**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

LÊ VĂN DŨNG MSSV:N17DCCN020 Xây dựng website bán thiết bị di động Lớp:D17CQCP01-N 2017-2022

**Gáy**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

------------------------------



**MÔN HỌC**

**KỸ THUẬT LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**

***Đề tài*: “Xây dựng website bán thiết bị di động”**

**Giảng viên : THS. HUỲNH TRUNG TRỤ**

**Sinh viên thực hiện : LÊ VĂN DŨNG**

**Mã số sinh viên : N17DCCN020**

**Lớp : D17CQCP01-N**

**Khoá** **: 2017 - 2022**

**Hệ** **: CHÍNH QUY**

TP. HCM

2021

**TP.HCM, tháng 12/2022**

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**MÔN HỌC**

**KỸ THUẬT LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**

***Đề tài: “*Xây dựng website bán thiết bị di động*”***

**Giảng Viên : THS. HUỲNH TRUNG TRỤ**

**Sinh viên thực hiện : LÊ VĂN DŨNG**

**Mã số sinh viên : N17DCCN020**

**Lớp : D17CQCP01-N**

**Khoá** **: 2017 - 2022**

**Hệ** **: CHÍNH QUY**

**TP.HCM, tháng 12/2022**

**LỜI CẢM ƠN**

Thời gian học tập vừa qua được học dưới mái trường Học Viện Công Nghệ Bưu Chính Viễn Thông cơ sở tại TP.Hồ Chí Minh của bản thân em đã đến chặng đường cuối cùng. Dưới sự dẫn dắt của quý thầy cô mà em dần hoàn thiện dần từ kiến thức kinh nghiệm lẫn tính cách. Với quá trình học tập bản thân em rất thích làm về website, do đó em quyết định làm về thương mại điện tử cụ thể ở đây là trang web bán thiết bị di động.

Dưới sự dẫn chu đáo và hướng dẫn tận tình của thầy Huỳnh Trung Trụ, Em đã dễ dàng thực hiện được những yêu cầu của đề tài.

Em xin chân thành cảm ơn toàn thể thầy cô nói chung và thầy Trụ nói riêng đã tận tình hướng dẫn, chỉ bảo để em hoàn thành tốt lần thực tập này.

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

Thành phố Hồ Chí Minh, tháng 12 năm 2022

Sinh viên thực hiện

**Lê Văn Dũng**

# 

# **MỤC LỤC**

[MỤC LỤC 1](#_Toc81480828)

[DANH MỤC CÁC KÝ HIỆU VÀ CHỮ VIẾT TẮT 4](#_Toc81480829)

[DANH MỤC CÁC BẢNG VẼ 5](#_Toc81480830)

[DANH MỤC CÁC HÌNH VẼ 6](#_Toc81480831)

[MỞ ĐẦU 7](#_Toc81480832)

[CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI 8](#_Toc81480833)

[1.1 Tổng quan 8](#_Toc81480834)

[1.1.1 Lý do chọn đề tài 8](#_Toc81480835)

[1.1.2 Mục tiêu của đề tài 8](#_Toc81480836)

[1.1.3 Giới hạn và phạm vi của đề tài 8](#_Toc81480837)

[1.2 Tìm hiểu website bán thiết bị di động 8](#_Toc81480838)

[1.3 Công cụ sử dụng 9](#_Toc81480839)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT CỦA ĐỀ TÀI 10](#_Toc81480840)

[2.1 Giới thiệu về website 10](#_Toc81480841)

[2.2 Giới thiệu về xây dựng Front-End 10](#_Toc81480842)

[2.2.1 HTML, CSS, JS, Bootstrap 10](#_Toc81480843)

[2.2.2 Giới thiệu về framework ReactJS 10](#_Toc81480844)

[a. React là gì 10](#_Toc81480845)

[b. Thành phần chính của ReactJS 11](#_Toc81480846)

[c. Lợi ích khi sử dụng ReactJS 11](#_Toc81480847)

[d. Tương lai của ReactJS 11](#_Toc81480848)

[2.3 Giới Thiệu Về Xây dựng BackEnd 11](#_Toc81480849)

[2.3.1 JAVA 11](#_Toc81480850)

[a. Khái niệm JAVA 11](#_Toc81480851)

[b. Những ứng dụng nên viết bằng JAVA 12](#_Toc81480852)

[c. Những kết luận sai lầm về JAVA 12](#_Toc81480853)

[d. Lý do nên sử dụng JAVA 12](#_Toc81480854)

[2.3.2 Framework SpringBoot 12](#_Toc81480855)

[a. Định nghĩa SpringBoot 12](#_Toc81480856)

[b. Lợi ích của SpringBoot 12](#_Toc81480857)

[2.4 Giới thiệu về Restful Api 13](#_Toc81480858)

[2.4.1 Định nghĩa Restful Api là gì 13](#_Toc81480859)

[2.4.2 Các thành phần của Restful Api 13](#_Toc81480860)

[2.4.3 Cách hoạt động Restful Api 14](#_Toc81480861)

[2.4.4 Ưu điểm Restful Api 14](#_Toc81480862)

[2.5 Giới thiệu về MySQL 14](#_Toc81480863)

[2.5.1 MySQL là gì 14](#_Toc81480864)

[2.5.2 Ưu điểm của MySQL 15](#_Toc81480865)

[2.5.3 Nhược điểm của MySQL 15](#_Toc81480866)

[2.6 Giới thiệu về công cụ hỗ trợ lập trình 16](#_Toc81480867)

[2.6.1 Visual Studio Code 16](#_Toc81480868)

[2.6.2 Postman 16](#_Toc81480869)

[2.6.3 Redux Dev Tool 17](#_Toc81480870)

[2.6.4 Xampp 17](#_Toc81480871)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 18](#_Toc81480872)

[3.1 Khảo sát hệ thống bán thiết bị di động 18](#_Toc81480873)

[3.2 Các chức năng của hệ thống 18](#_Toc81480874)

[3.2.1 Các đối tượng sử dụng 18](#_Toc81480875)

[3.2.2 Các yêu cầu về chức năng 19](#_Toc81480876)

[a. Đối với Admin 19](#_Toc81480877)

[b. Đối với khách hàng 19](#_Toc81480878)

[c. Khác 20](#_Toc81480879)

[3.3 Đặc tả Use-Case 21](#_Toc81480880)

[3.3.1 Admin 21](#_Toc81480881)

[3.3.2 Khách hàng 21](#_Toc81480882)

[3.4 Phân tích cơ sở dữ liệu 22](#_Toc81480883)

[3.4.1 Bảng User 22](#_Toc81480884)

[3.4.2 Bảng Category 23](#_Toc81480885)

[3.4.3 Bảng Product 23](#_Toc81480886)

[3.4.4 Bảng Cart 23](#_Toc81480887)

[3.4.5 Bảng page 24](#_Toc81480888)

[3.4.6 Bảng order 24](#_Toc81480889)

[3.4.7 Bảng UserAddress 24](#_Toc81480890)

[CHƯƠNG 4: MINH HỌA SẢN PHẨM ĐỀ TÀI 26](#_Toc81480891)

[4.1 Giao diện Admin 26](#_Toc81480892)

[4.1.1 Trang đăng nhập 26](#_Toc81480893)

[4.1.2 Trang đăng ký 26](#_Toc81480894)

[4.1.3 Quản lý User khách hàng 26](#_Toc81480895)

[4.1.4 Tạo Tài Khoản Khách Hàng 27](#_Toc81480896)

[4.1.5 Xem Số lượng đơn hàng của 1 khách hàng 27](#_Toc81480897)

[4.1.6 Quản Lý Danh Mục 28](#_Toc81480898)

[4.1.7 Thêm Danh Mục 28](#_Toc81480899)

[4.1.8 Sửa Danh Mục 29](#_Toc81480900)

[4.1.9 Xóa Danh Mục 29](#_Toc81480901)

[4.1.10 Quản lý sản phẩm 30](#_Toc81480902)

[4.1.11 Thêm mới sản phẩm 30](#_Toc81480903)

[4.1.12 Sửa Sản Phẩm 31](#_Toc81480904)

[4.1.13 Xóa Sản Phẩm 31](#_Toc81480905)

[4.1.14 Chi tiết sản phẩm 32](#_Toc81480906)

[4.1.15 Quản lý đơn hàng 32](#_Toc81480907)

[4.1.16 Chi tiết đơn hàng 33](#_Toc81480908)

[4.1.17 Xác nhận đơn hàng 33](#_Toc81480909)

[4.1.18 Tạo mới Page giao diện 34](#_Toc81480910)

[4.2 Giao diện người dùng 34](#_Toc81480911)

[4.2.1 Xem sản phẩm theo giá 34](#_Toc81480912)

[4.2.2 Xem sản phẩm theo danh mục 35](#_Toc81480913)

[4.2.3 Đăng nhập người dùng 35](#_Toc81480914)

[4.2.4 Đăng ký người dùng 36](#_Toc81480915)

[4.2.5 Chi tiết sản phẩm 36](#_Toc81480916)

[4.2.6 Giỏ hàng 37](#_Toc81480917)

[4.2.7 Thông tin thanh toán 37](#_Toc81480918)

[4.2.8 Xác nhận địa chỉ 38](#_Toc81480919)

[4.2.9 Thêm mới địa chỉ 38](#_Toc81480920)

[4.2.10 Chỉnh sửa địa chỉ 39](#_Toc81480921)

[4.2.11 Xem lại đơn hàng 39](#_Toc81480922)

