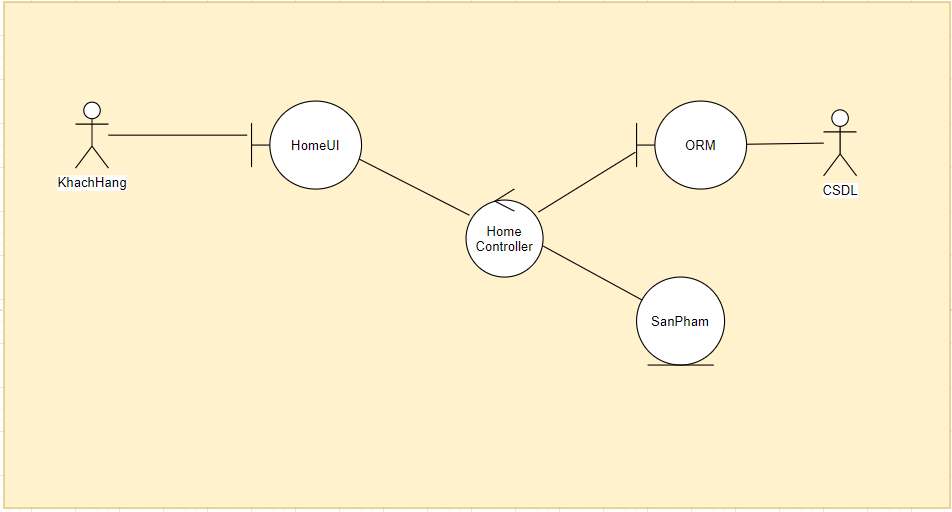
## Phân tích use case

### Phân tích use case “Xem hàng”

#### Biểu đồ trình tự

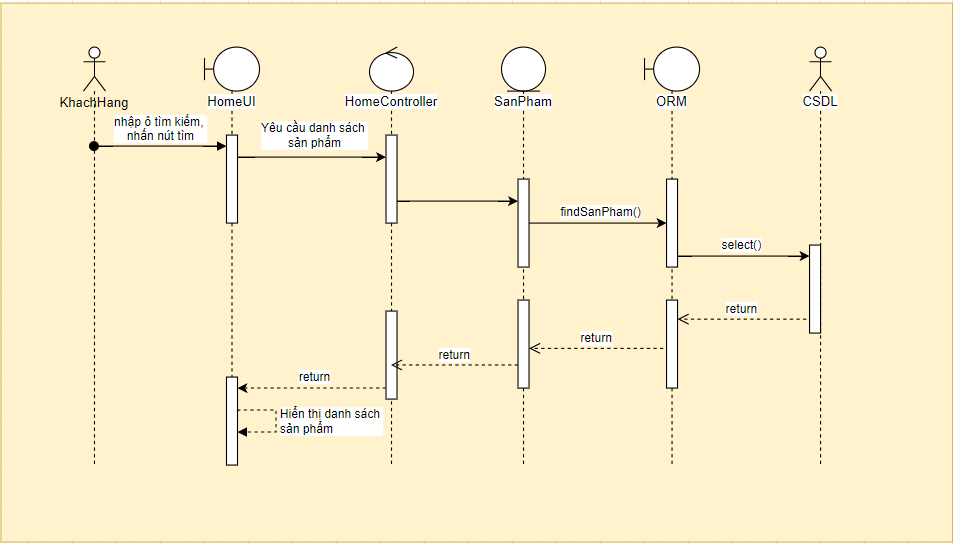


#### Biểu đồ lớp phân tích

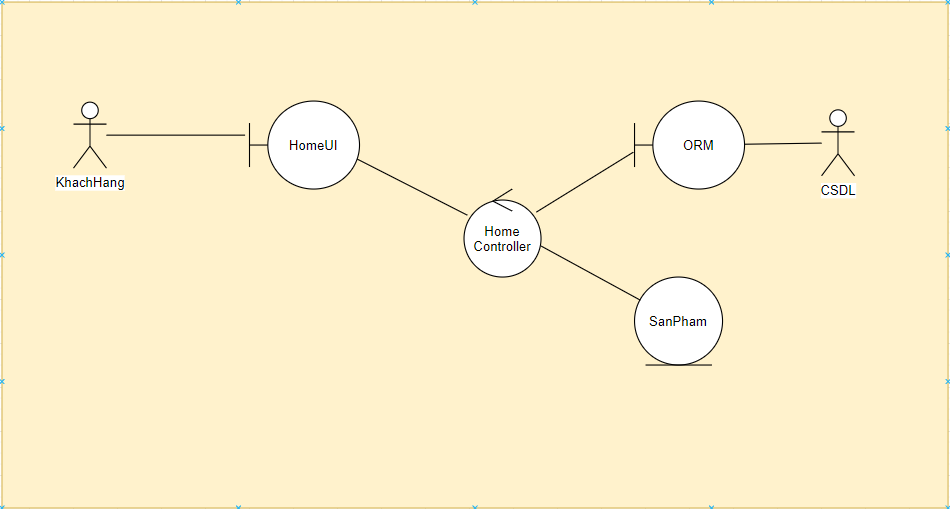


### Phân tích use case “Tìm kiếm”

#### Biểu đồ trình tự

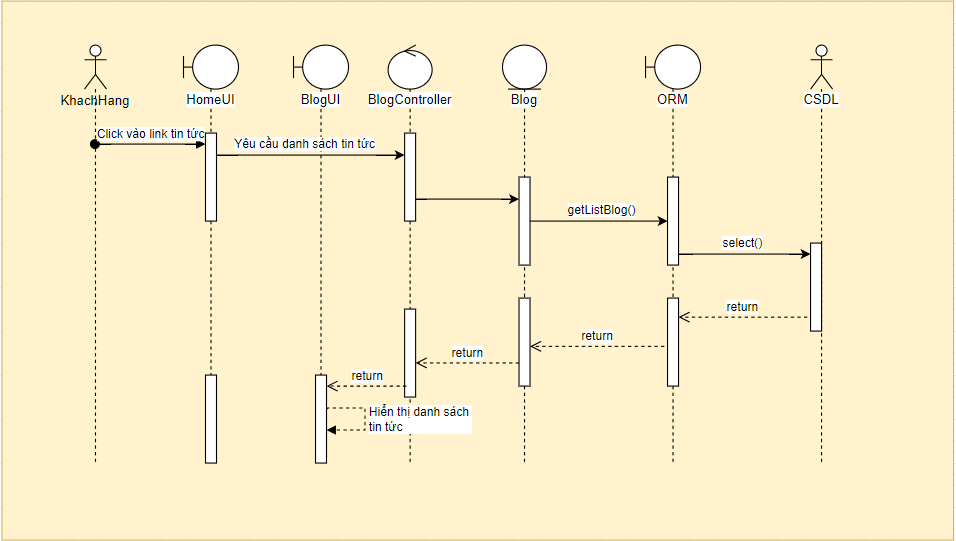


#### Biểu đồ lớp phân tích

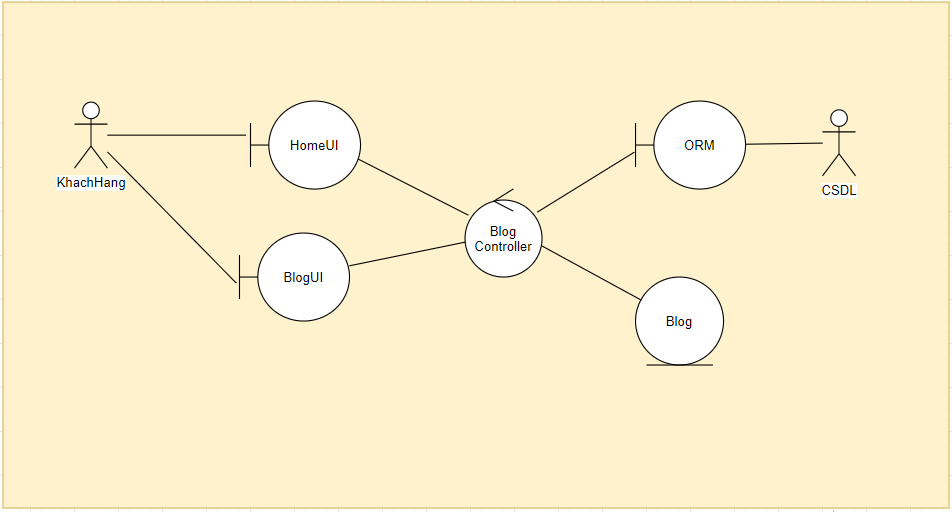


### Phân tích use case “Xem tin tức”

#### Biểu đồ trình tự

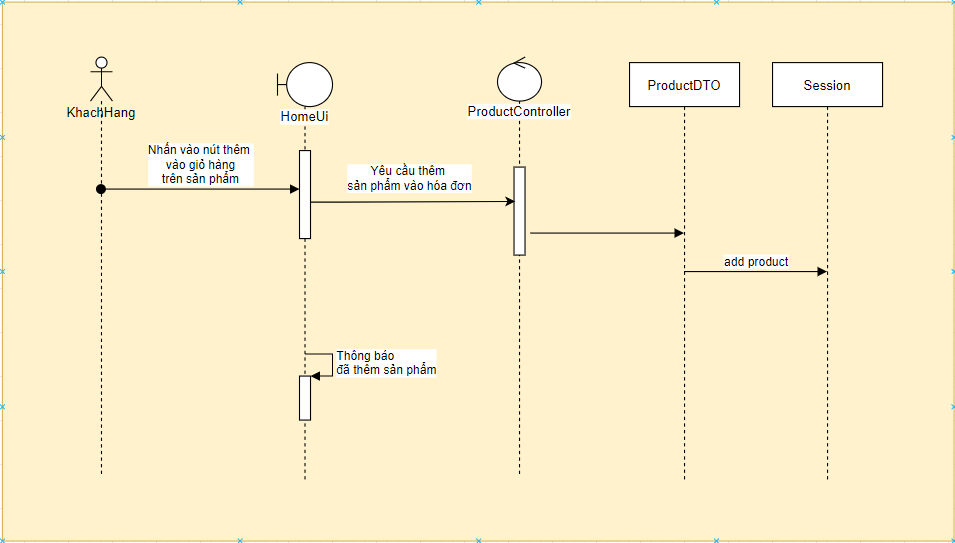


#### Biểu đồ lớp phân tích

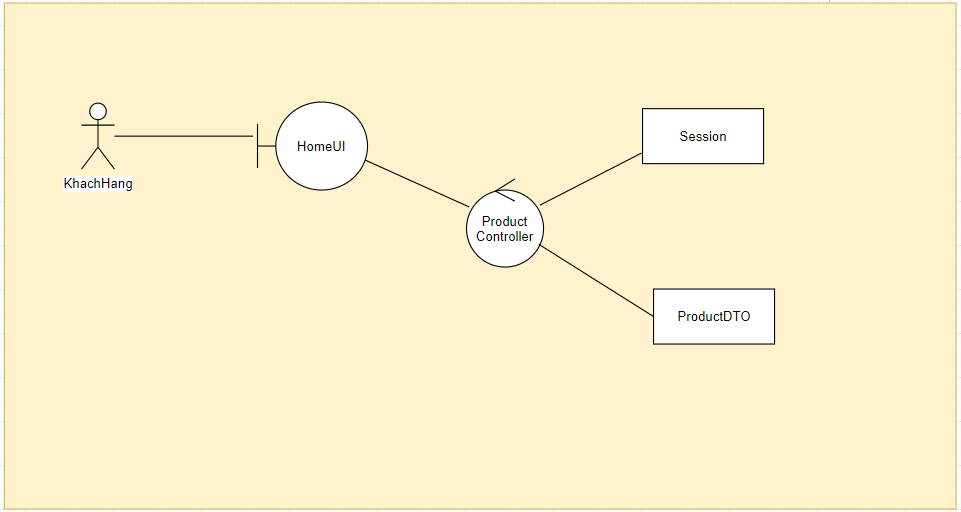


### Phân tích use case “Tạo giỏ hàng”

#### Biểu đồ trình tự

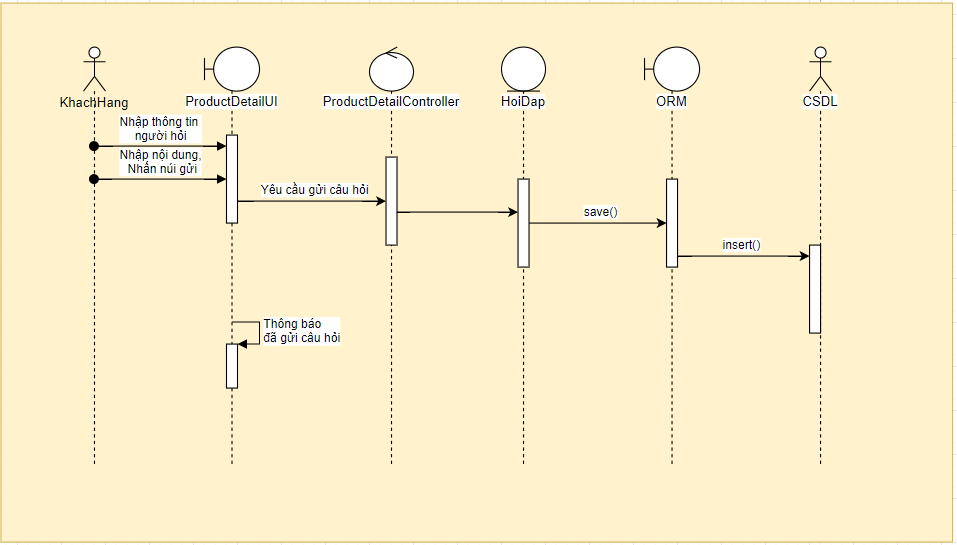


#### Biểu đồ lớp phân tích

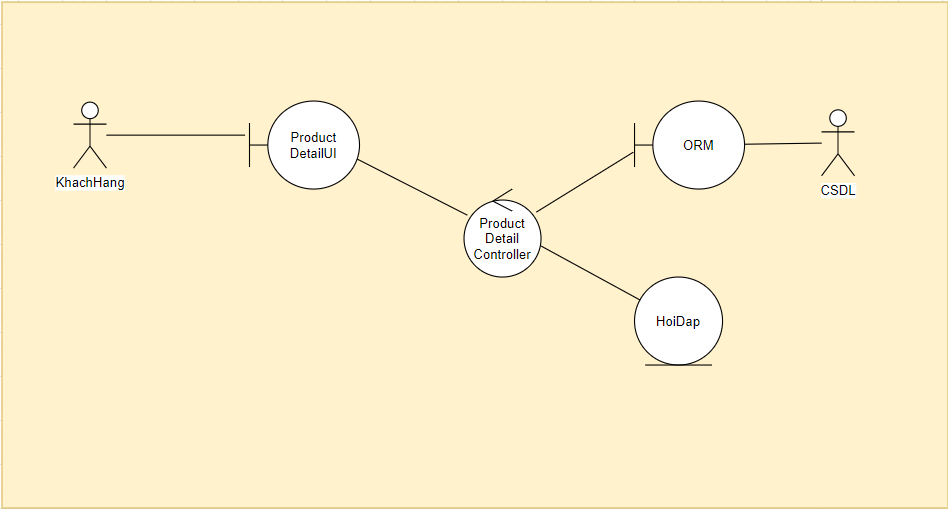


### Phân tích use case “Hỏi thông tin”

#### Biểu đồ trình tự

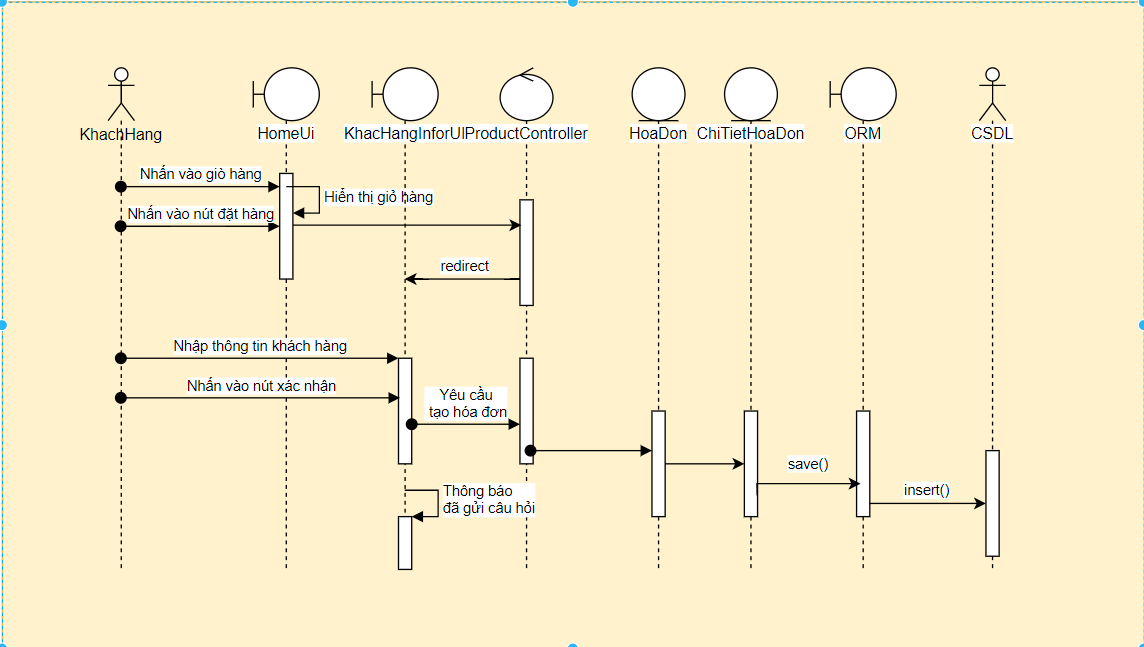


#### Biểu đồ lớp phân tích



### Phân tích use case “Mua hàng”

#### Biểu đồ trình tự

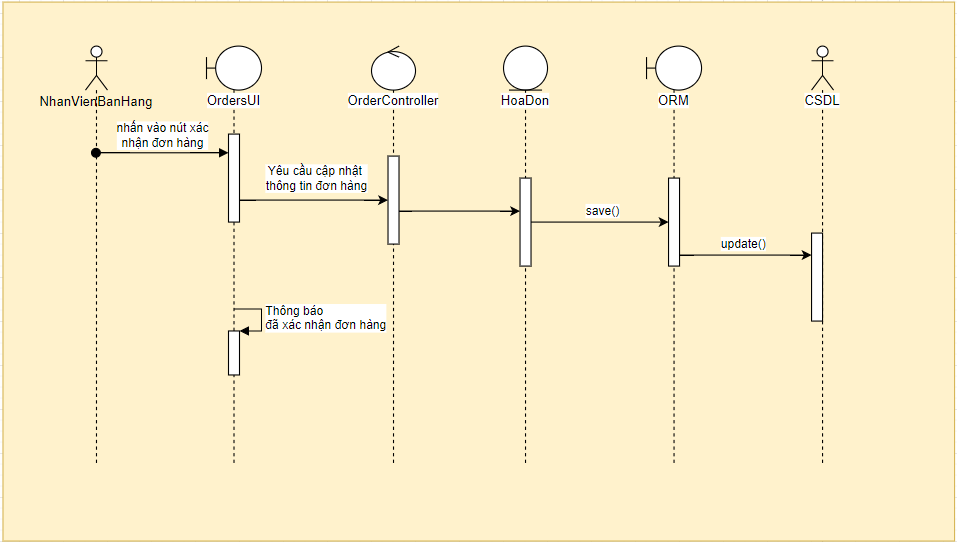


#### Biểu đồ lớp phân tích

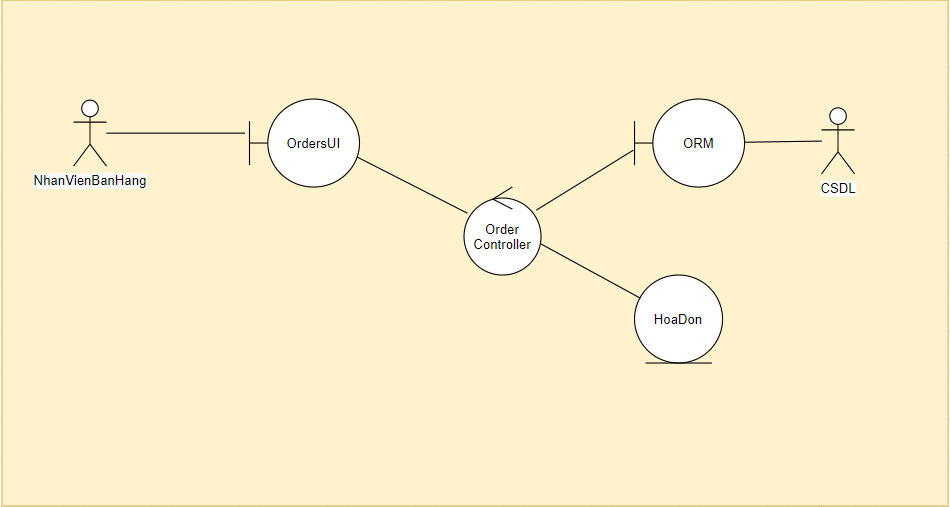


### Phân tích use case “Nhận đơn hàng”

#### Biểu đồ trình tự

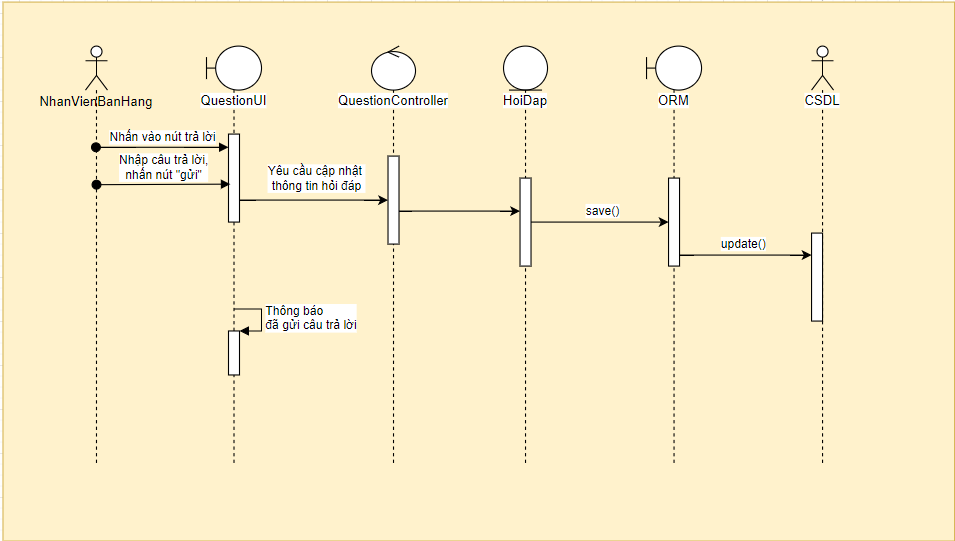


#### Biểu đồ lớp phân tích

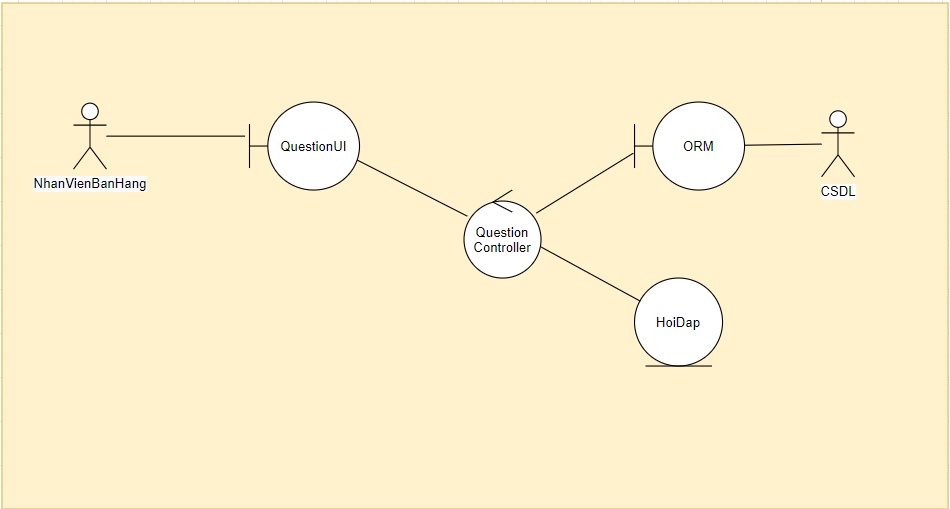


### Phân tích use case “Trả lời thắc mắc”

#### Biểu đồ trình tự



#### Biểu đồ lớp phân tích



# Thiết kế cơ sở dữ liệu

## Mô tả hệ thống

Hệ thống cần lưu thông tin về các sản phẩm bao gồm: Mã sản phẩm, Tên sản phẩm, ảnh, đường dẫn đến sản phẩm, loại sản phẩm, chi tiết sản phẩm.

Mỗi sản phẩm có nhiều giá các nhau phụ thuộc vào dung lượng bộ nhớ và màu sắc.

Mỗi sản phẩm có thể có 0 hoặc 1 chế độ bảo hành. Mỗi chế độ bảo hành có thể thuộc về 1 hoặc nhiều sản phẩm.

Mỗi sản phẩm có thể có 0 hoặc 1 chương trình khuyến mãi. Mỗi chương trình khuyến mãi có thể thuộc về một hoặc nhiều sản phẩm

Khách hàng có thể đến cửa hàng để mua sản phẩm hoặc đặt hàng qua website.

Mỗi khách hàng có thể mua nhiều sản phẩm, mỗi sản phẩm có thể được nhiều khách hàng mua.

Mỗi sản phẩm chỉ thuộc về một hãng, mỗi hãng có một hoặc nhiều sản phẩm.

Mỗi nhân viên bán hàng sẽ tiếp nhận và quản lí 1 đơn hàng, đơn hàng sẽ có nhiều trạng thái(đang xác nhận, đang giao, đã hoàn thành).

Nhân viên quản lý kho sẽ xử lý các phiếu nhập. Mỗi phiếu nhập được xử lí bởi một nhân viên và mỗi nhân viên xử lý nhiều phiếu nhập. Mỗi phiếu nhập được lấy từ một nhà cùng cấp, mỗi nhà cung cấp có thể cung cấp nhiều phiếu nhập. Mỗi phiếu nhập có một hoặc nhiều sản phẩm. Mỗi sản phẩm thuộc về nhiều phiếu nhập.

Mỗi sản phẩm được đánh giá từ một hoặc nhiều khách hàng. Mỗi khách hàng có thể đánh giá một hoặc nhiều sản phẩm.

Mỗi sản phẩm có 0, 1 hoặc nhiều câu hỏi. Mỗi câu hỏi chỉ thuộc về một hoặc nhiều sản phẩm. Mỗi câu hỏi có một hoặc nhiều cầu trả lời.

## Xác định các lớp thực thể

Bảng Hãng(hang) lưu thông tin của các hãng sản phẩm bao gồm

|  |  |  |
| --- | --- | --- |
| STT | Tên cột | Mô tả |
| 1 | id\_hang(PK) | Mã hãng |
| 2 | image | Ảnh của hãng |
| 3 | name | Tên hãng |

Bảng Sản Phẩm(san\_pham) lưu thông tin của các sản phẩm bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | id\_san\_pham(PK) | Mã sản phẩm |
| 2 | cpu | Mô tả cpu điện thoại |
| 3 | name | Tên sản phẩm |
| 4 | bo\_nho\_trong | Bộ nhớ trong |
| 5 | Camera | Mô cả camera |
| 6 | cong\_nghe\_man\_hinh | Mô tả công nghệ màn hình |
| 7 | Description | Mô tả chung |
| 8 | Do\_phan\_giai\_man\_hinh | Độ phân giải màn hình |
| 9 | Kich\_thuoc\_man\_hinh | Kích thước màn hình |
| 10 | Pin | Dung lượng pin |
| 11 | Sim | Thông tin về sim |
| 12 | Type | Điện thoại hoặc phụ kiện |
| 13 | Id\_hang(FK) | Mã hãng |

Bảng chi tiết sản phẩm(chi\_tiet\_san\_pham) lưu thông tin về chi tiết sản phẩm bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_chi\_tiet\_san\_pham(PK) | Mã chi tiết sản phẩm |
| 2 | Gia\_cu | Giá cũ |
| 3 | Gia\_moi | Giá mới |
| 4 | Mau\_sac | Màu sắc |
| 5 | Version | Phiên bản sản phẩm |
| 6 | Id\_khuyen mai(FK) | Mã khuyến mãi |
| 7 | Id\_ san\_pham(FK) | Mã sản phẩm |

Bảng khuyến mãi (khuyen\_mai) lưu thông tin về các chương trình khuyến mãi bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_khuyen\_mai(PK) | Mã khuyến mãi |
| 2 | Chi\_tiet | Chi tiết khuyến mãi |
| 3 | Phan\_tram | Phần trăm giảm giá |

Bảng nhân viên (nhan\_vien) lưu thông tin về các nhân viên bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_nhan\_vien(PK) | Mã nhân viên |
| 2 | Full\_name | Họ và tên |
| 3 | Password | Mật khẩu |
| 4 | Phone | Số điện thoại |
| 5 | Role | Quyền |
| 6 | Username | Tên đăng nhập |
| 7 | Active | Hoạt động |

Bảng khách hàng (khach\_hang) lưu thông tin về khách hàng bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_khach\_hang(PK) | Mã khách hàng |
| 2 | Name | Tên khách hàng |
| 3 | Phone | Số điện thoại |
| 4 | Address | Địa chỉ |
| 5 | Email | email |

Bảng nhà cung cấp(nha\_cung\_cap) lưu thông tin về nhà cung cấp gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_nha\_cung\_cap (PK) | Mã nhà cung cấp |
| 2 | Name | Tên nhà cung cấp |
| 3 | Phone | Số điện thoại |
| 4 | Address | Địa chỉ |
| 5 | Email | Email |

Bảng phiếu nhập(phieu\_nhap) lưu thông tin về phiếu nhập bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_phieu\_nhap(PK) | Mã phiếu nhập |
| 2 | Ngay\_lap | Ngày lập phiếu |
| 3 | Id\_nha\_cung\_cap(FK) | Mã nhà cung cấp |
| 4 | Id\_nhan\_viên(FK) | Mã nhân viên |

Bảng hóa đơn (hoa\_don) lưu thông tin về hóa đơn bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_hoa\_don(PK) | Mã hóa đơn |
| 2 | Ngay\_lap | Ngày lập hóa đơn |
| 3 | Status | Tình trạng đơn hàng |
| 4 | Id\_nhan\_viên(FK) | Mã nhân viên |
| 5 | Id\_khach\_hang(FK) | Mã khách hàng |

Bảng Chi tiết hóa đơn lưu thông tin chi tiết của một hóa đơn bao gồm:

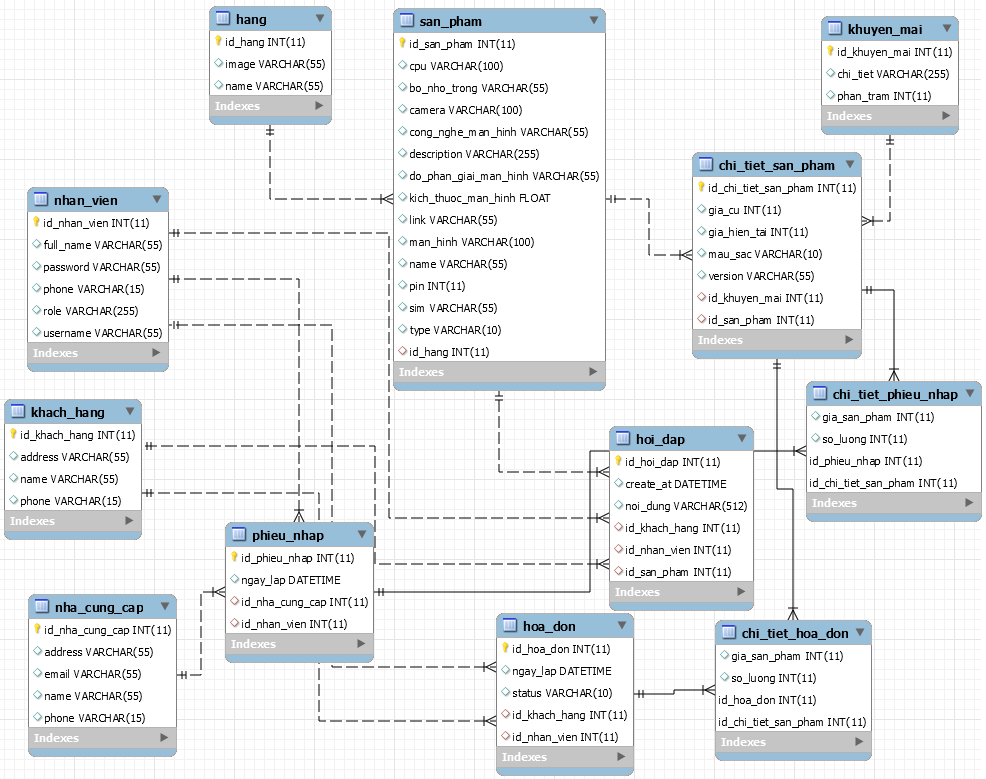
|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_chi\_tiet\_hoa\_don(PK) | Mã chi tiết hóa đơn |
| 2 | Gia\_san\_pham | Giá sản phẩm |
| 3 | So\_luong | Số lượng sản phẩm |
| 4 | Id\_hoa\_don(FK) | Mã hóa đơn |
| 5 | Id\_chi\_tiet\_san\_pham(FK) | Mã chi tiết sản phẩm |

Bảng hỏi đáp lưu thông tin về hỏi đáp sản phẩm bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_hoi\_dap(PK) | Mã hỏi đáp |
| 2 | Create\_at | Thời gian hỏi |
| 3 | Noi\_dung | Nội dung hỏi |
| 4 | Id\_khach\_hang(FK) | Mã khách hàng |
| 5 | Id\_san\_pham(FK) | Mã sản phẩm |
| 6 | Reply | Trả lời |
| 7 | Id\_nhan\_vien(FK) | Mã nhân viên |

Bảng đánh giá lưu thông tin về đánh giá của khách hàng về sản phẩm bao gồm:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên cột** | **Mô tả** |
| 1 | Id\_danh\_gia(PK) | Mã đánh giá |
| 2 | Star | Số sao |
| 3 | comment | Nội dung đánh giá |
| 4 | Id\_khach\_hang(FK) | Mã khách hàng |
| 5 | Id\_san\_pham(FK) | Mã sản phẩm |



# Thiết kế giao diện

## Thiết kế giao diện use case “Xem hàng”

### Hình dùng màn hình



## Thiết kế giao diện use case “Tạo giỏ hàng”

### Hình dung màn hình



## Thiết kế giao diện use case “mua hàng”

### Hình dung màn hình



## Thiết kế giao diện use case “Xem tình trạng đơn hàng”



## Thiết kế giao diện use case “Tìm kiếm”

### Hình dung màn hình



# Cài đặt chương trình

## Cài đặt MySQL

### Tải MySQL

**MySQL có 2 phiên bản đó là:**

MySQL Cummunity (bản miễn phí)

MySQL Enterprise Edition (bản trả phí)

Trong bài này mình sẽ hướng dẫn bạn cài đặt MySQL phiên bản miễn phí MySQL Community 5.6.21

**Download MySQL Community về tại:**<https://downloads.mysql.com/archives/community/>

**Sau khi tải về bạn sẽ có đủ 3 file như sau:**

Microsoft .NET Framework 4 Client Profile

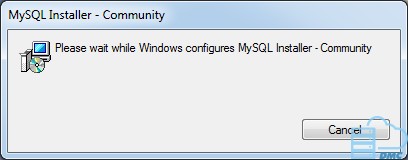
Visual C++ Redistributable for Visual Studio 2013