[4.2.12 Chọn phương thức thanh toán 40](#_Toc81480923)

[4.2.13 Xem đơn hàng đã đặt 40](#_Toc81480924)

[4.2.14 Xem chi tiết đơn hàng 41](#_Toc81480925)

[KẾT LUẬN 42](#_Toc81480926)

[1. Kết quả đạt được 42](#_Toc81480927)

[2. Kết quả chưa đạt được 42](#_Toc81480928)

[DANH MỤC TÀI LIỆU THAM KHẢO 43](#_Toc81480929)

# **DANH MỤC CÁC KÝ HIỆU VÀ CHỮ VIẾT TẮT**

API - Application Programming Interface – giao diện lập trình ứng dụng

# **DANH MỤC CÁC BẢNG VẼ**

[3.4.1 Bảng User 22](#_Toc81480884)

[3.4.2 Bảng Category 23](#_Toc81480885)

[3.4.3 Bảng Product 23](#_Toc81480886)

[3.4.4 Bảng Cart 23](#_Toc81480887)

[3.4.5 Bảng page 24](#_Toc81480888)

[3.4.6 Bảng order 24](#_Toc81480889)

[3.4.7 Bảng UserAddress 24](#_Toc81480890)

# **DANH MỤC CÁC HÌNH VẼ**

[4.1 Giao diện Admin 26](#_Toc81480892)

[4.1.1 Trang đăng nhập 26](#_Toc81480893)

[4.1.2 Trang đăng ký 26](#_Toc81480894)

[4.1.3 Quản lý User khách hàng 26](#_Toc81480895)

[4.1.4 Tạo Tài Khoản Khách Hàng 27](#_Toc81480896)

[4.1.5 Xem Số lượng đơn hàng của 1 khách hàng 27](#_Toc81480897)

[4.1.6 Quản Lý Danh Mục 28](#_Toc81480898)

[4.1.7 Thêm Danh Mục 28](#_Toc81480899)

[4.1.8 Sửa Danh Mục 29](#_Toc81480900)

[4.1.9 Xóa Danh Mục 29](#_Toc81480901)

[4.1.10 Quản lý sản phẩm 30](#_Toc81480902)

[4.1.11 Thêm mới sản phẩm 30](#_Toc81480903)

[4.1.12 Sửa Sản Phẩm 31](#_Toc81480904)

[4.1.13 Xóa Sản Phẩm 31](#_Toc81480905)

[4.1.14 Chi tiết sản phẩm 32](#_Toc81480906)

[4.1.15 Quản lý đơn hàng 32](#_Toc81480907)

[4.1.16 Chi tiết đơn hàng 33](#_Toc81480908)

[4.1.17 Xác nhận đơn hàng 33](#_Toc81480909)

[4.1.18 Tạo mới Page giao diện 34](#_Toc81480910)

[4.2 Giao diện người dùng 34](#_Toc81480911)

[4.2.1 Xem sản phẩm theo giá 34](#_Toc81480912)

[4.2.2 Xem sản phẩm theo danh mục 35](#_Toc81480913)

[4.2.3 Đăng nhập người dùng 35](#_Toc81480914)

[4.2.4 Đăng ký người dùng 36](#_Toc81480915)

[4.2.5 Chi tiết sản phẩm 36](#_Toc81480916)

[4.2.6 Giỏ hàng 37](#_Toc81480917)

[4.2.7 Thông tin thanh toán 37](#_Toc81480918)

[4.2.8 Xác nhận địa chỉ 38](#_Toc81480919)

[4.2.9 Thêm mới địa chỉ 38](#_Toc81480920)

[4.2.10 Chỉnh sửa địa chỉ 39](#_Toc81480921)

[4.2.11 Xem lại đơn hàng 39](#_Toc81480922)

[4.2.12 Chọn phương thức thanh toán 40](#_Toc81480923)

[4.2.13 Xem đơn hàng đã đặt 40](#_Toc81480924)

[4.2.14 Xem chi tiết đơn hàng 41](#_Toc81480925)

# **MỞ ĐẦU**

Giữa làn sóng chuyển lên các kênh trực tuyến để giảm thiểu tác động từ Covid-19 và phục hồi kinh doanh, Website thương mại điện tử bán hàng nổi lên như một “điểm bán” đặc biệt hiệu quả đối với các doanh nghiệp và nhà bán hàng. Theo các chuyên gia phân tích thị trường từ Vaimo, tiếp nối đà này và xu hướng D2C (tiếp cận trực tiếp khách hàng) ngày một được ưa chuộng, Website của riêng thương hiệu sẽ còn phát triển và có nhiều vai trò hơn trong tương lai.

Điểm khác biệt của Website Thương mại điện tử dành cho bán hàng: Thao tác thuận tiện, nhanh chóng phục vụ cho việc mua sắm, sở hữu tính năng giỏ hàng, sở hữu chức năng thanh toán, chức năng quản lý và báo cáo kinh doanh, dễ dàng tương tác với người bán, hình ảnh và video trực quan, chất lượng

Và điều quan trọng trang thương mại điện tử sẽ khiến gia tăng lợi nhuận của người bán hàng và nâng cáo tính cạnh tranh với đối thủ

Từ những điều trên, em đã chọn đề tài là “Xây dựng website bán thiết bị di động”

1. GIỚI THIỆU ĐỀ TÀI
   1. Tổng quan
      1. Lý do chọn đề tài

Ngày nay với sự phát triển mạnh mẽ về cuộc cách mạng công nghệ, đã và đang phát triển, người bán hàng nắm bắt được thị trường, mở rộng thị trường bằng cách tạo những website để tiếp cận khách hàng với quy mô rộng lớn. nhằm tăng chỉ số bán hàng, tạo thu nhập cao hơn. Hiện nay đa số đã tiếp cận được với internet và điều đó giúp khách hàng mua sản phẩm nhanh gọn hơn bằng 1 cú click chuột. Vì vậy em chọn đề tài “Xây dựng website bán thiết bị di động” để khách hàng dễ dàng mua sắm online, phục vụ cho việc kinh doanh, mua bán thiết bị di động online

* + 1. Mục tiêu của đề tài

Xây dựng được website bán thiết bị di động, các nghiệp vụ thương mại điện tử

Tạo ra giao diện dễ sử dụng của cả người quản lý bán hàng và khách hàng

Khách hàng có thể xem sản phẩm, mua hàng trực tuyến và theo dõi tình trạng đơn hàng của họ

Quản lý có thể quản lý được các sản phẩm, khách hàng của cửa hàng

Triển khai được website vào thực tế sử dụng

* + 1. Giới hạn và phạm vi của đề tài

Đề tài sử dụng các công nghệ lập trình như:

- ReactJs

- Java SpringBoot

- HTML, CSS, Bootstrap

- MongoDB

* 1. Tìm hiểu website bán thiết bị di động

1. Lý Thuyết

- Tìm hiểu nghiệp vụ thương mại điện tử

- Tìm hiểu các ngôn ngữ lập trình ReactJS, Redux, Restful API, SpringBoot, HTML, CSS, Bootstrap

- Hệ quản trị dữ liệu MySQL

1. Thực Hành

- Phân tích và thiết kế cơ sở dữ liệu

- Xây dựng giao diện bán thiết bị di động

- Đăng Nhập, Đăng Ký tài khoản

- Thêm, sửa, xóa giỏ hàng

- Trạng thái đơn hàng( chờ xác nhận, đang giao, hoàn tất đơn hàng )

- Phân trang, tìm kiếm sản phẩm

- Thêm xóa sửa sản phẩm (admin)

- Thêm xóa sửa danh mục (admin)

- Xác nhận đơn hàng(admin)

- Quản lý tài khoản user(admin)

* 1. Công cụ sử dụng

- Visual Studio Code

- Redux dev tool

- Postman

- IntelliJ IDEA

- MySQL

1. CƠ SỞ LÝ THUYẾT CỦA ĐỀ TÀI
   1. Giới thiệu về website

Website là một trang thông tin với mục đích là để giới thiệu, cập nhật những thông tin về các doanh nghiệp, sản phẩm, hoạt động cũng như tin tức, chia sẻ bí quyết,… để phát triển thương hiệu. Từ những thông tin này, mục tiêu lớn hơn mà không ít các doanh nghiệp muốn hướng đến khi xây dựng website là để bán hàng.

Thông thường, website được phân chia thành 4 hình thức chủ yếu theo đối tượng như website dành cho doanh nghiệp, website dành cho cá nhân, website của chính phủ và website của các tổ chức phi lợi nhuận. Căn cứ theo mục đích khác nhau của website mà web sẽ có những đặc điểm khác nhau. Tuy nhiên, nhìn chung tầm quan trọng của website là như nhau.

* 1. Giới thiệu về xây dựng Front-End
     1. HTML, CSS, JS, Bootstrap

HTML là chữ viết tắt của cụm từ Hypertext Markup Language (là ngôn ngữ đánh dấu siêu văn bản) được sử dụng để tạo một trang web, trên một website có thể sẽ chứa nhiều trang và mỗi trang được quy ra là một tài liệu HTML. Cho dù bạn lập trình bằng bất kì ngôn ngữ gì, bất kì framework nào thì khi chạy trên nền web nó đều biên dịch ra ngôn ngữ HTML.

CSS là từ viết tắt của Cascade Style Sheet, là ngôn ngữ giúp trình duyệt hiểu được các thiết lập định dạng và bố cục cho trang web. CSS cho phép bạn điều khiển thiết kế của nhiều thành phần HTML chỉ với duy nhất 1 vùng chọn CSS. Điều này giúp giảm thiểu thời gian thiết kế và chỉnh sửa, khi bạn có thể tách biệt được cấu trúc (HTML) và định dạng (CSS).