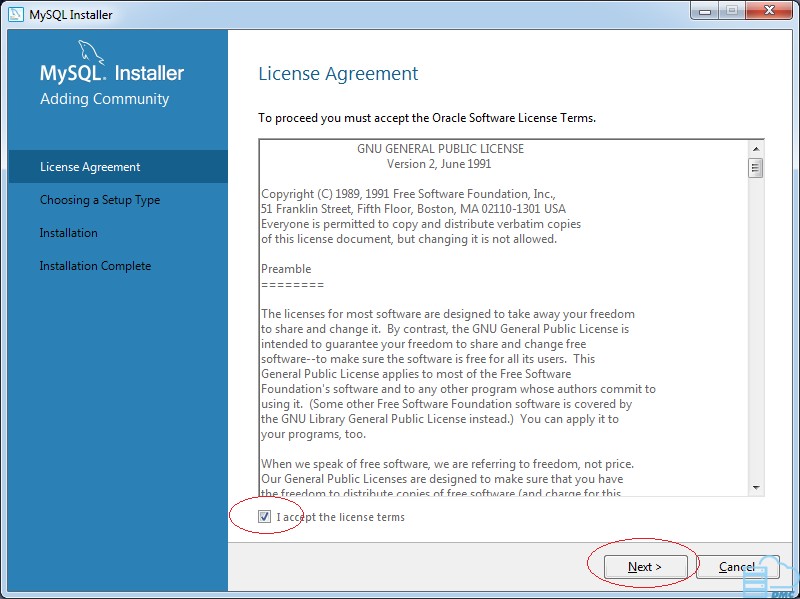
MySQL

### Cài đặt

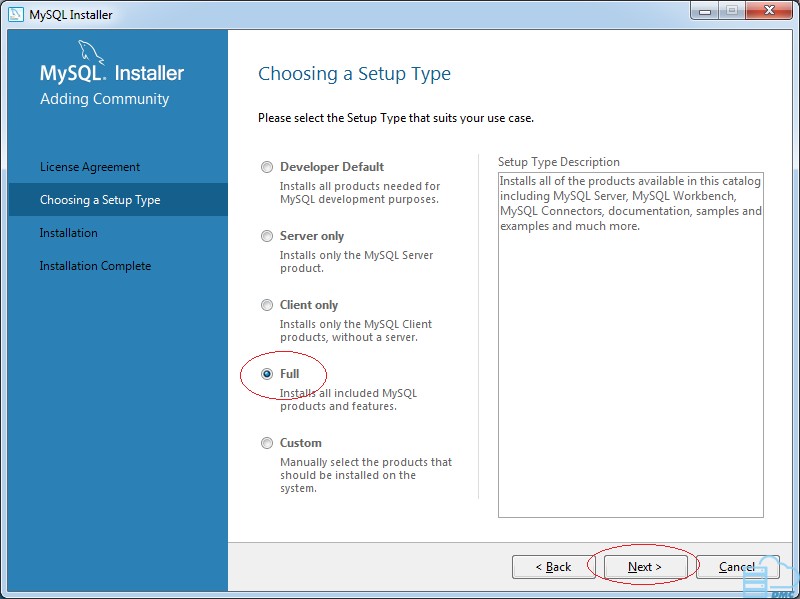
Đầu tiên bạn cần cài đặt Microsoft .NET Framework 4 Client Profile, Visual C++ Redistributable for Visual Studio 2013 trước sau đó cài đặt đến file MySQL

Mở file cài đặt —> **Accept** —>**Next**





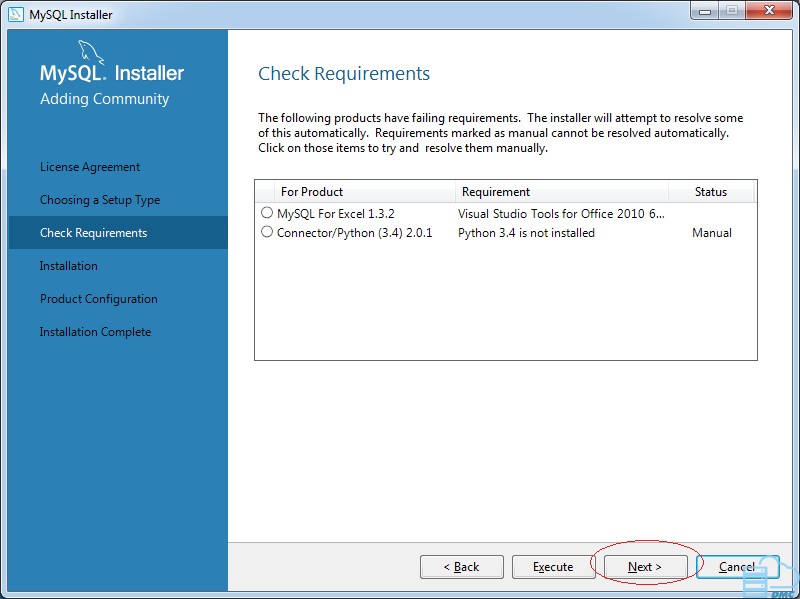
Chọn **Full** để cài đặt tất cả, bao gồm cả Database —> **Next**

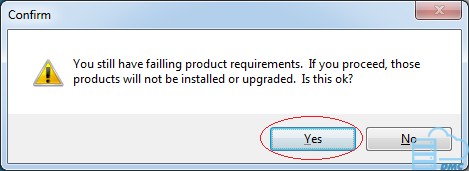


Tại đây bạn sẽ nhận được thông báo thiếu file

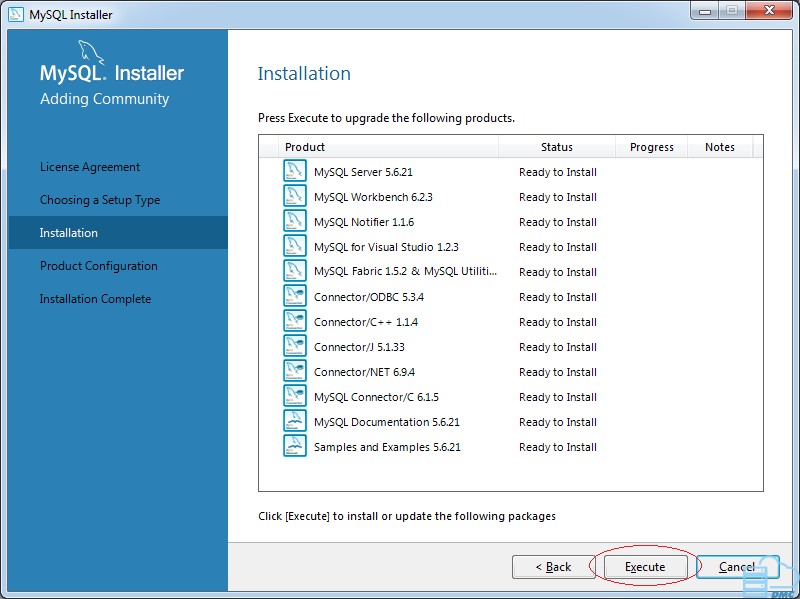
* Visual Studio Tools for Office … & Python 3.4.

Tuy nhiên có thể **bỏ qua** (vì không quan trọng) —> **Next**

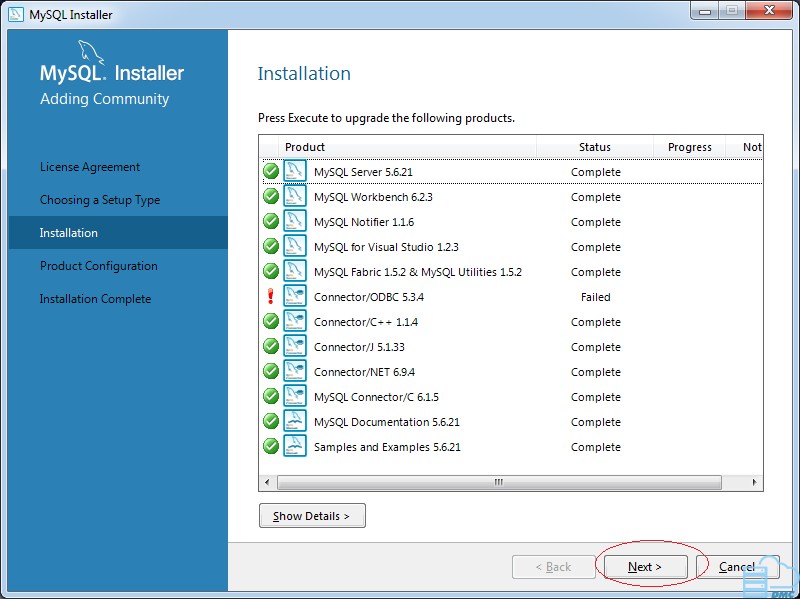




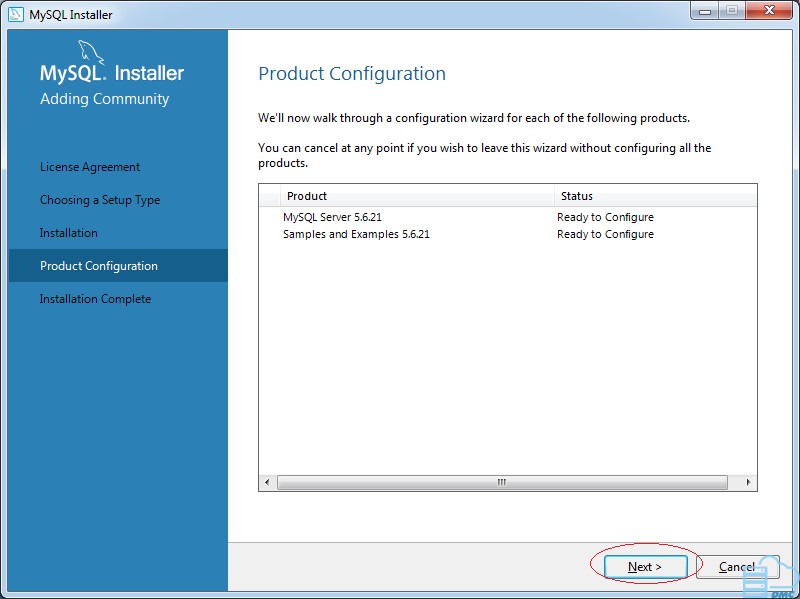
Ở bước này bạn sẽ thấy tất cả các gói được cài đặt. Chọn **Execute**



Chọn **Next**



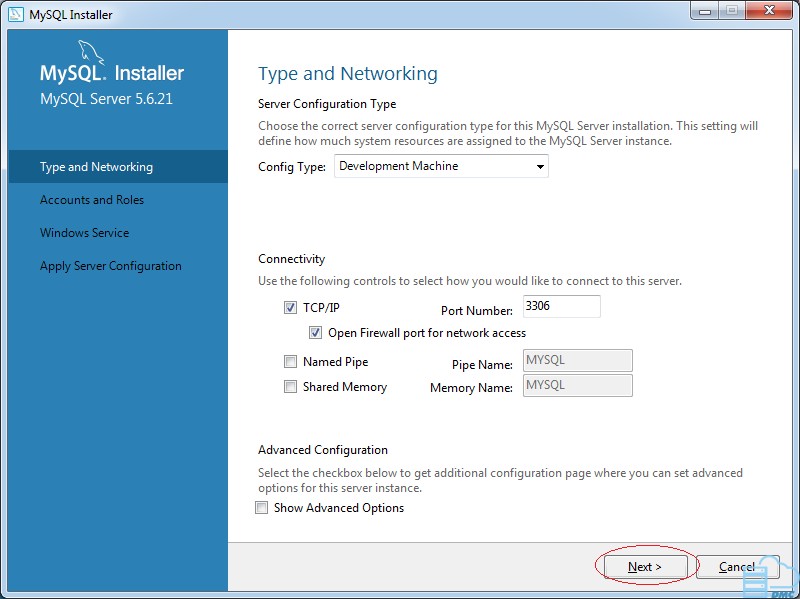
Tiếp theo chúng ta sẽ đến phần cấu hình cho MySQL Server. Chọn **Next**



**Lựa chọn**

* **Config Type:** Development Machine
* **Connectivity:** Click chọn **TCP/IP** —> **Open Firewall**

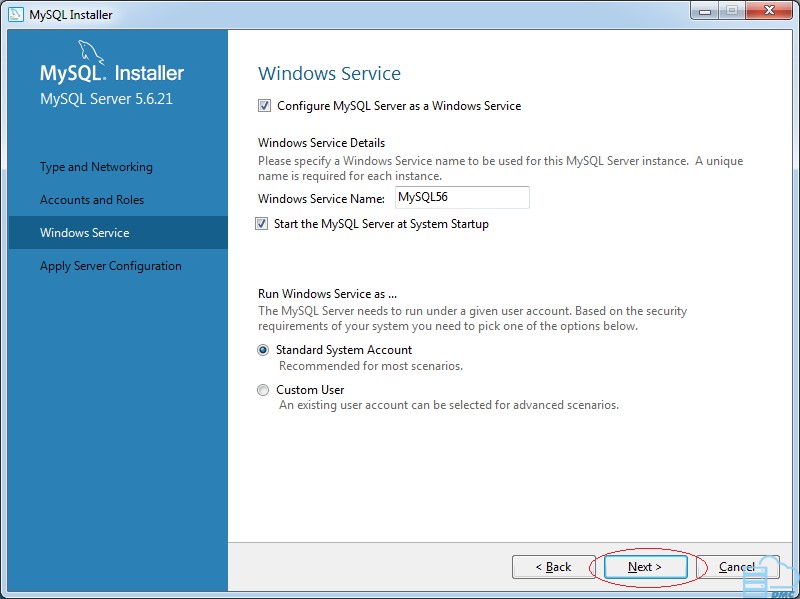
Sau đó click **Next**

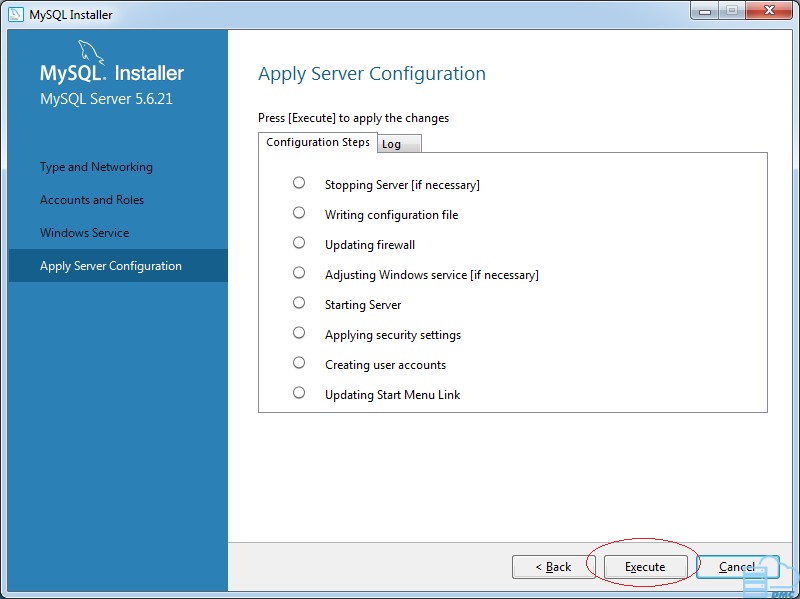


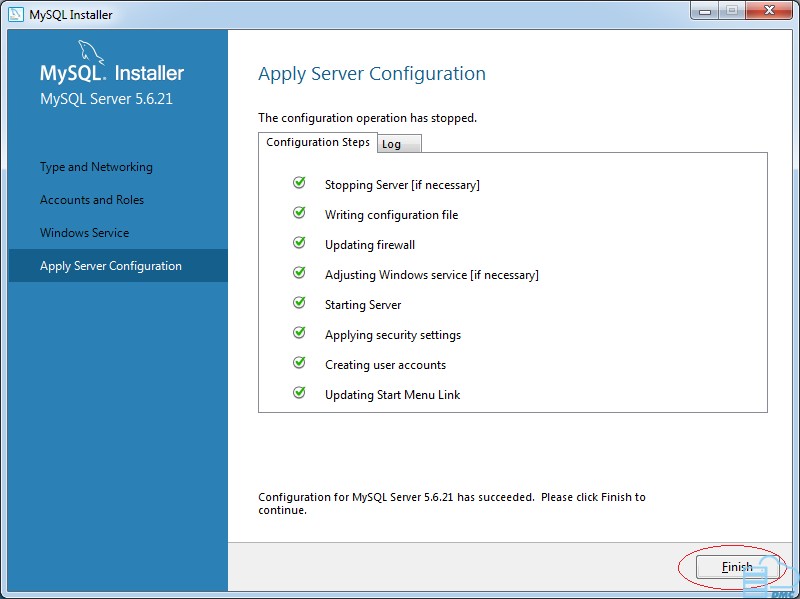
Tại **Accounts and Roles** bạn **điền mật khẩu** của mình vào, sau đó chọn **Next** để đến bước tiếp theo