Javascript là một ngôn ngữ lập trình kịch bản dựa vào đối tượng phát triển có sẵn hoặc tự định nghĩa ra, Javascript được sử dụng rộng rãi trong các ứng dụng Website.

Javascript được hỗ trợ hầu như trên tất cả các trình duyệt như Firefox, Chrome, … thậm

chí các trình duyệt trên thiết bị di động cũng có hỗ trợ. Ngày nay, Javascript được sử dụng rộng rãi trong hầu hết tất cả các website

Bootstrap là một framework bao gồm các HTML, CSS và JavaScript template dùng để phát triển website chuẩn responsive. Bootstrap cho phép quá trình thiết kế website diễn ra nhanh chóng và dễ dàng hơn dựa trên những thành tố cơ bản sẵn có như typography, forms, buttons, tables, grids, navigation, image carousels…

* + 1. Giới thiệu về framework ReactJS

React là gì

ReactJS được hiểu nôm na là một thư viện trong đó có chứa nhiều JavaScript mã nguồn mở và cha đẻ của ReactJS đó chính là một ông lớn với cái tên ai cũng biết đó chính là Facebook. Mục đích của việc tạo ra ReactJS là để tạo ra những ứng dụng website hấp dẫn với tốc độ nhanh và hiệu quả cao với những mã hóa tối thiểu. Và mục đích chủ chốt của ReactJS đó chính là mỗi website khi đã sử dụng ReactJS thì phải chạy thật mượt thật nhanh và có khả năng mở rộng cao và đơn giản thực hiện.

Thành phần chính của ReactJS

Redux: Đây có thể gọi là một phần cực kỳ quan trọng đối với ReactJS và không một ai sử dụng mà không biết đến redux. Trong một reactJS không bao gồm những module chuyên dụng để xử lý dữ liệu vì thế ReactJS được thiết lập một cách độc lập bằng việc chia nhỏ view thành các component nhỏ để chúng liên kết chặt chẽ với nhau hơn

Lợi ích khi sử dụng ReactJS

Lợi ích đầu tiên mà ReactJS đó chính là việc tạo ra cho chính bản thân nó một dom ảo, đây là nơi mà các component được tồn tại trên đó

Lợi ích thứ hai mà ReactJS đem lại đó chính là việc viết các đoạn code JS sẽ trở nên dễ dàng hơn vì nó sử dụng một cú pháp đặc biệt đó chính là cú pháp JSX nghĩa là cú pháp này cho phép ta trộn được giữa code HTML và Javascript

Thân thiện với SEO: đây là một trong những điều đặc biệt và chỉ có mỗi ReactJS đây cũng chính là vấn đề lớn của các JS Frameworks vì hầu như các JS Frameworks không thân thiện với các cổ máy tìm kiếm mặc dù đã được cải thiện nhiều vì dưới sự hỗ trợ của các render và trả về trình duyệt dưới dạng webpage khi mà bạn chạy ReactJS trên server và các Virtual Dom

Tương lai của ReactJS

Có thêm những loại render mới với các chức năng như add thêm những đoạn cú pháp mới và độc đáo hơn nhiều vào trong JSX mà không cần đến một keys nào cả

Cải thiện được việc xử lý các lỗi phát sinh còn được gọi là các lỗi trong component mà chúng có thể làm hỏng state của component và cũng gây ra các lỗi trong quá trình render.

* 1. Giới Thiệu Về Xây dựng BackEnd
     1. Java

Khái niệm Java

Những ứng dụng nên viết bằng Java

Những kết luận sai lầm về Java

Lý do nên sử dụng Java

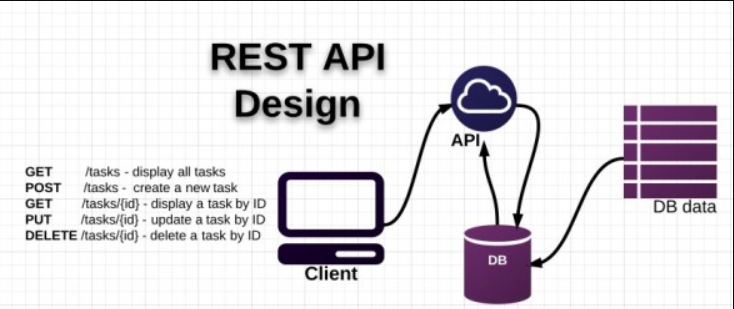
* + 1. Framework SpringBoot

Định nghĩa Java SpringBoot

Lợi ích của Java SpringBoot

* 1. Giới thiệu về Restful Api
     1. Định nghĩa Restful Api là gì

RESTful API là một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web (thiết kế Web services) để tiện cho việc quản lý các resource. Nó chú trọng vào tài nguyên hệ thống (tệp văn bản, ảnh, âm thanh, video, hoặc dữ liệu động…), bao gồm các trạng thái tài nguyên được định dạng và được truyền tải qua HTTP.



Hình 1: Rest Api Design

* + 1. Các thành phần của Restful Api

API (Application Programming Interface) là một tập các quy tắc và cơ chế mà theo đó, một ứng dụng hay một thành phần sẽ tương tác với một ứng dụng hay thành phần khác. API có thể trả về dữ liệu mà bạn cần cho ứng dụng của mình ở những kiểu dữ liệu phổ biến như JSON hay XML.

REST (REpresentational State Transfer) là một dạng chuyển đổi cấu trúc dữ liệu, một kiểu kiến trúc để viết API. Nó sử dụng phương thức HTTP đơn giản để tạo cho giao tiếp giữa các máy. Vì vậy, thay vì sử dụng một URL cho việc xử lý một số thông tin người dùng, REST gửi một yêu cầu HTTP như GET, POST, DELETE, vv đến một URL để xử lý dữ liệu.

RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource. RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile…) khác nhau giao tiếp với nhau.

Chức năng quan trọng nhất của REST là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE…) và cách định dạng các URL cho ứng dụng web để quản các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API.

* + 1. Cách hoạt động Restful Api

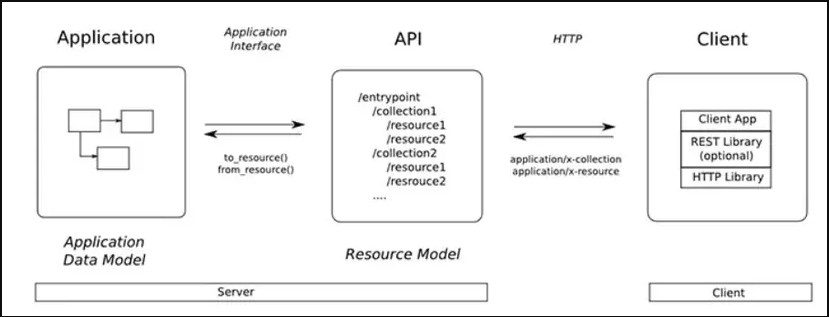
REST hoạt động chủ yếu dựa vào giao thức HTTP. Các hoạt động cơ bản nêu trên sẽ sử dụng những phương thức HTTP riêng.

GET (SELECT): Trả về một Resource hoặc một danh sách Resource.

POST (CREATE): Tạo mới một Resource.

PUT (UPDATE): Cập nhật thông tin cho Resource.

DELETE (DELETE): Xóa một Resource.



Hình 2.4.3: Cách hoạt động của Restful API

* + 1. Ưu điểm Restful Api

- Giúp cho ứng dụng rõ ràng hơn

- REST URL đại diện cho resource chứ không phải hành động

- Dữ liệu được trả về với nhiều định dạng khác nhau như: xml, html, json….

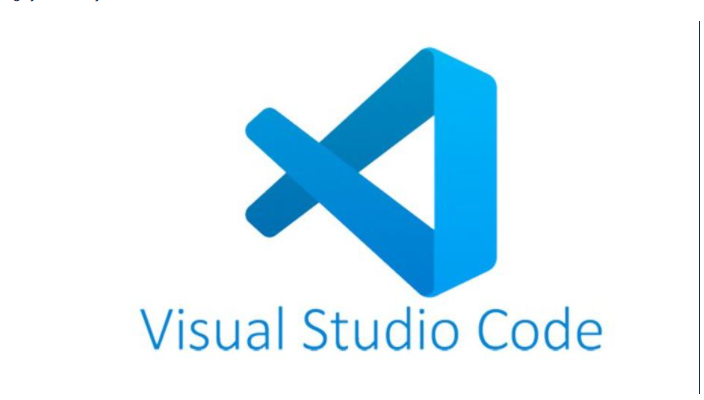
- Code đơn giản và ngắn gọn

- REST chú trọng vào tài nguyên của hệ thống

* 1. Giới thiệu về MySQL
     1. MySQL là gì
     2. Ưu điểm của MySQL
     3. Nhược điểm của MySQL
  2. Giới thiệu về công cụ hỗ trợ lập trình
     1. Visual Studio Code

Là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio Code được phát triển bởi Microsoft. Nó được xem là một sự kết hợp hoàn hảo giữa IDE và Code Editor.

Visual Studio Code hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép người dùng thay đổi theme, phím tắt, và các tùy chọn khác.



Hình 2.6.1: Visual Studio Code

* + 1. Postman

Postman là một công cụ cho phép chúng ta thao tác với API, phổ biến nhất là REST. Postman hiện là một trong những công cụ phổ biến nhất được sử dụng trong thử nghiệm các API. Với Postman, ta có thể gọi Rest API mà không cần viết dòng code nào.

Postman hỗ trợ tất cả các phương thức HTTP (GET, POST, PUT, PATCH, DELETE, …). Bên cạnh đó, nó còn cho phép lưu lại lịch sử các lần request, rất tiện cho việc sử dụng lại khi cần.



Hình 2.6.2: Postman

* + 1. Redux Dev Tool

Nó là 1 extension cài được trên các browser chrome hay firefox nó giúp ra view store của app một cách trực quan nhất mà không cần console, ngoài ra còn xem list các commit, các actions step by step. Giúp chúng ta biết được đường đi nước bước, logic từng màn hình dễ dàng nhất. Sau khi cái xong để kích hoạt Redux DevTools có 2 cách

Cách 1: Bấm vào icon trên tabbar của trình duyệt

Cách 2: Chuột phải vào trang, và chọn Redux DevTools

* + 1. EntelliJ IDEA
    2. Xampp

1. PHÂN TÍCH THIẾT KẾ HỆ THỐNG
   1. Khảo sát hệ thống bán thiết bị di động

* Giao diện website bán hàng đẹp mắt, chuyên nghiệp thân thiện người dùng.
  + Thiết lập kích thước ảnh: Hiển thị nhiều ảnh mô tả cho sản phẩm. Xem ảnh sản phẩm dạng slide show đẹp mắt. Zoom ảnh khi xem sản phẩm
  + Hiển thị khuyến mại, giảm giá
  + Đánh giá, bình luận sản phẩm
  + Thiết lập sản phẩm liên quan
  + Thông tin giá: Giá bán
* Đặt hàng giỏ hàng
  + Cập nhật thông tin giỏ hàng
  + Xóa sản phẩm khỏi giỏ hàng
  + Quy trình thanh toán và đặt hàng mới
  + Tiếp nhận đơn đặt hàng trực tuyến=> Xem thông tin sản phẩm, giá trị đơn hàng=> Xem thông tin người đặt hàng
  + Xác nhận đơn hàng thành công
  + Xác nhận hủy đơn hàng
* Chức năng thanh toán
* Quản lý kho hàng
  + Quản lý danh sách sản phẩm
  + Tùy chọn ẩn/hiển thị danh mục/ sản phẩm
  + Lọc, tìm kiếm sản phẩm theo nhiều tiêu chí
  + Thiết lập trạng thái còn hàng/hết hàng
  + Thêm, sửa, xóa sản phẩm
* Chức năng tìm kiếm
* Tương thích các thiết bị di động thông minh như iphone, ipad và các loại smartphone.
* Thiết kế web bán hàng chuẩn SEO
* Tốc độ tải trang nhanh
* Hệ quản trị đơn giản dễ sử dụng
  1. Các chức năng của hệ thống
     1. Các đối tượng sử dụng
* Admin ( quản lý bán hàng )
* Customer (khách hàng)
  + 1. Các yêu cầu về chức năng

Đối với Admin

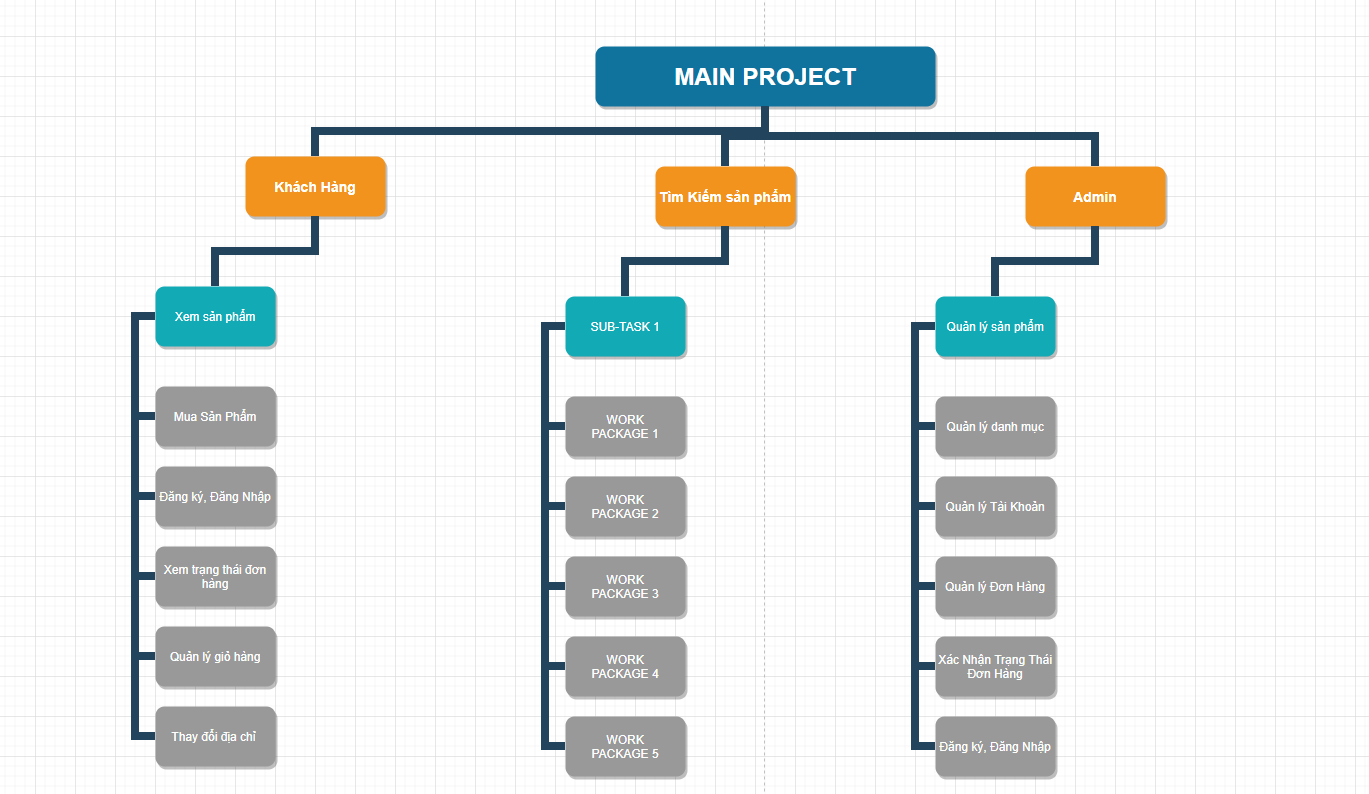
* Quản lý thêm, xóa, sửa Danh mục
  + Khi Danh Mục đã có Sản Phẩm thì không cho xóa
  + Chỉ cho xóa các Danh mục con khi các Danh mục phân cấp
  + Cho phép xóa danh mục cha khi đã xóa hết danh mục con
* Quản lý thêm, xóa, sửa Sản Phẩm
  + Chỉ cho phép xóa Sản Phẩm khi chưa lưu vào đơn hàng
  + Cho phép thêm nhiều hình ảnh cho sản phẩm
* Tạo mới page
  + Thêm banner cho kiểu hiển thị page
* Quản lý khách hàng
  + Tạo mới tài khoản khách hàng
  + Xem số lượng đơn hàng và trạng thái đơn hàng của khách hàng
* Quản lý đơn hàng
  + Xem danh sách các đơn hàng
  + Xác nhận các trạng thái của đơn hàng
* Đăng ký, đăng nhập tài khoản quản trị viên

Đối với khách hàng

* Xem danh sách sản phẩm
  + Xem danh sách sản phẩm theo 3 giao diện
  + Xem sản phẩm theo danh mục
  + Xem sản phẩm theo từng mức giá
* Tìm kiếm sản phẩm
  + Theo danh mục
  + Theo tên sản phẩm
  + Theo mức giá
* Thêm, xóa, sửa giỏ hàng
  + Giỏ hàng sẽ lưu lại nếu chưa thanh toán (đã đăng nhập)
* Xem đơn hàng
  + Xem trạng thái các đơn hàng của minh
* Đặt mua sản phẩm
* Địa chỉ giao hàng
  + Thay đổi địa chỉ giao hàng
  + Thêm mới địa chỉ giao hàng
* Thanh toán mua hàng
  + Chọn phương thức thanh toán
* Đăng nhập, đăng ký tài khoản

Khác

* Cho phép hiển thị 3 kiểu giao diện khi xem sản phẩm theo danh mục
* Xem các thông tin về thiết bị di động
* Hiệu suất: hiệu suất liên quan đến tài nguyên được sử dụng trong các điều kiện xác định được đảm bảo.
* Tính tương thích: thông tin cập nhật chính xác sản phẩm hiện đang bán của cửa hàng.
* Tính khả dụng: ứng dụng có thể hiểu được, học cách sử dụng được và làm hài lòng người sử dụng trong các trường hợp cụ thể.
* Tính tin cậy: có khả năng thực hiện các chức năng với những tham số đầu vào xác định trong một khoảng thời gian xác định và trả lại kết quả chính xác trong phạm vi cho phép.
* Giao diện: trình bày rõ ràng, trực quan, giúp người dùng sử dụng dễ thao tác
* Website xử lý thông tin nhanh, có các phản hồi chính xác
* Thông tin được lưu trữ cần chính xác
* Không bị sập khi thực hiện các thao tác
* Cần phân quyền đúng các nhóm chức năng của từng nhóm user