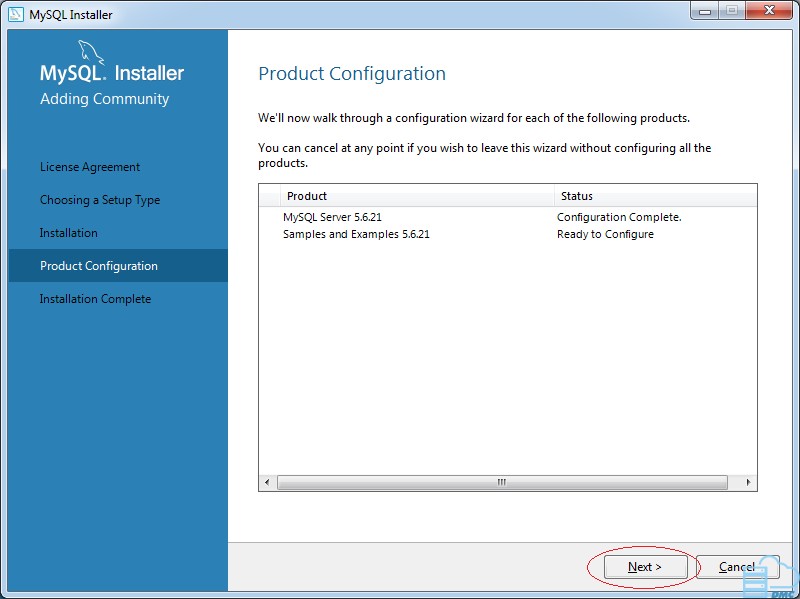
Click chọn như hình sau —> **Next**





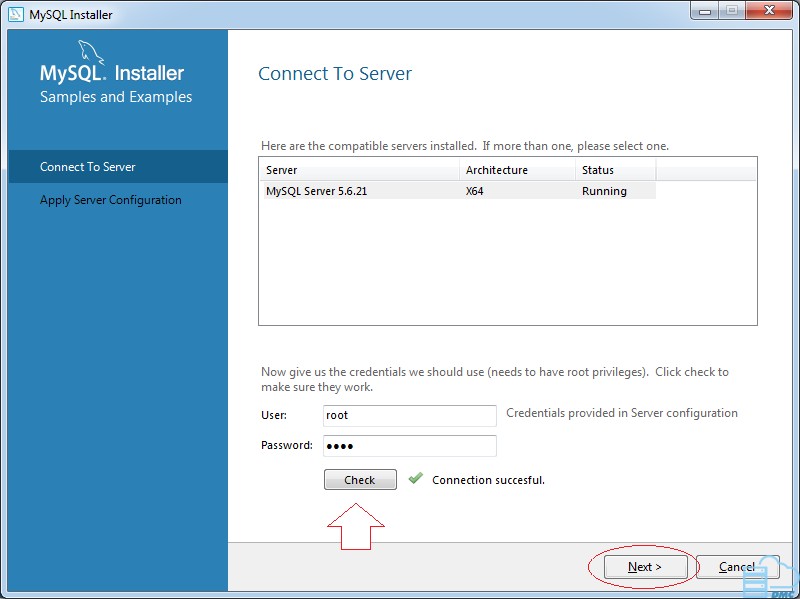


**Đến phần cấu hình Database**

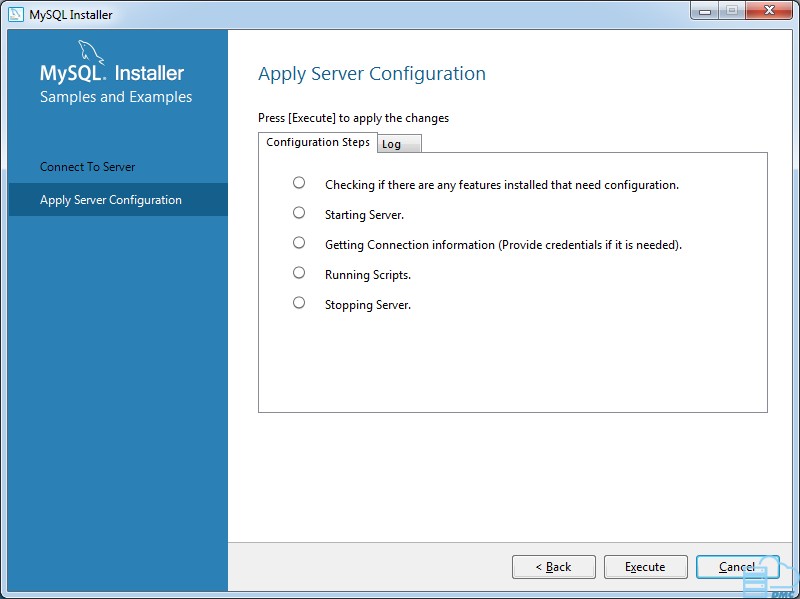


Mặc định **User là root**, bạn **điền mật khẩu đã thiết lập** bên trên vào để **check** và kết nối với MySQL server.

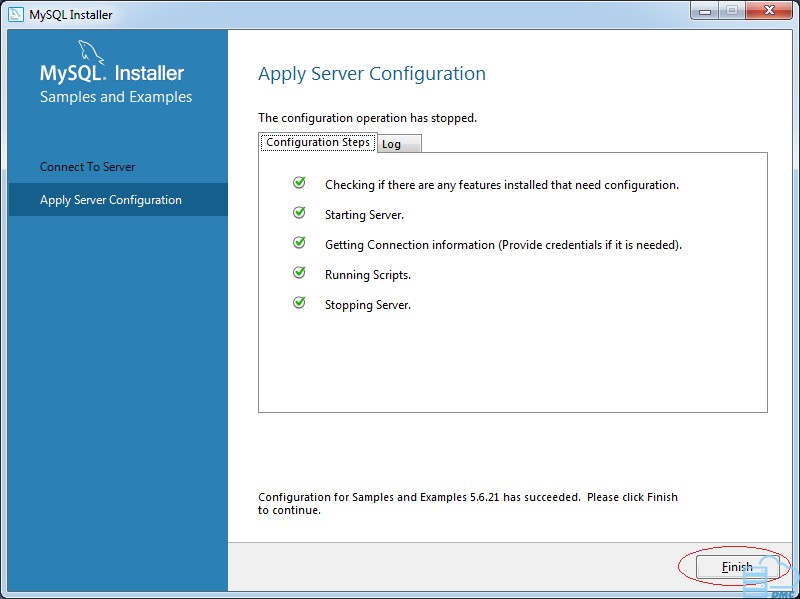
Sau đó chọn **Next**



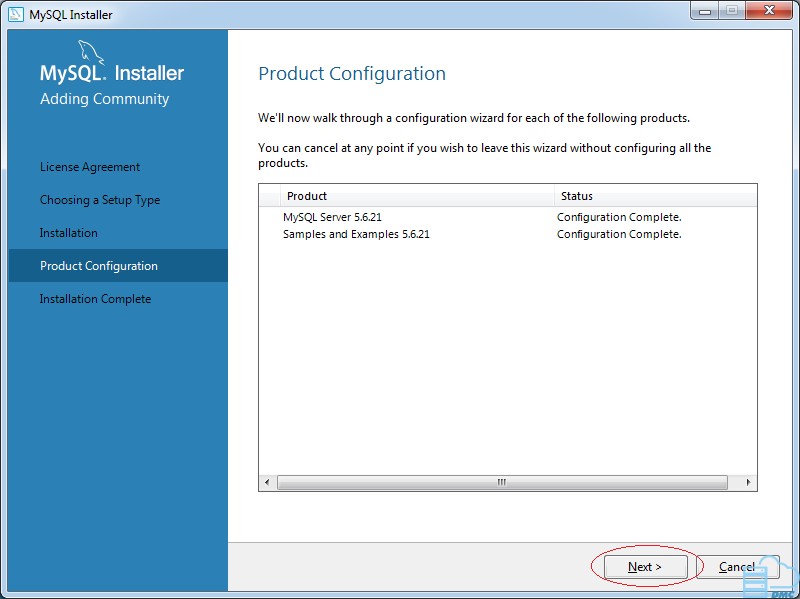
Chọn **Execute**



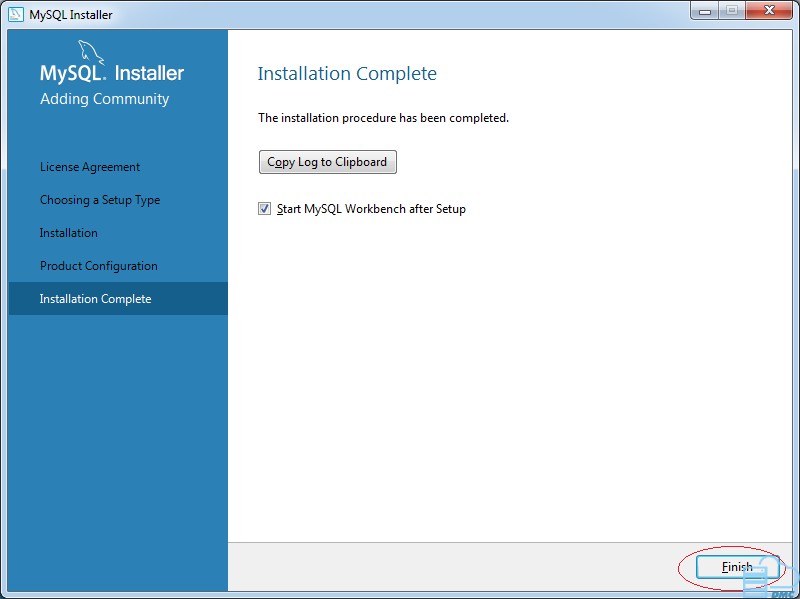
Chờ đợi hoàn tất, chọn**Finish**



Chọn **Next**



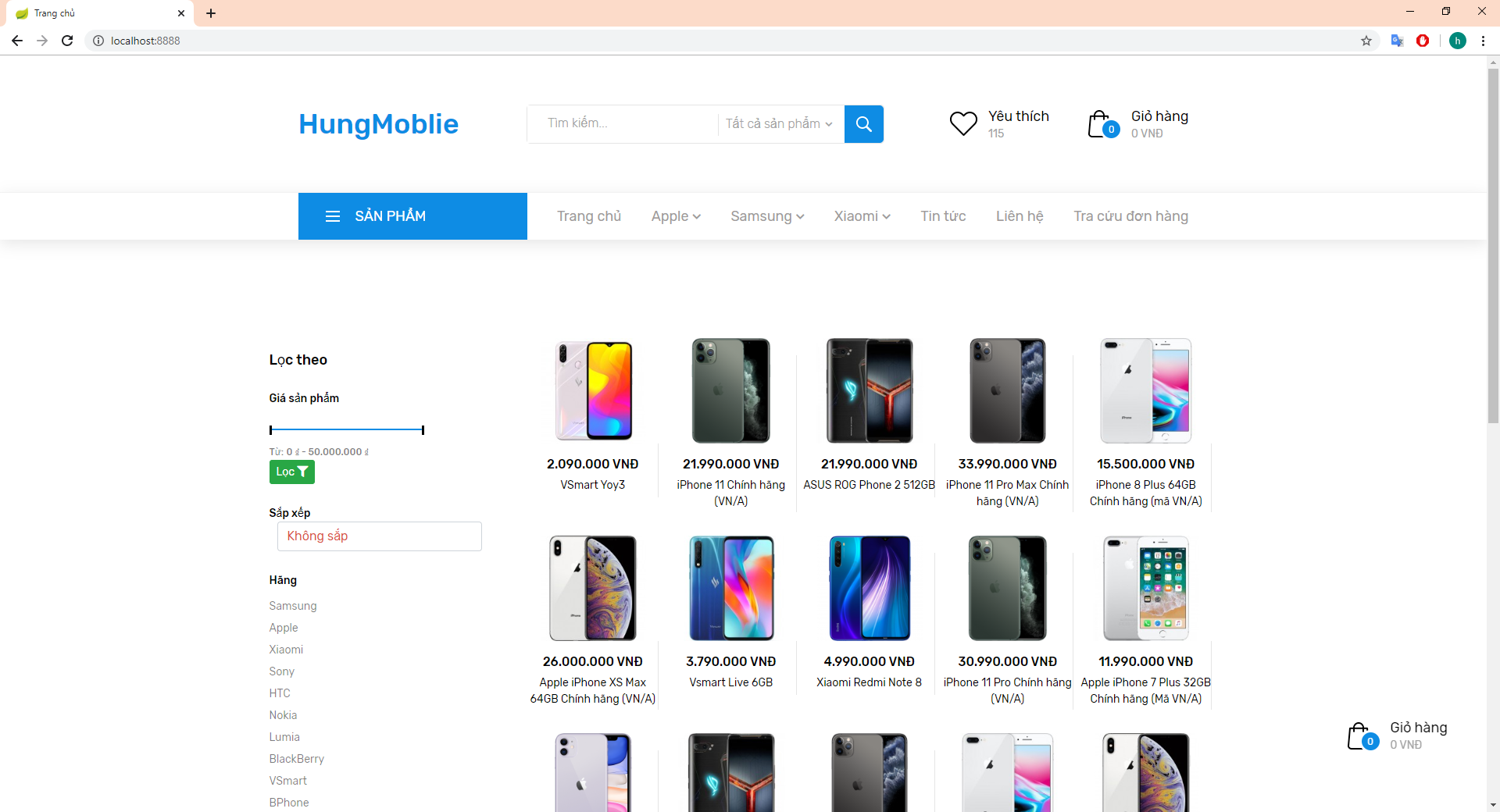
Nhấn**Finish** để hoàn tất quá trình cài đặt MySQL Server



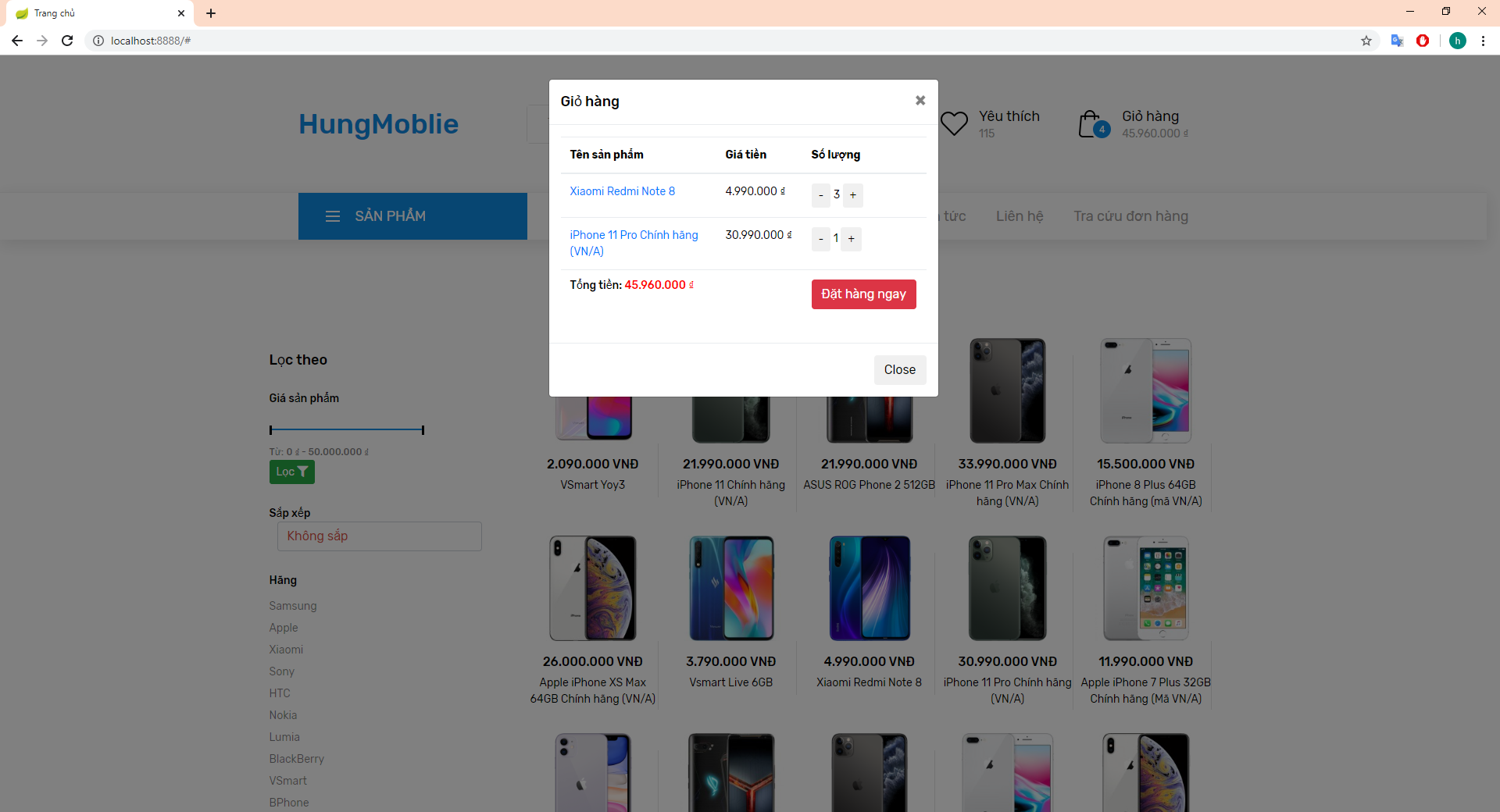
## Cài đặt chương trình

### Giao diện phần người dùng

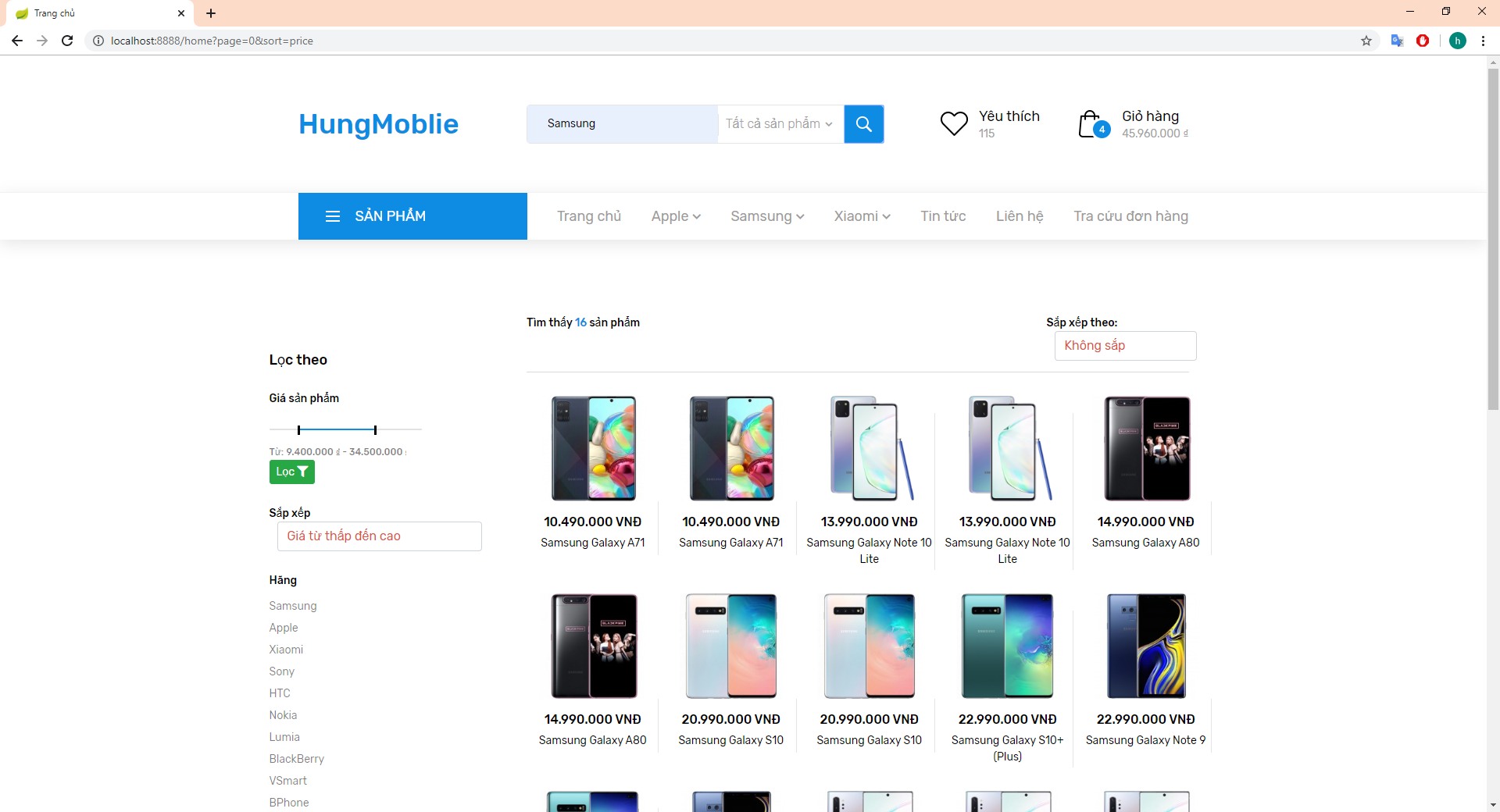
#### Trang chủ



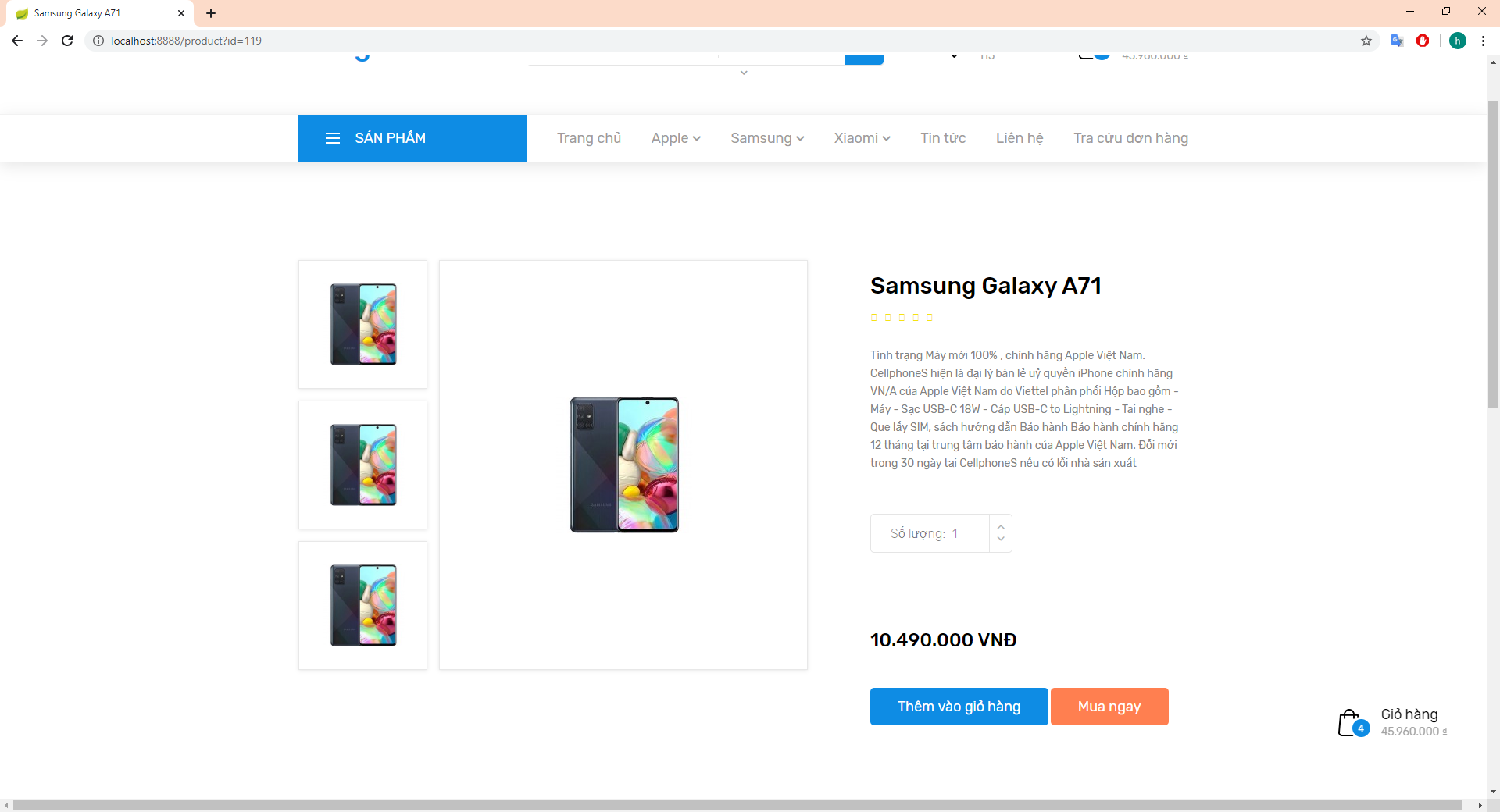
#### Giỏ hàng



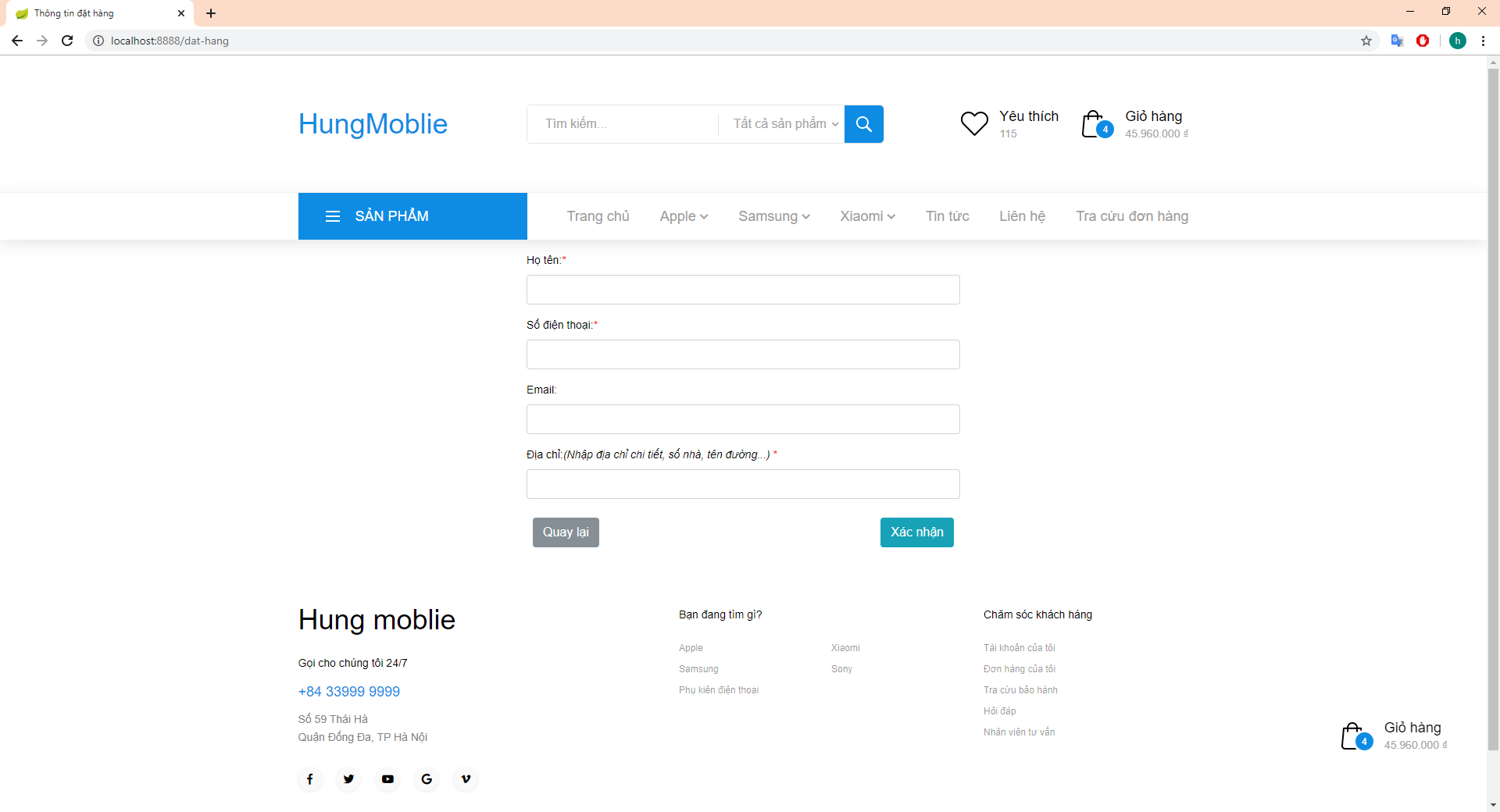
#### Tìm kiếm

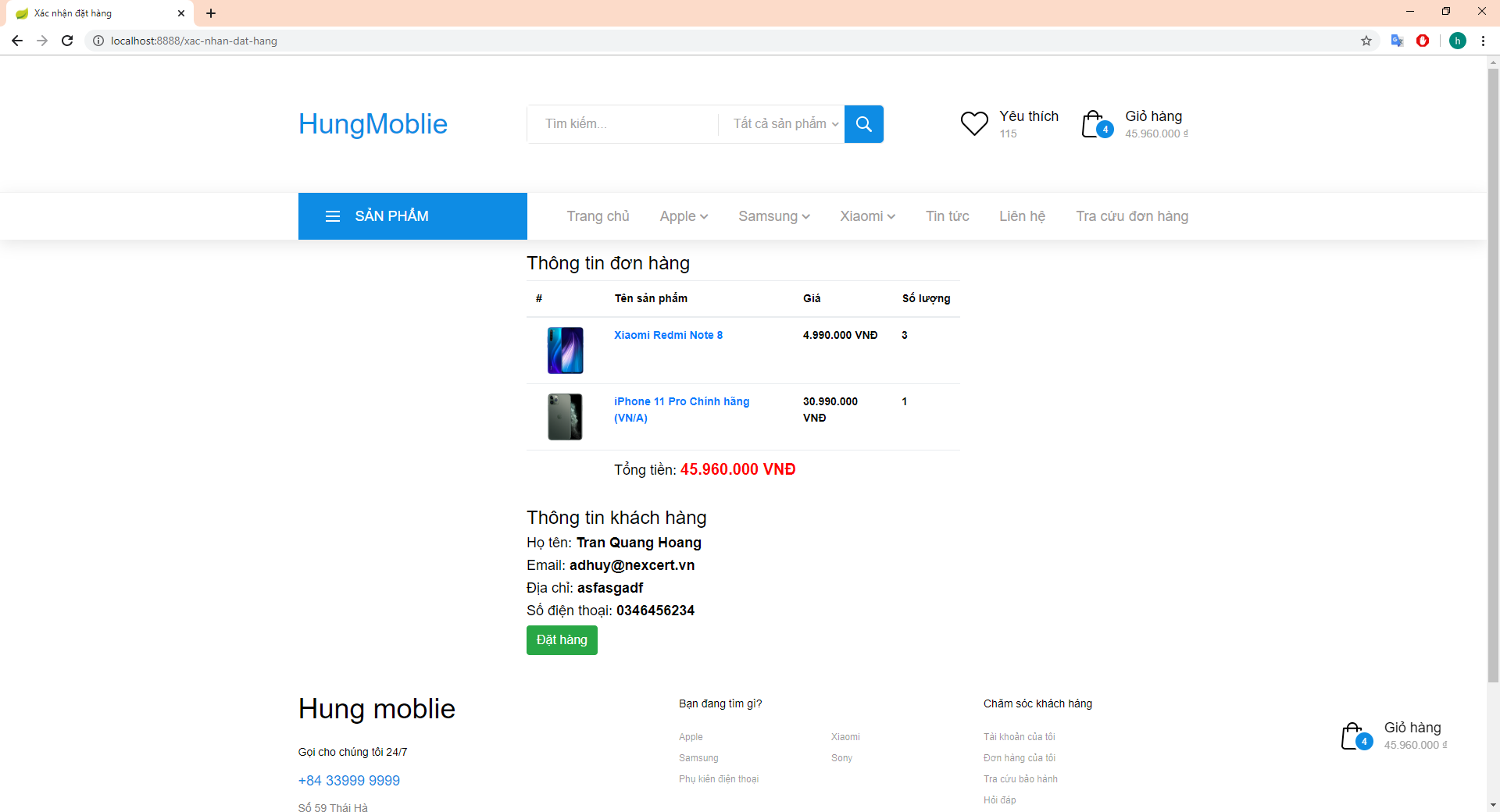


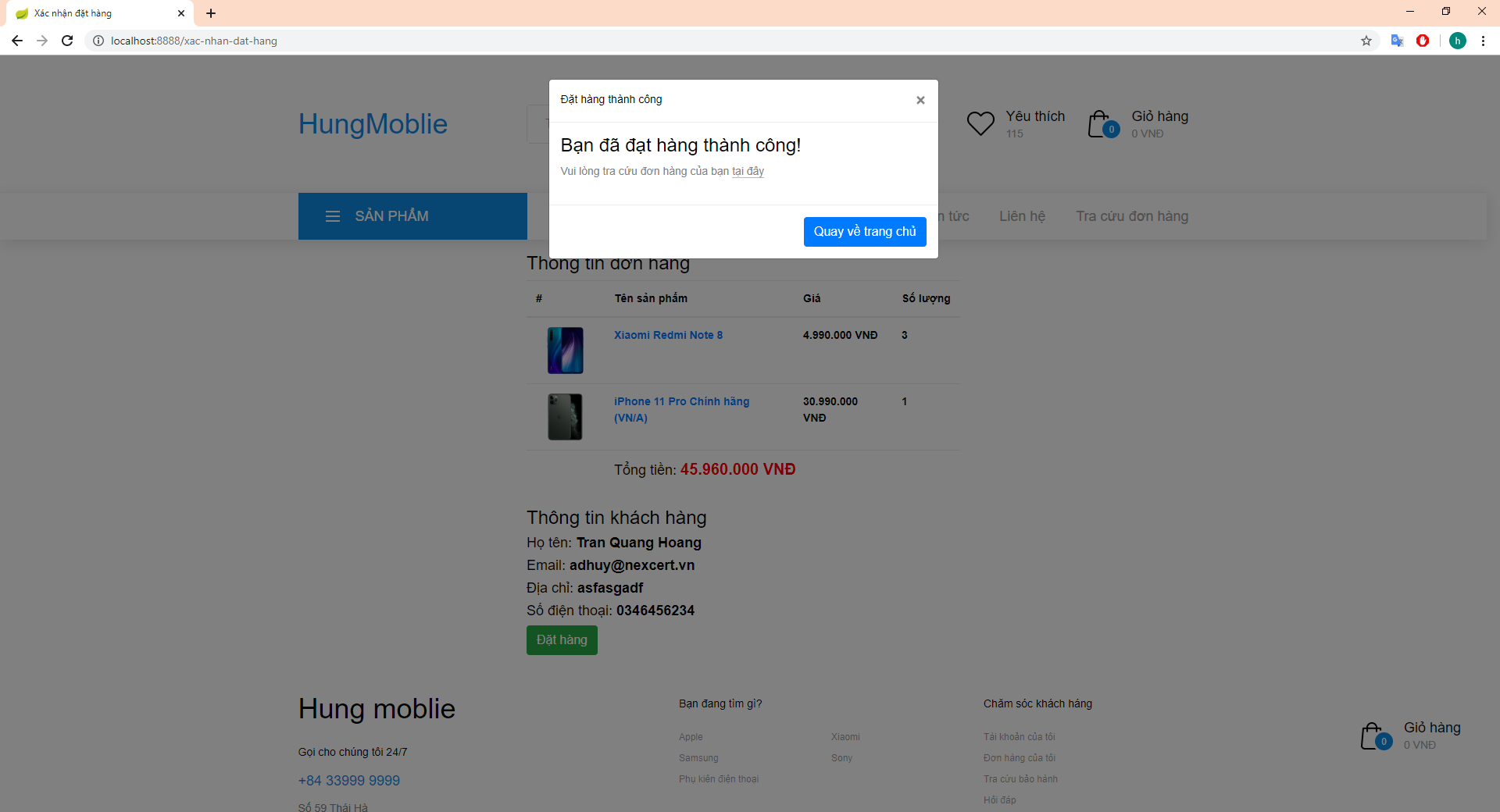
#### Chi tiết sản phẩm



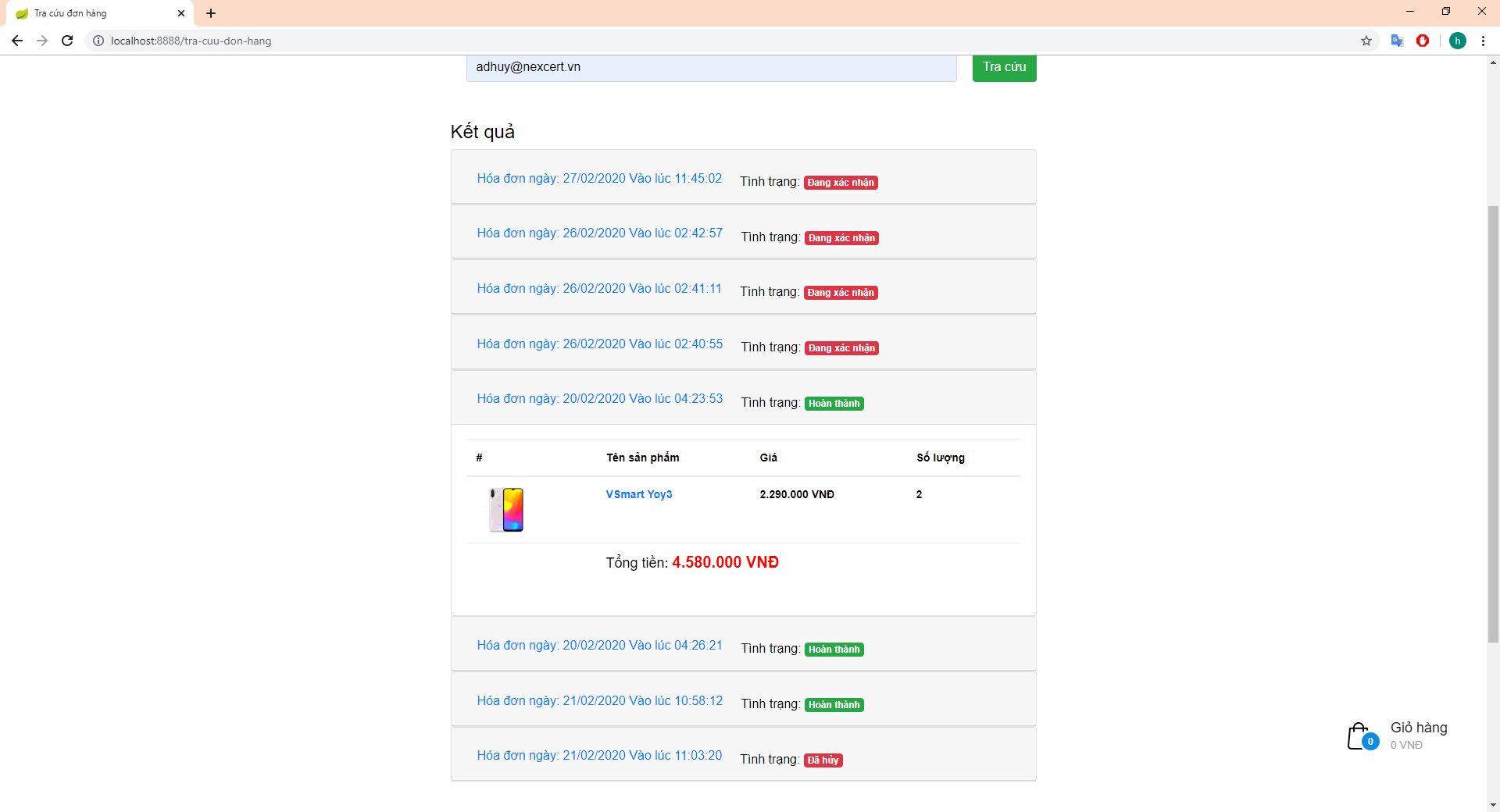