Hình 3.2.2: Mô hình chức năng

Diagrams

Ảnh có chứa văn bản, ảnh chụp màn hình, trong nhà, máy tính

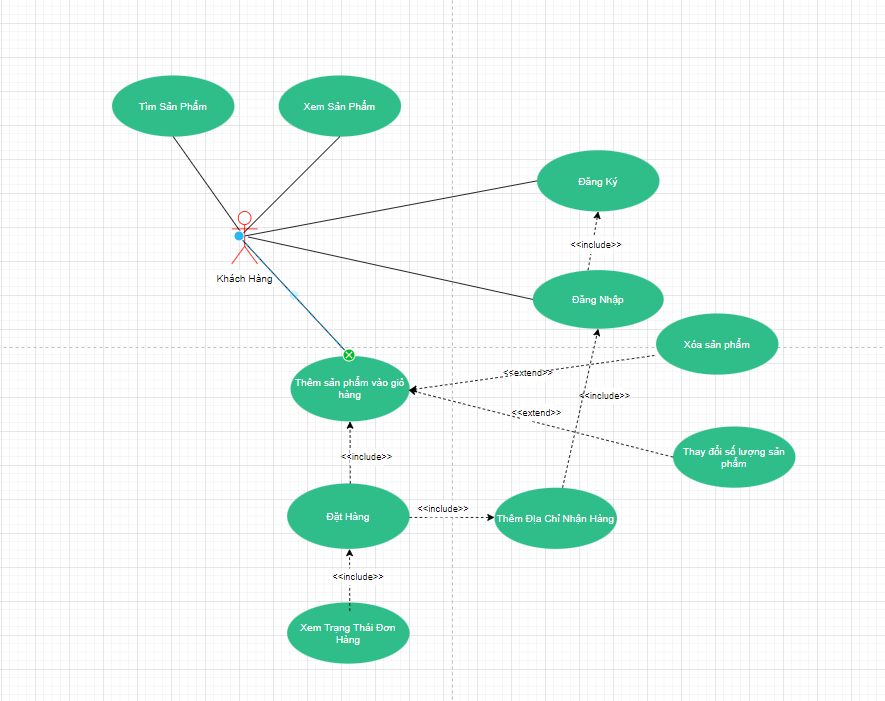
Mô tả được tạo tự động

* 1. Đặc tả Use-Case
     1. Admin



Hình 3.3.1: Mô hình use-case của quản lý bán hàng

* + 1. Khách hàng



Hình 3.3.2: Mô hình use-case của khách hàng

* 1. Phân tích cơ sở dữ liệu

Các bảng đều có 2 field là createdAt và updatedAt do Mongoose tạo sẵn

* + 1. Bảng User

-Dùng để lưu trữ dữ liệu khách hàng và quản lý

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của user | Khóa chính tự tạo |
| 2 | role | String | Phân quyền [User, Admin] | Dạng enum |
| 3 | firstName | String | Tên của user | require |
| 4 | lastName | String | Họ của user | require |
| 5 | email | String | Email đăng nhập | Require,unique |
| 6 | hash\_password | String | Lưu mật khẩu băm | require |
| 7 | username | String | Tên User | Require,unique |

* + 1. Bảng Category
* Dùng để lưu trữ danh mục

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của danh mục | Khóa chinh tự tạo |
| 2 | name | String | Tên danh mục | Require |
| 3 | slug | String | Tên mã hóa của danh mục | Require, unique |
| 4 | parentId | String | ID của danh mục cha |  |
| 5 | type | String | Loại giao diện hiển thị |  |
| 6 | categoryImages | String | Đường dẫn hình ảnh |  |

* + 1. Bảng Product
* Dùng để lưu trữ sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của sản phẩm | Khóa chính tự tạo |
| 2 | name | String | Tên sản phẩm | Require |
| 3 | slug | String | Tên mã hóa của sản phẩm | Require, unique |
| 4 | price | Int32 | Giá sản phẩm | require |
| 5 | quantity | Int32 | Số lượng sản phẩm | require |
| 6 | description | String | Mô tả sản phẩm | require |
| 7 | productImages | Array | Hình ảnh sản phẩm |  |
| 8 | category | ObjectId |  | Khoa ngoại |
| 9 | reviews | Array | Đánh giá sản phẩm |  |

* + 1. Bảng Cart
* Dùng để lưu giỏ hàng khi chưa thanh toán

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của giỏ hàng | Khóa chính tự tạo |
| 2 | user | ObjectId | Id của khách hàng | Khoa ngoại |
| 3 | CartItem | Array | Chứa sản phẩm và số lượng | Có chứa khoá ngoại của product |

* + 1. Bảng page
* Dùng để lưu giỏ hàng khi chưa thanh toán

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của page | Khóa chính tự tạo |
| 2 | title | String | Tiêu đề | require |
| 3 | description | String | Mô tả |  |
| 4 | banner | Array | Chứa ảnh và điều hướng |  |
| 5 | products | Array | Chứa ảnh và điều hướng |  |
| 6 | category | ObjectId | Id của danh mục | Khóa ngoại |
| 7 | createdBy | ObjectId | Id của admin | Khóa ngoại |

* + 1. Bảng order
* Lưu trữ các đơn hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của order | Khóa chính tự tạo |
| 2 | addressId | ObjectId | ID của address | require |
| 3 | totalAmount | Int32 | Tổng tiền | require |
| 4 | items | Array | Chứa productId,  payablePrice,  purchasedQty, |  |
| 5 | paymentStatus | Array | Chứa ảnh và điều hướng |  |
| 6 | paymentType | ObjectId | Id của danh mục | Khóa ngoại |
| 7 | user | ObjectId | Id của admin | Khóa ngoại |
| 8 | orderStatus | String | Enum: trạng thái đơn hàng |  |

* + 1. Bảng UserAddress
* Lưu trữ các địa chỉ của user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của useraddress | Khóa chính tự tạo |
| 2 | user | ObjectId | ID của user | require |
| 3 | Address | Array | List địa chỉ của 1 khách hàng | require |

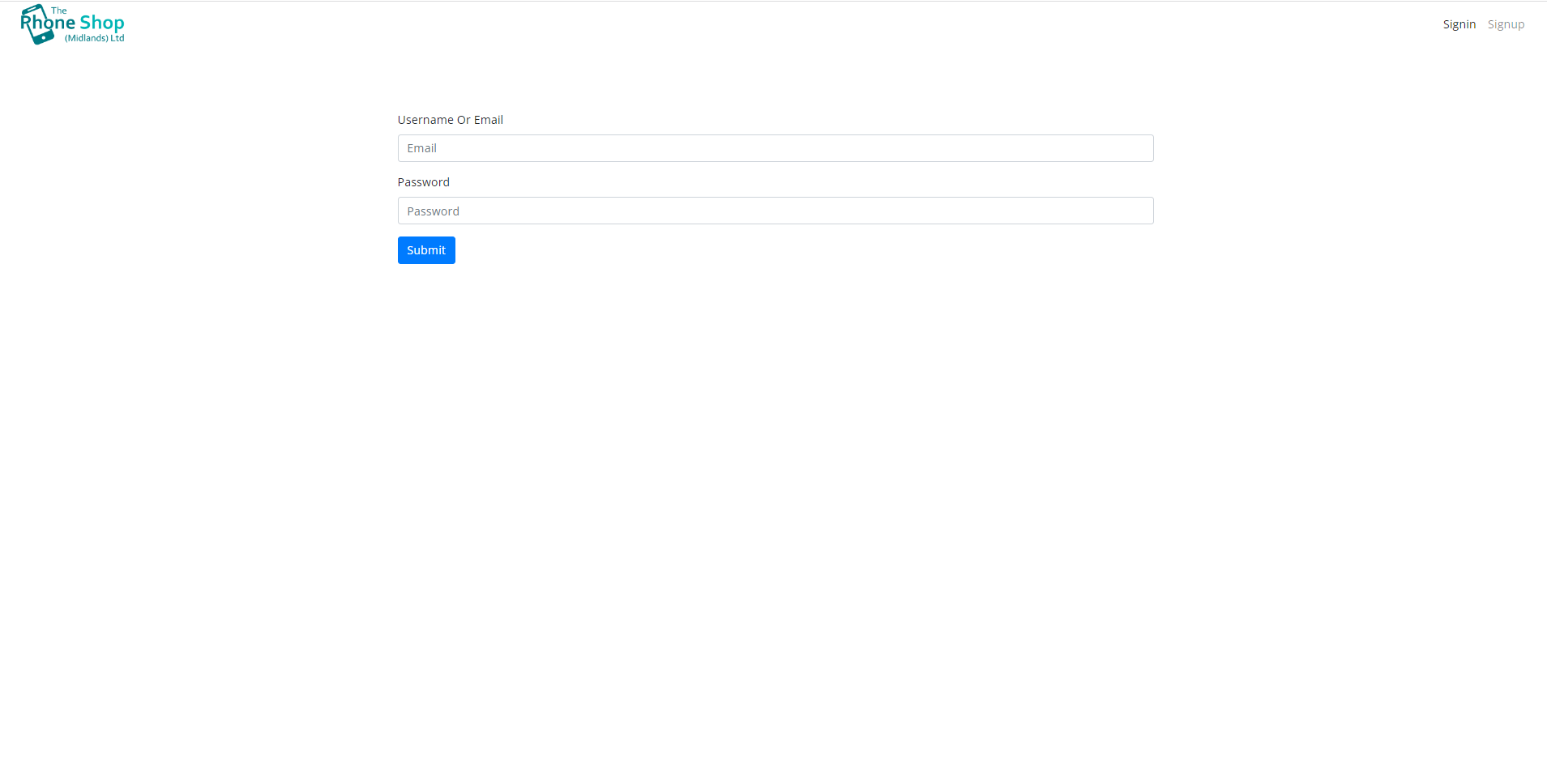
* + 1. Bảng Receipt
* Lưu trữ các địa chỉ của user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của phiếu nhập | Khóa chính tự tạo |
| 2 | items | Array | productid,price,quantity |  |
| 3 | supplier | ObjectId | ID của ncc | Khóa ngoại |
| 4 | createdBy | ObjectId | ID của user | Khóa ngoại |
| 5 | totalAmount | Number | Tổng giá | require |

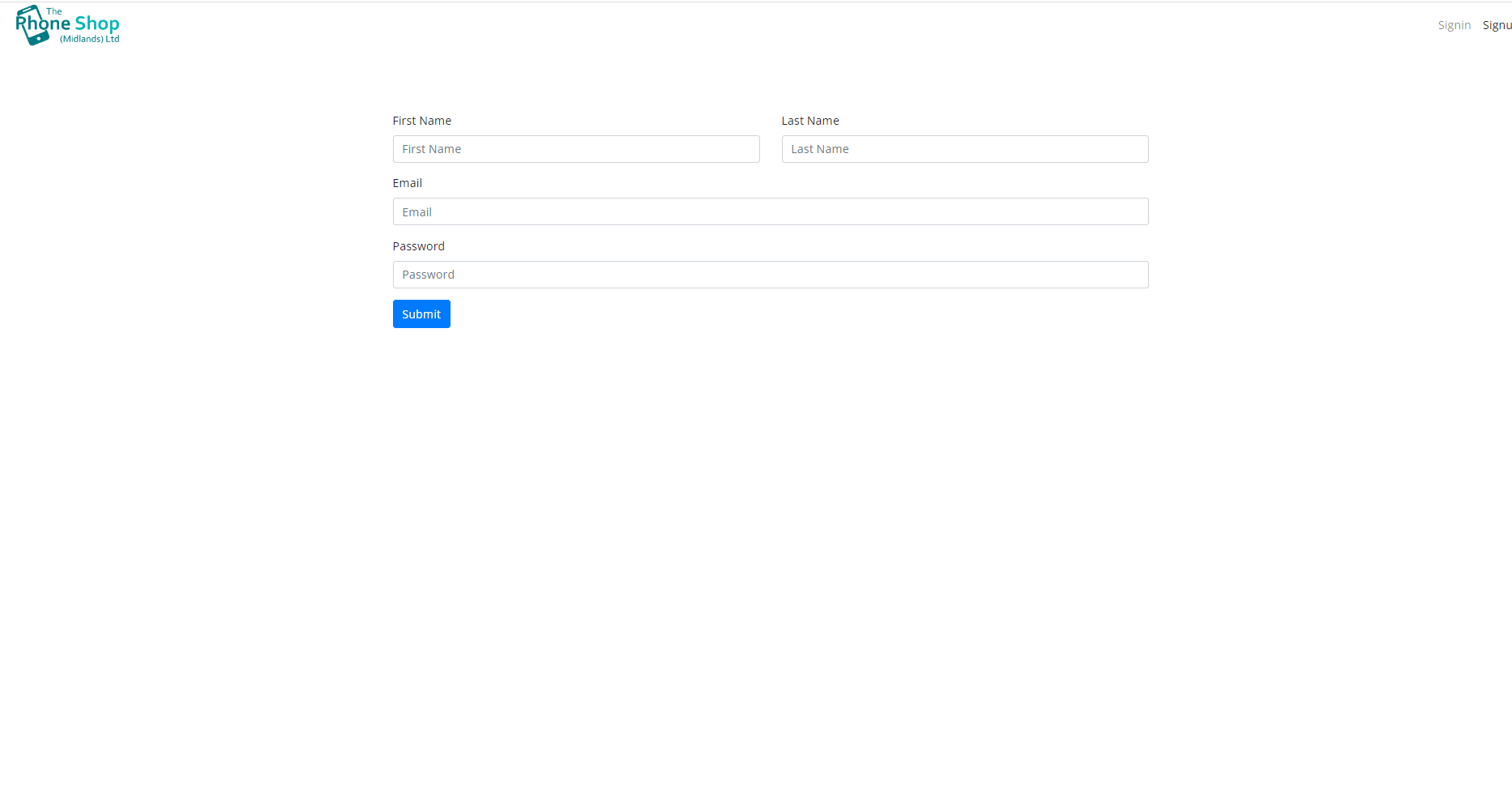
* + 1. Bảng Supplier
* Lưu trữ các địa chỉ của user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Field** | **Kiểu dữ liệu** | **Diễn giải** | **Ràng buộc** |
| 1 | id | ObjectId | ID của supplier | Khóa chính tự tạo |
| 2 | name | String | Tên Ncc | require |
| 3 | slug | String | Tên Ncc theo slug | require |
| 4 | address | String | Địa Chỉ | require |
| 5 | note | String | Ghi Chú |  |

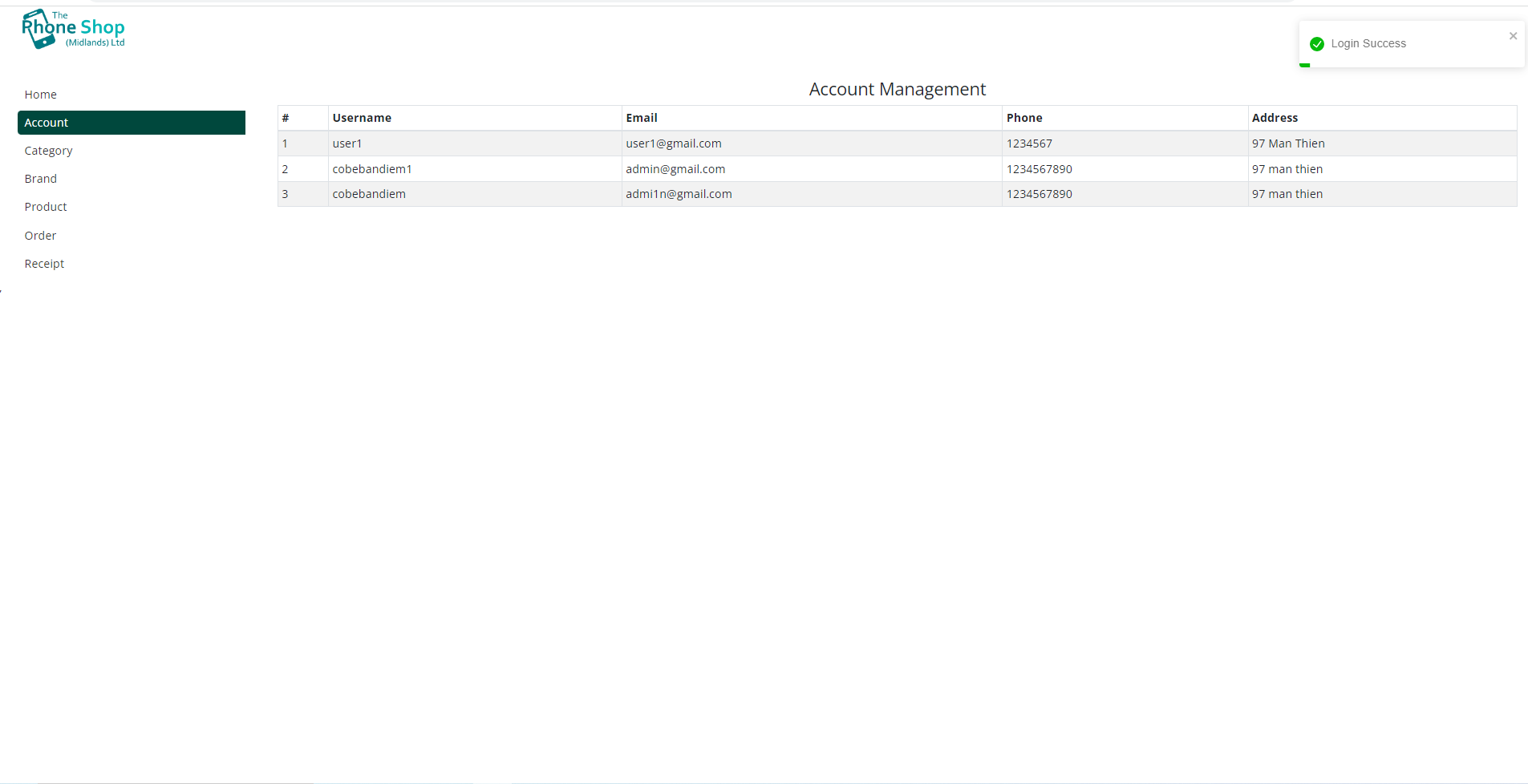
1. MINH HỌA SẢN PHẨM ĐỀ TÀI
   1. Giao diện Admin
      1. Trang đăng nhập