#### Đặt hàng





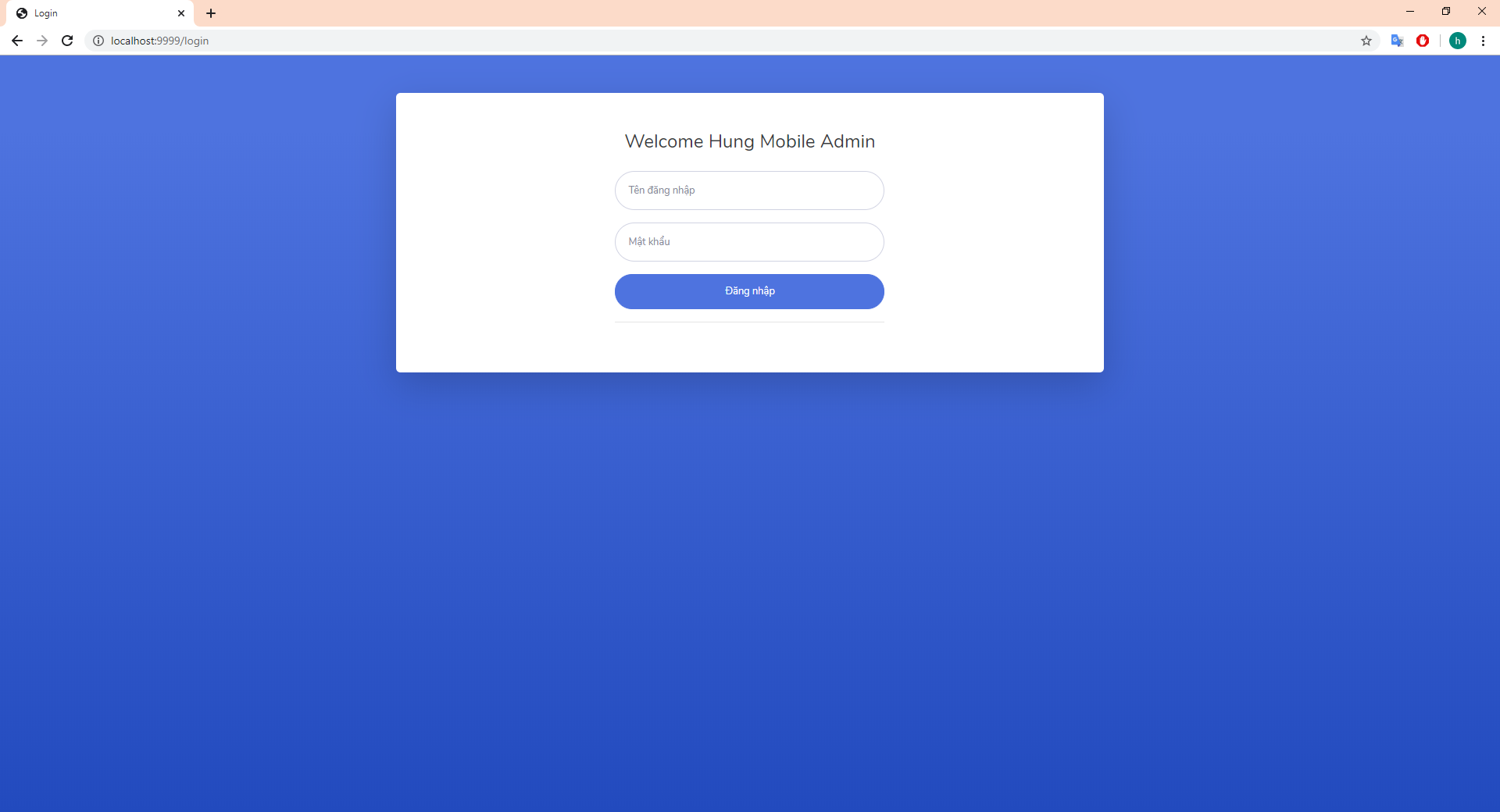


#### Tra cứu đơn hàng

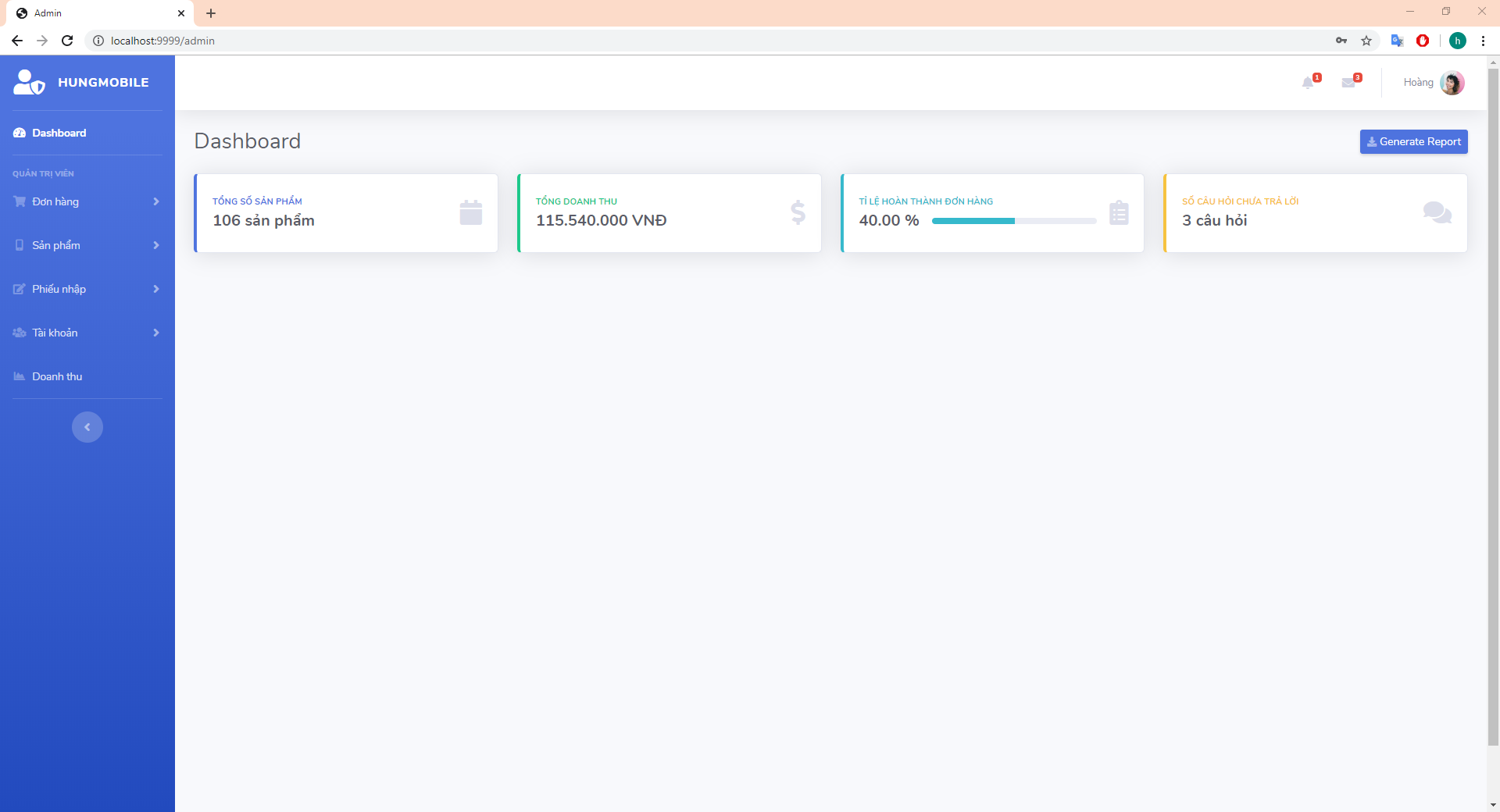


### Giao diện phần quản lý

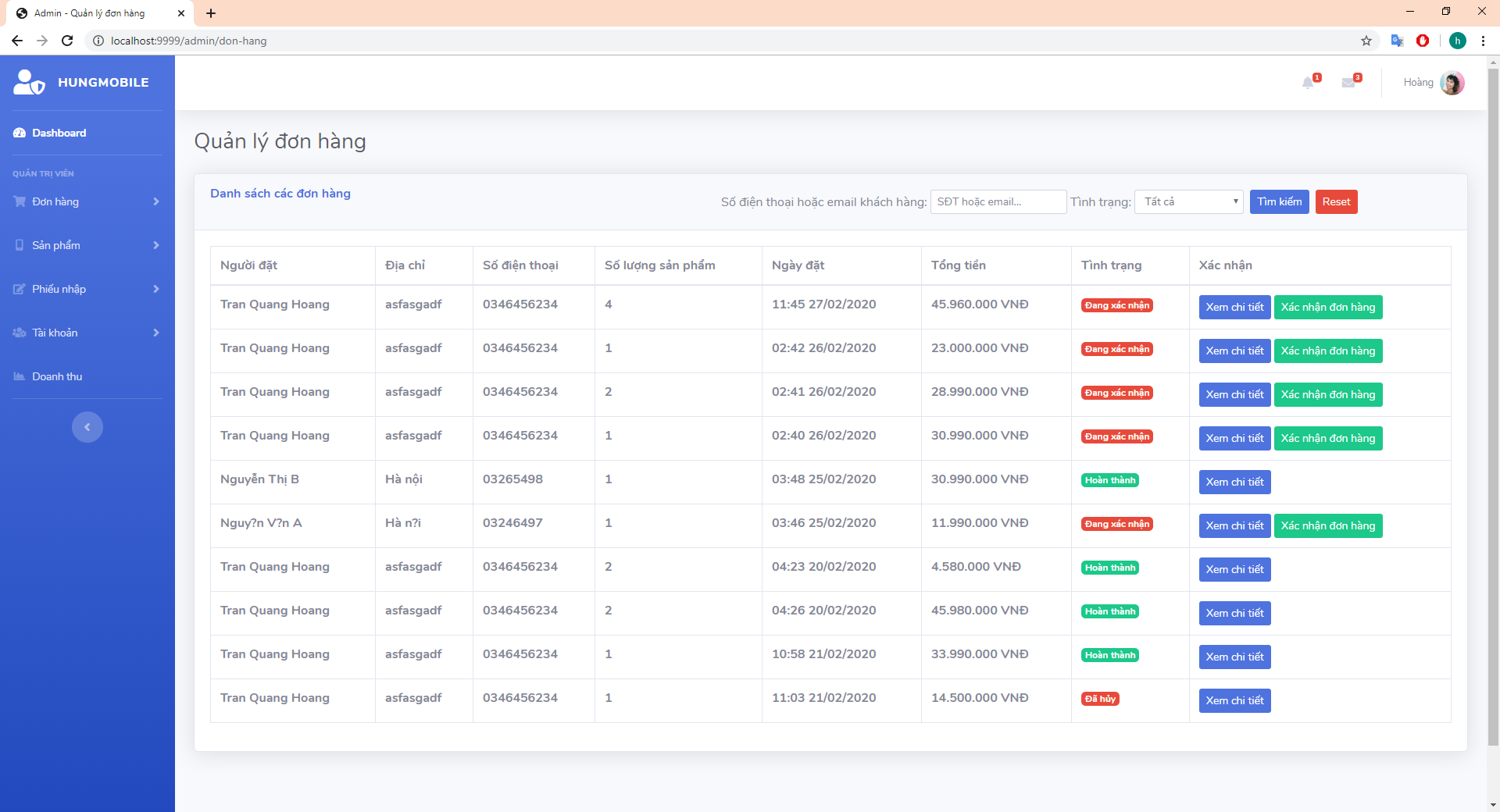
#### Đăng nhập

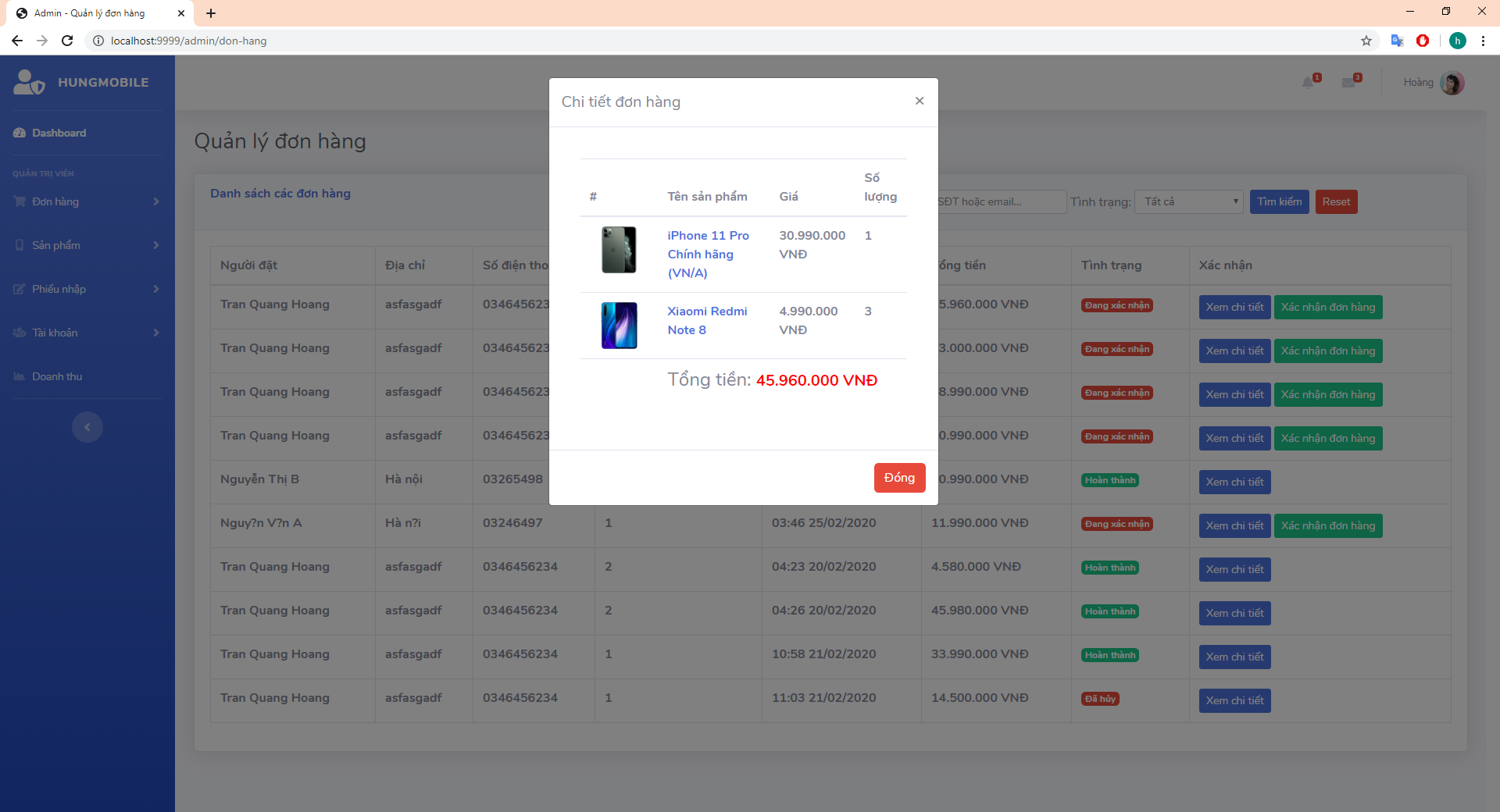


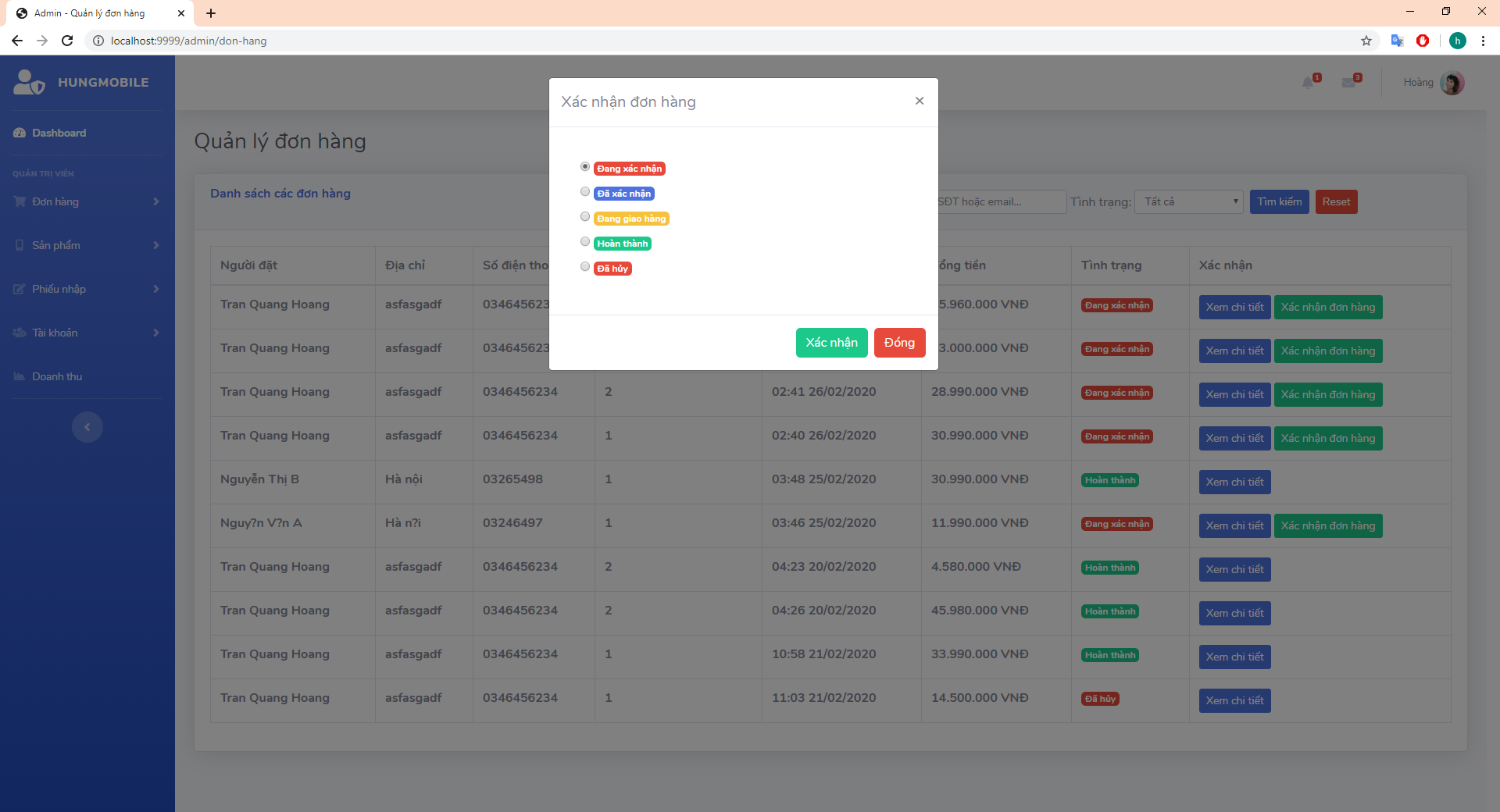
#### Trang chính



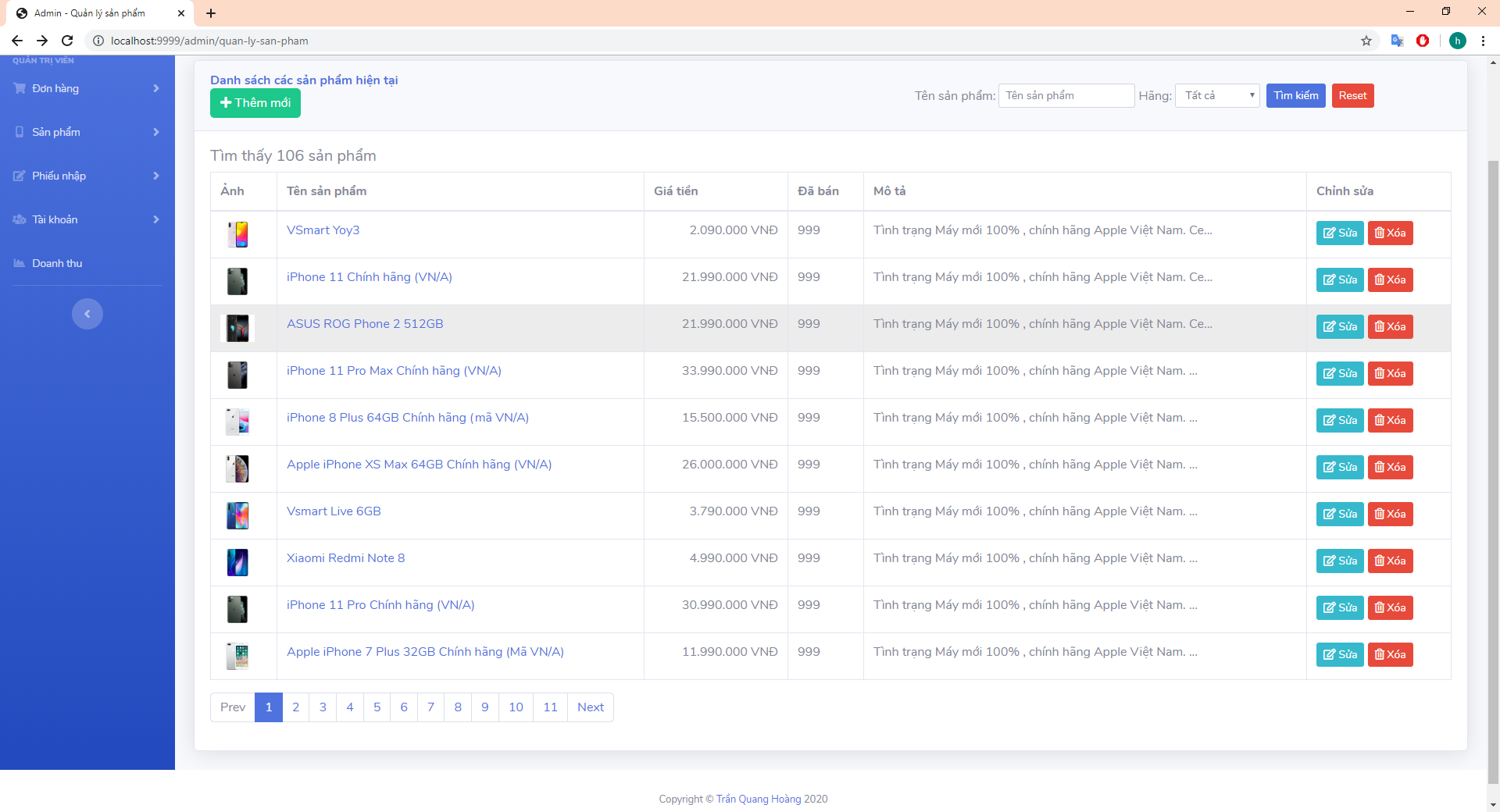