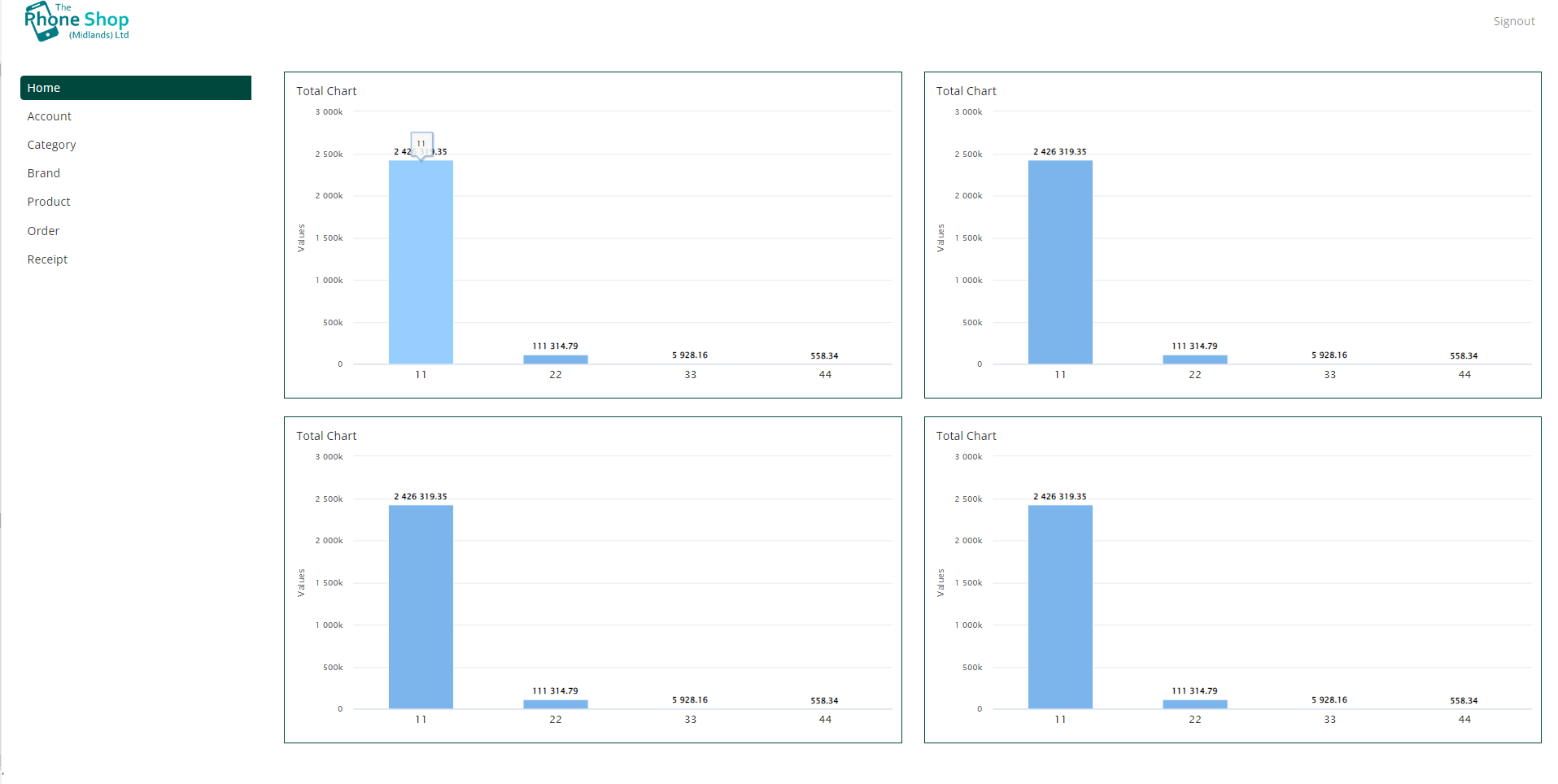
* + 1. Trang đăng ký



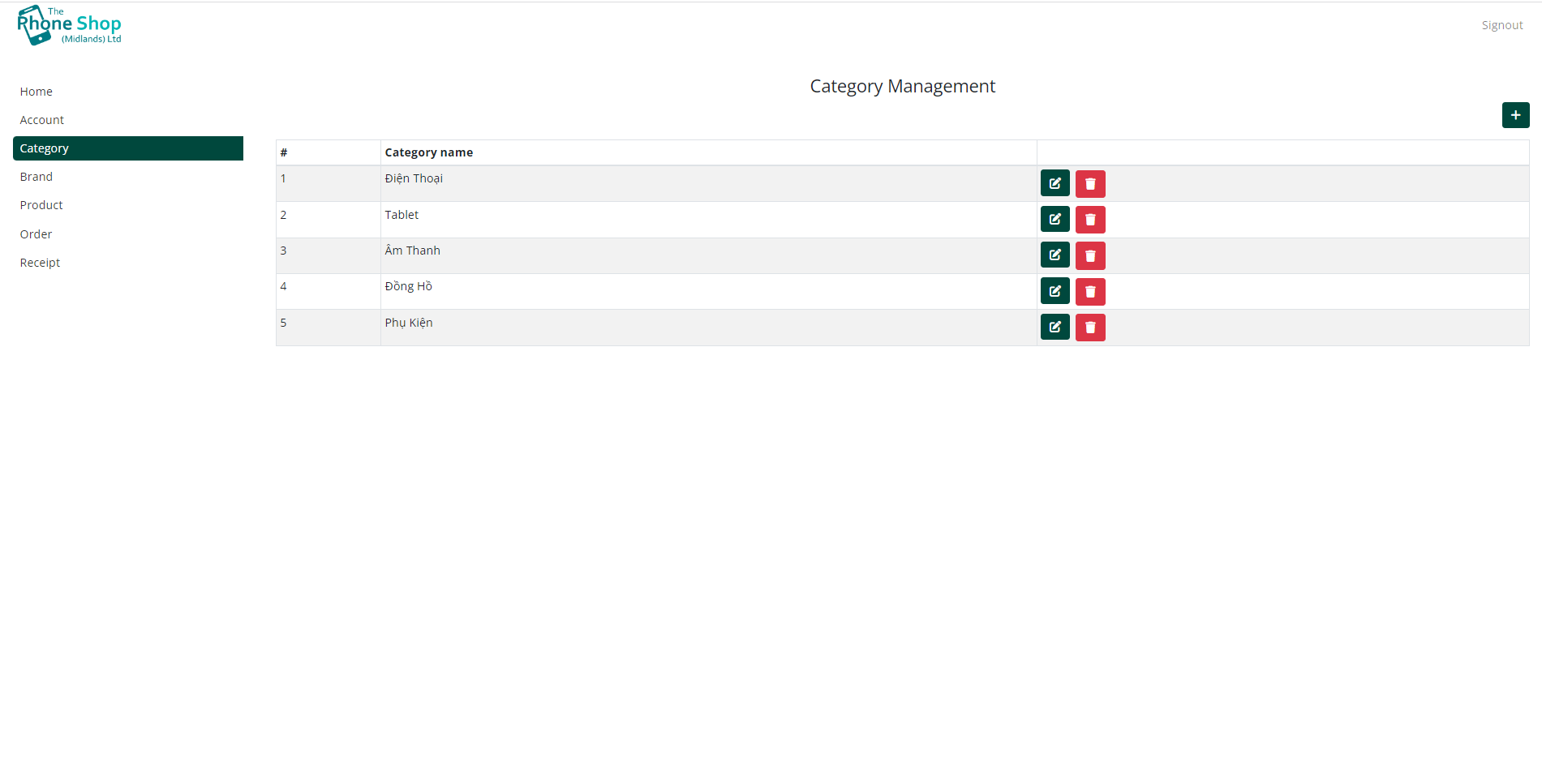
* + 1. Quản lý User khách hàng



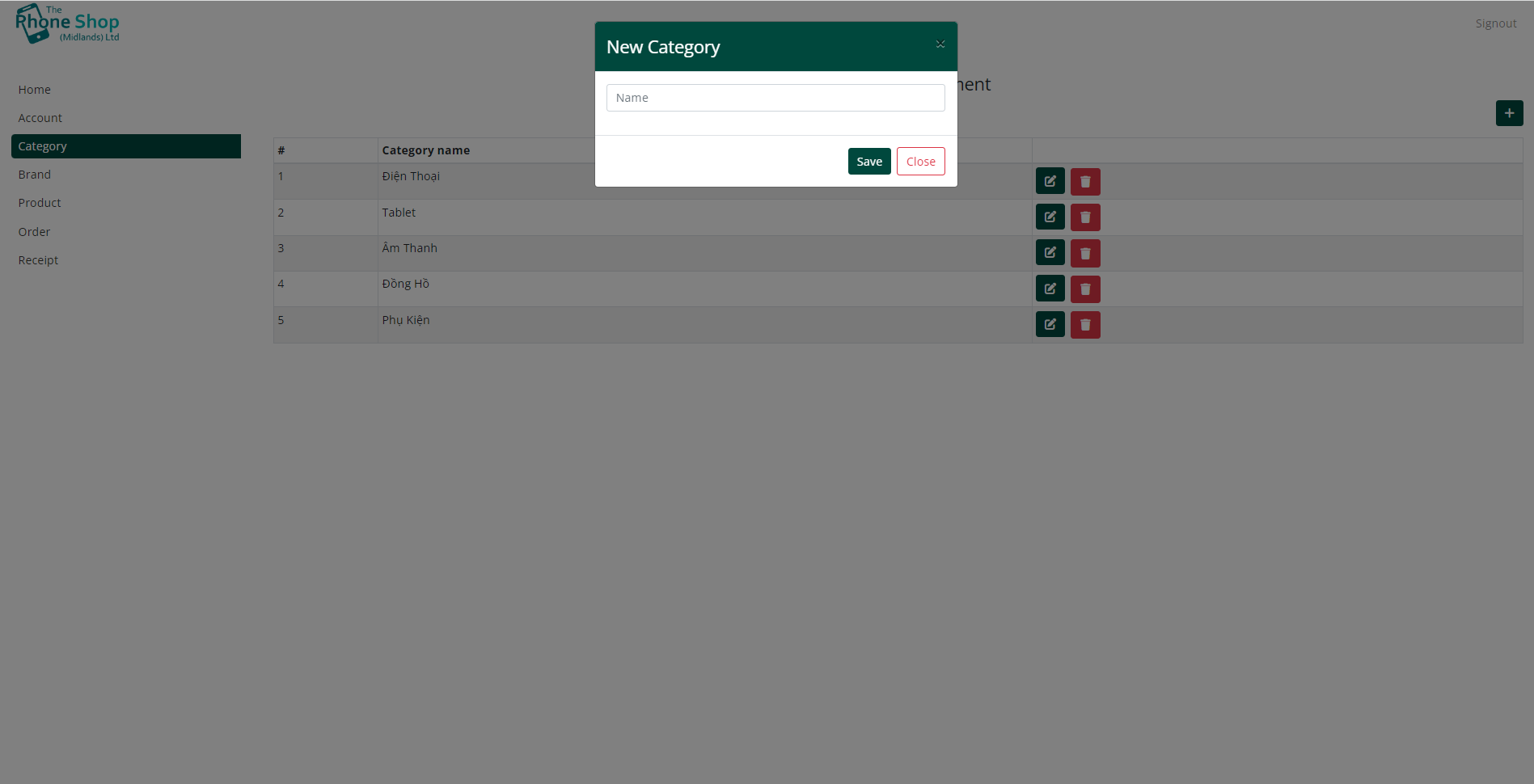
* + 1. Xem thống kê doanh thu bán hàng hàng quý