#### Xử lý đơn hàng



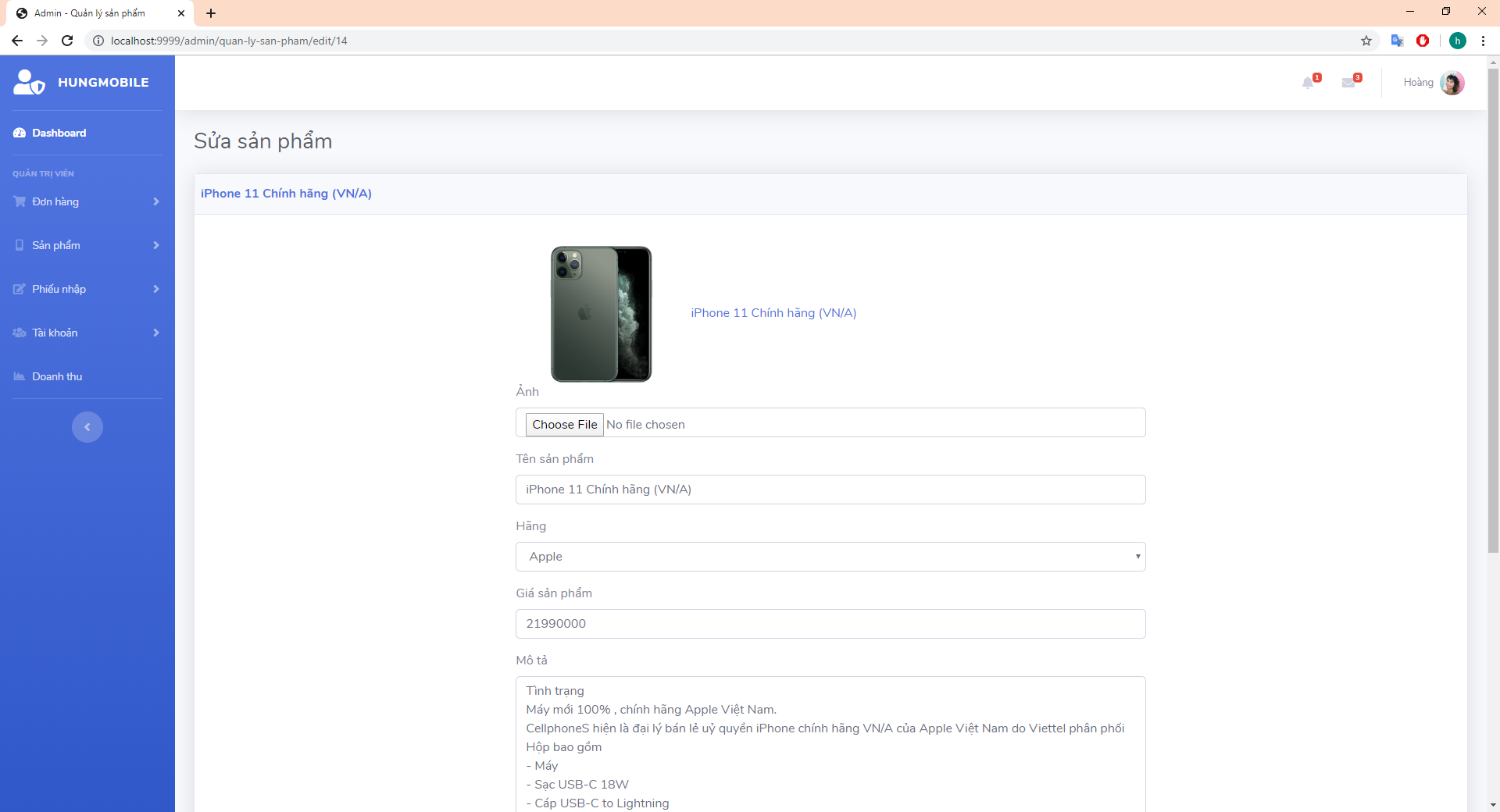




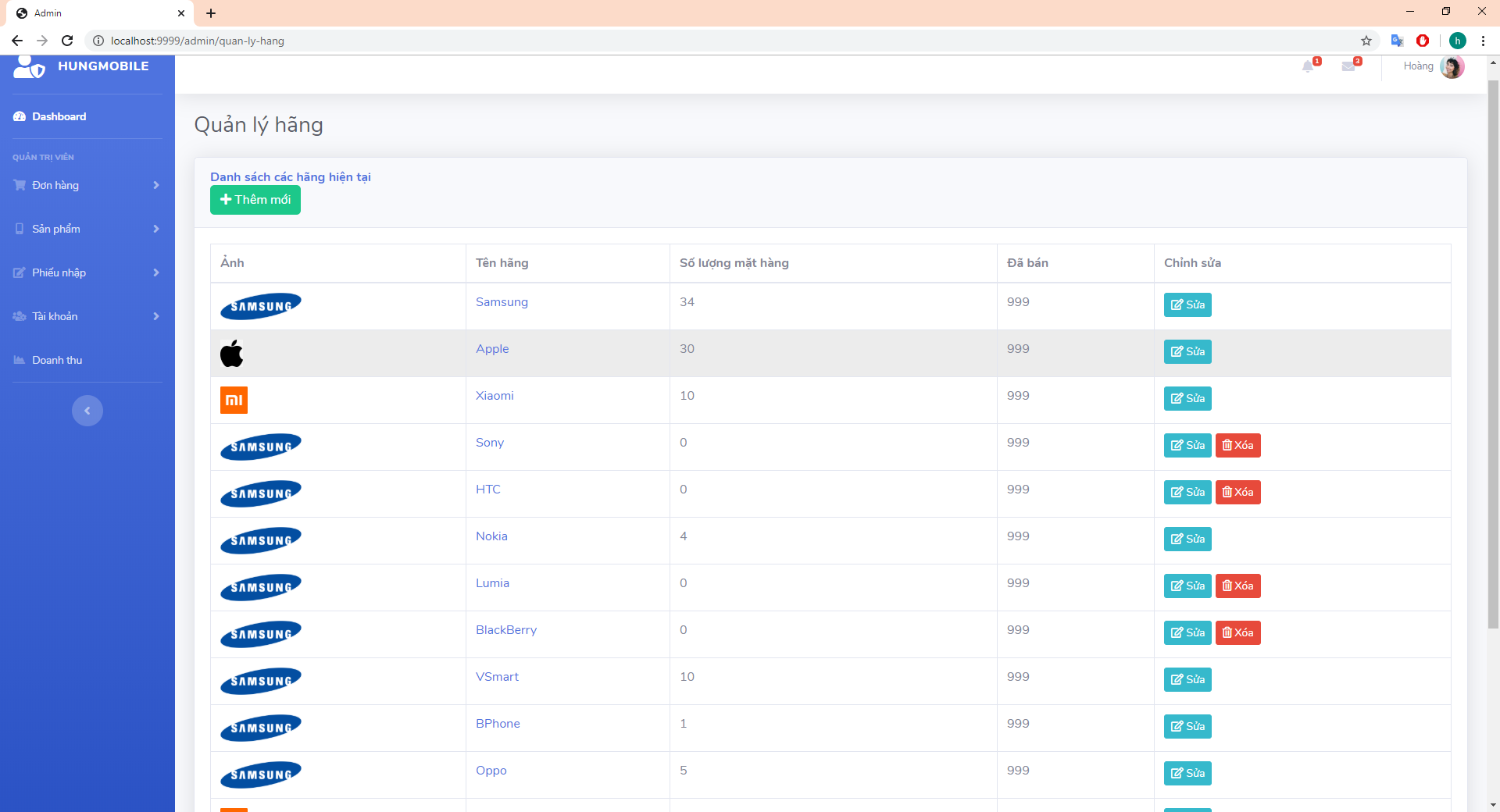
#### Quản lý sản phẩm

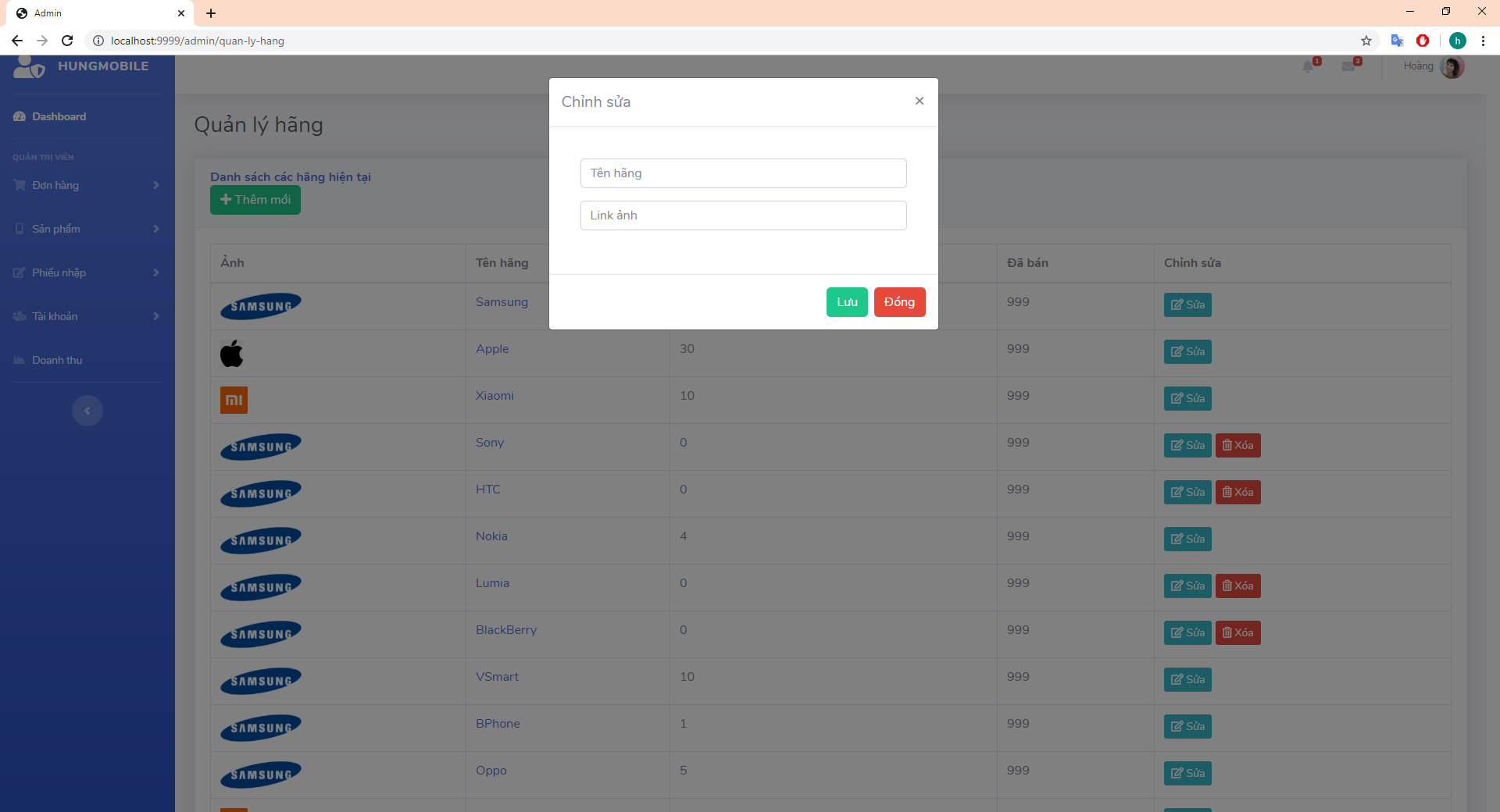


* Sửa sản phẩm

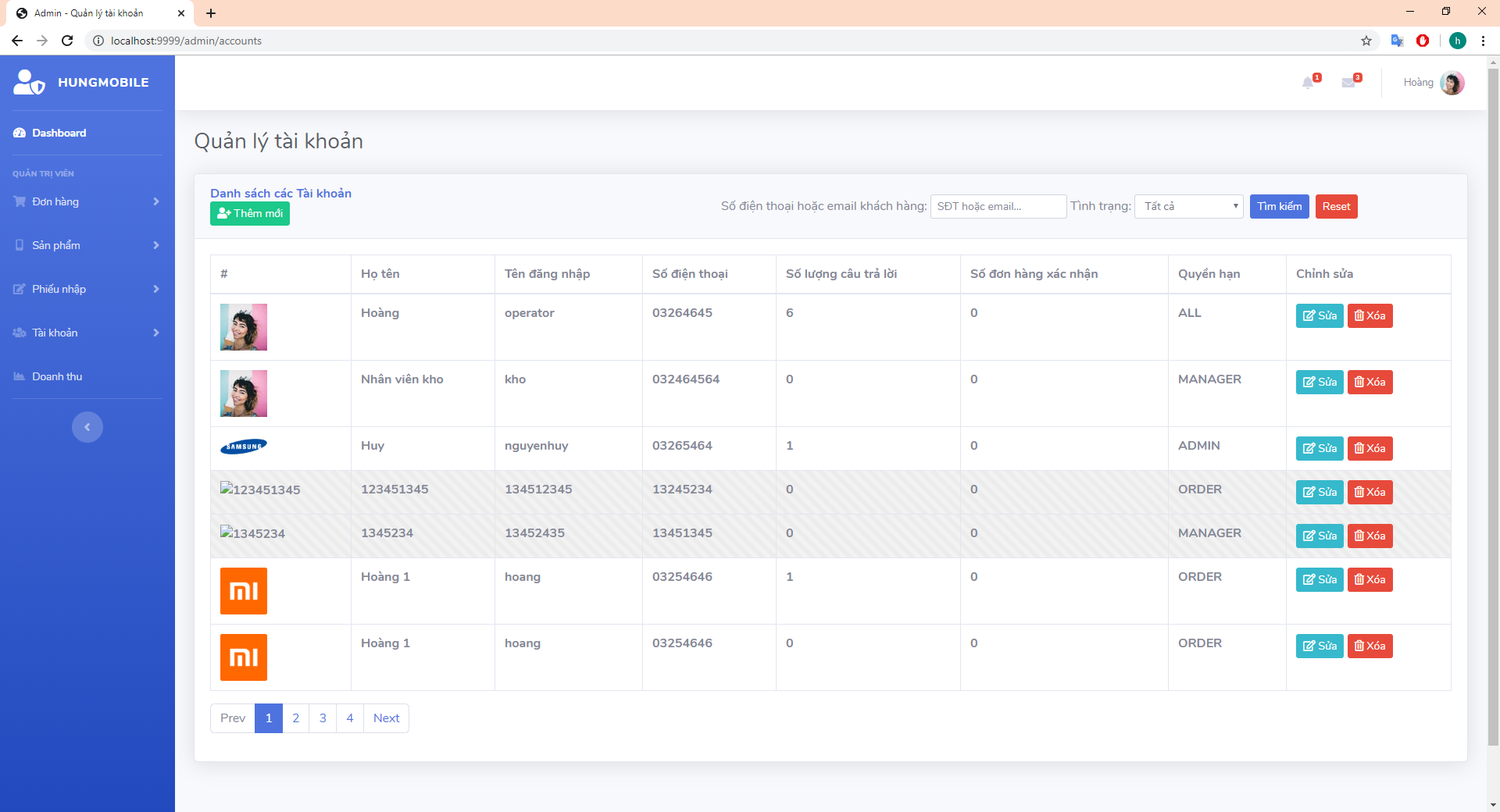


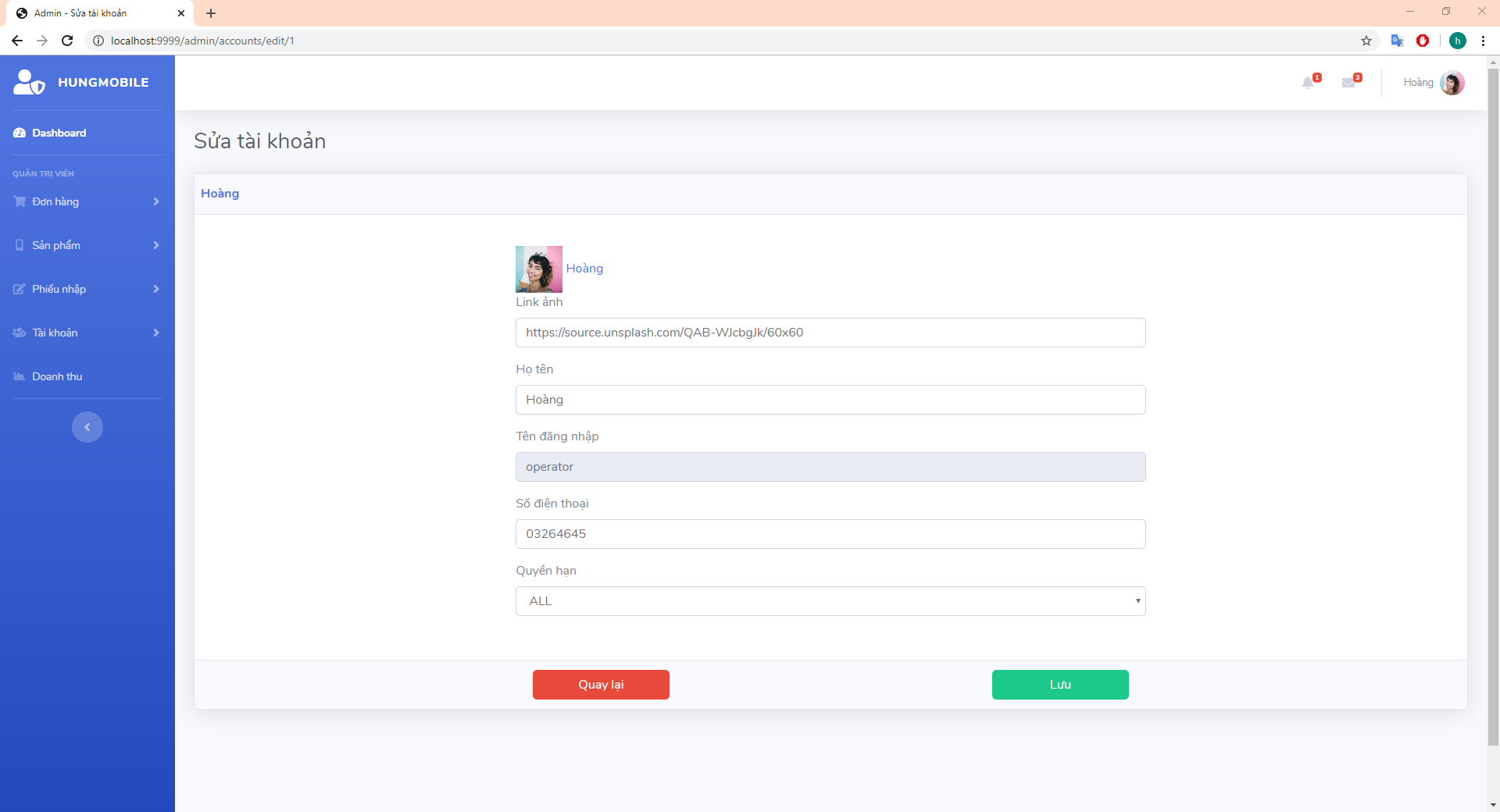
#### Quản lý hãng



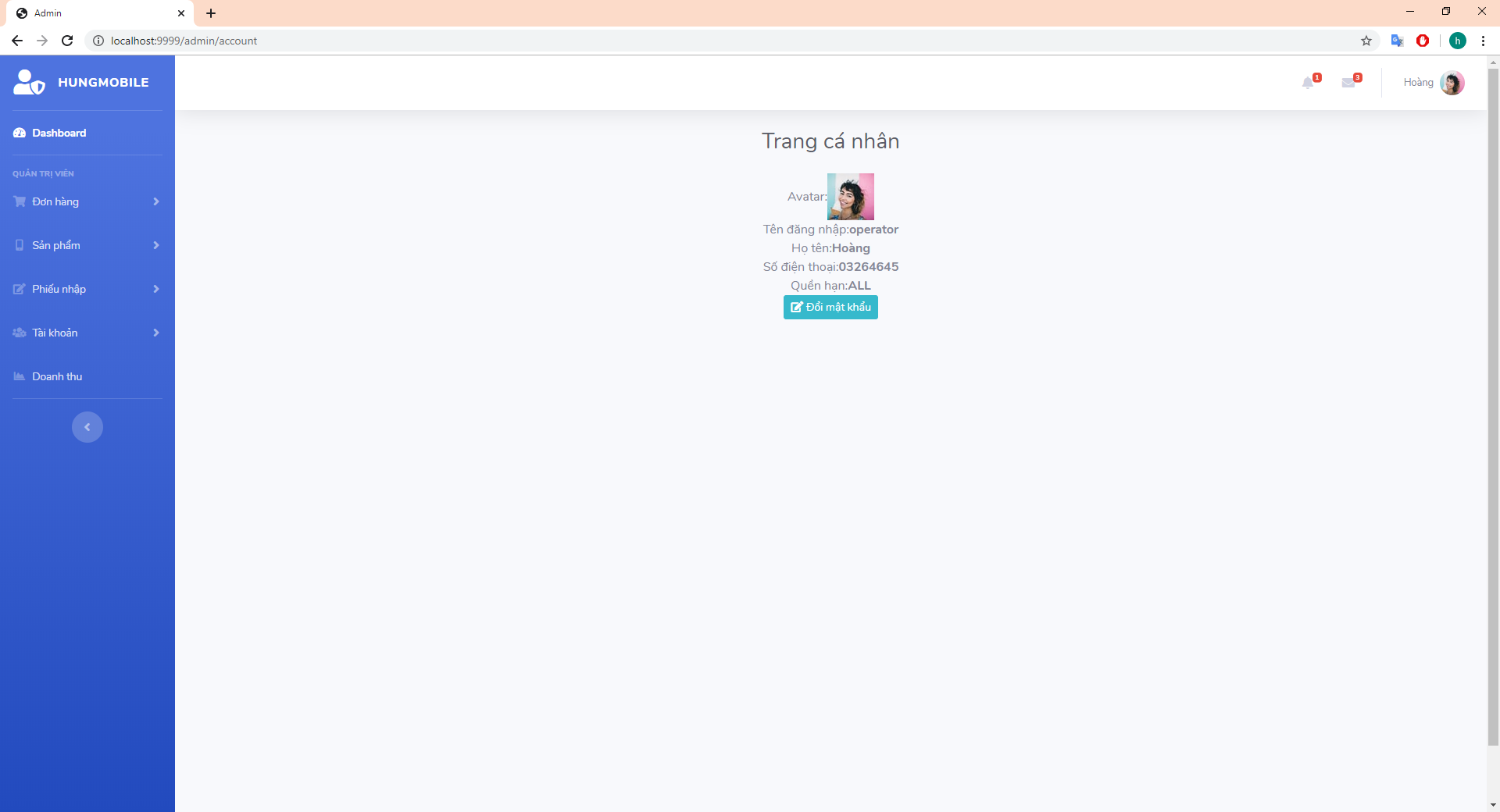


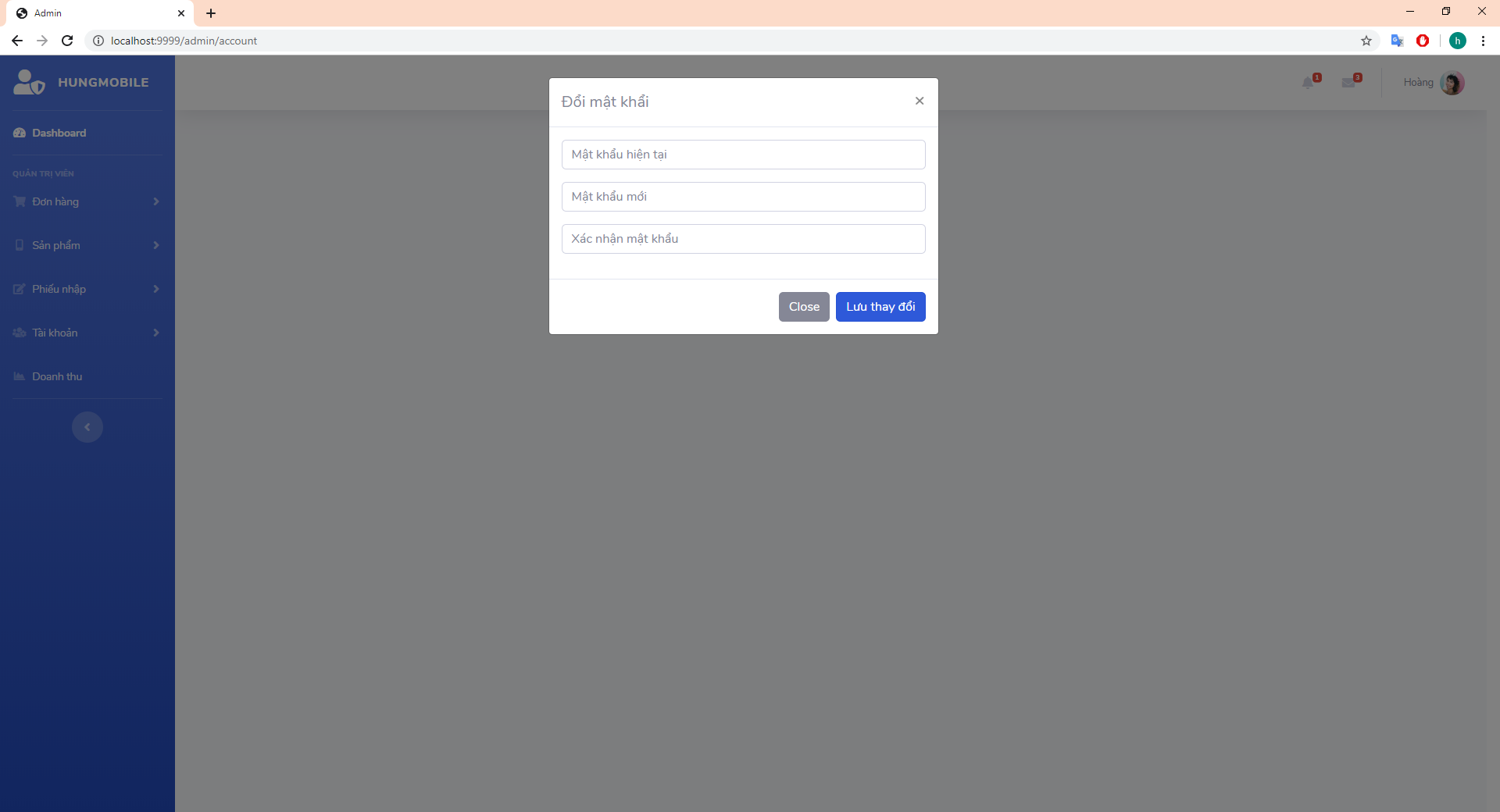
#### Quản lý tài khoản





#### Trang cá nhân





# Tài liệu tham khảo

* 1. [*https://spring.io/*](https://spring.io/)
  2. [*https://startbootstrap.com/themes/admin-dashboard/*](https://startbootstrap.com/themes/admin-dashboard/)
  3. [*https://vdodata.vn/huong-dan-cai-dat-mysql-server/*](https://vdodata.vn/huong-dan-cai-dat-mysql-server/)
  4. [*https://www.tutorialspoint.com/spring\_boot/index.htm*](https://www.tutorialspoint.com/spring_boot/index.htm)