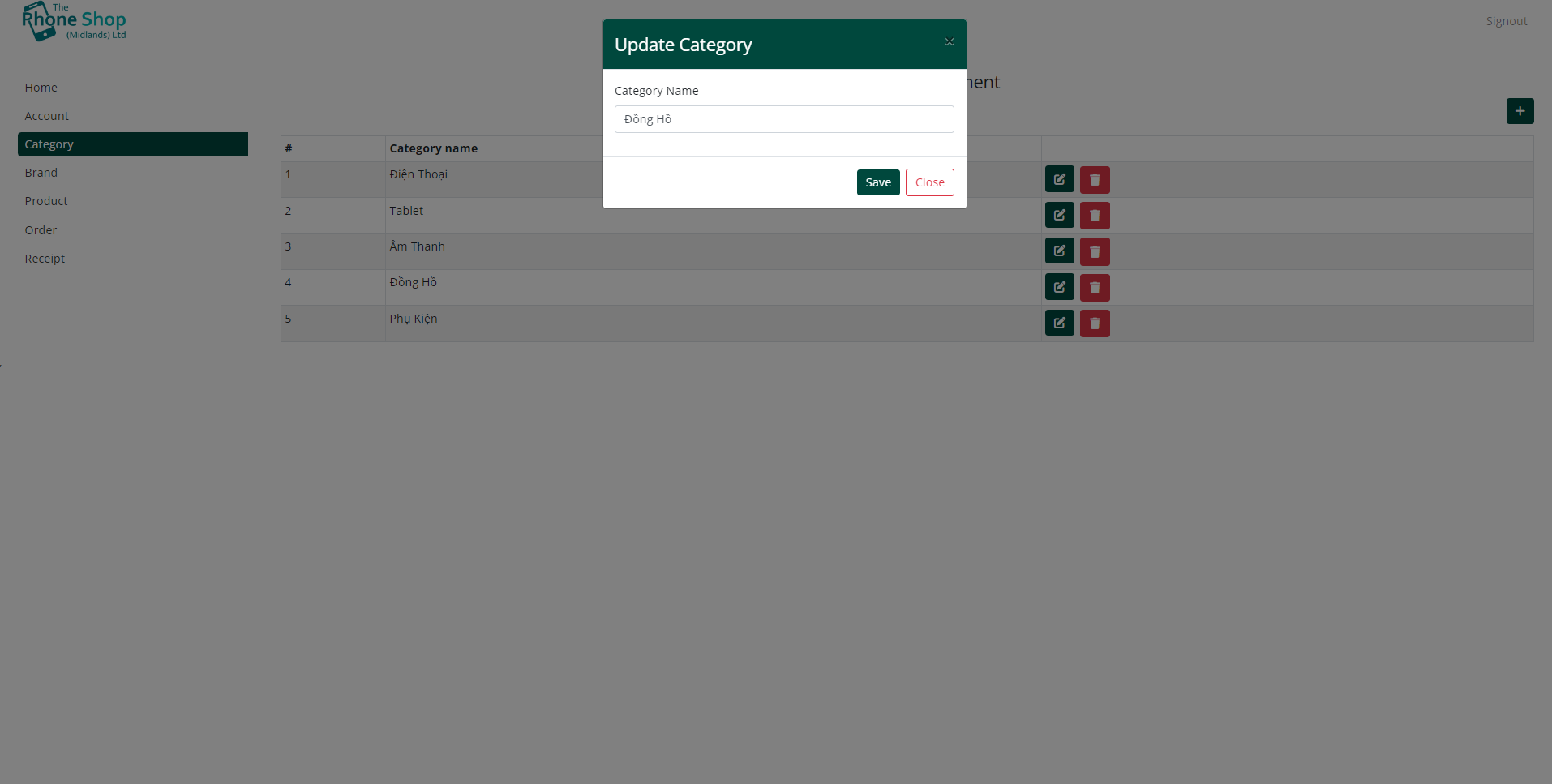
* + 1. Quản Lý Danh Mục



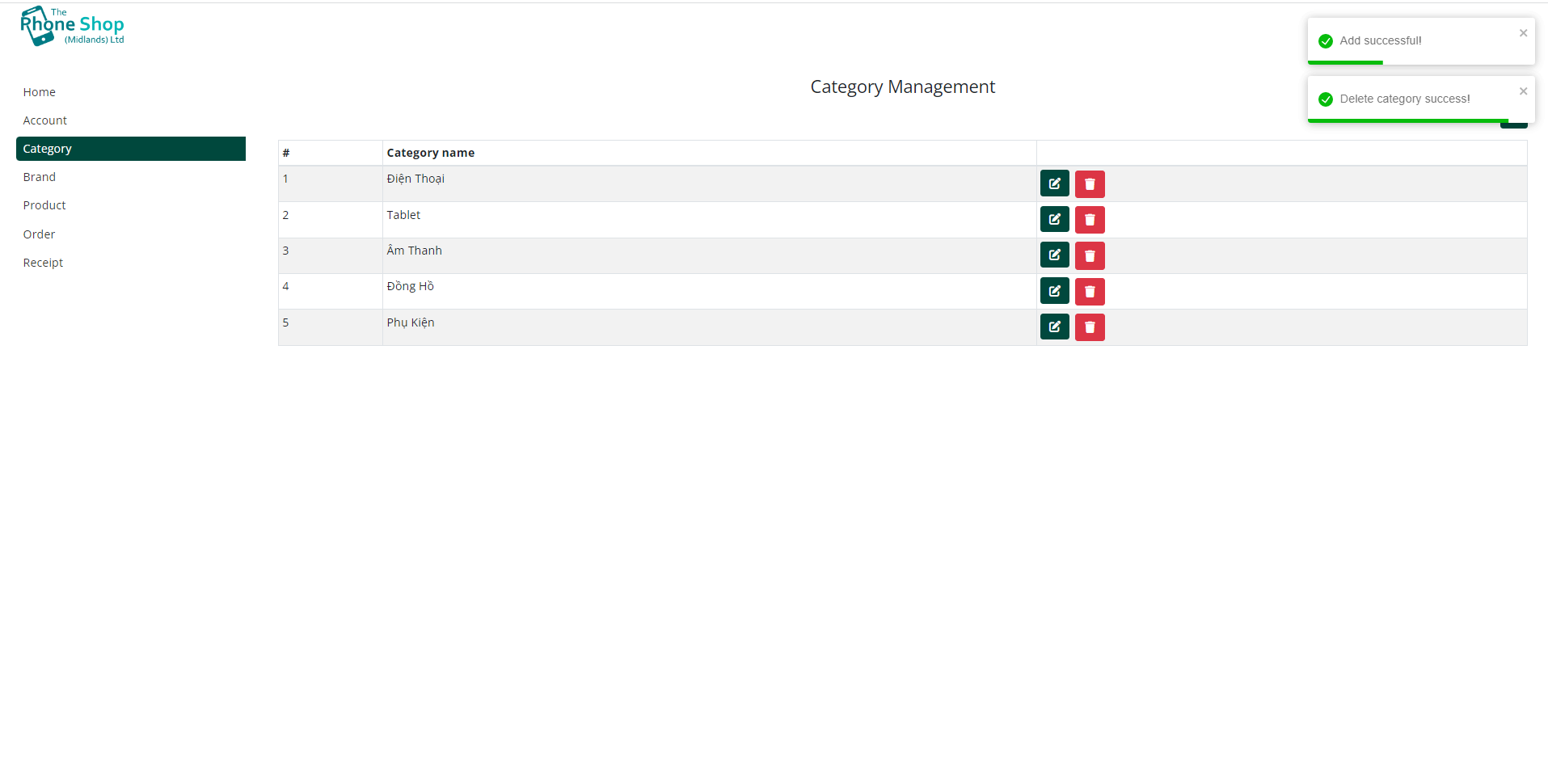
* + 1. Thêm Danh Mục



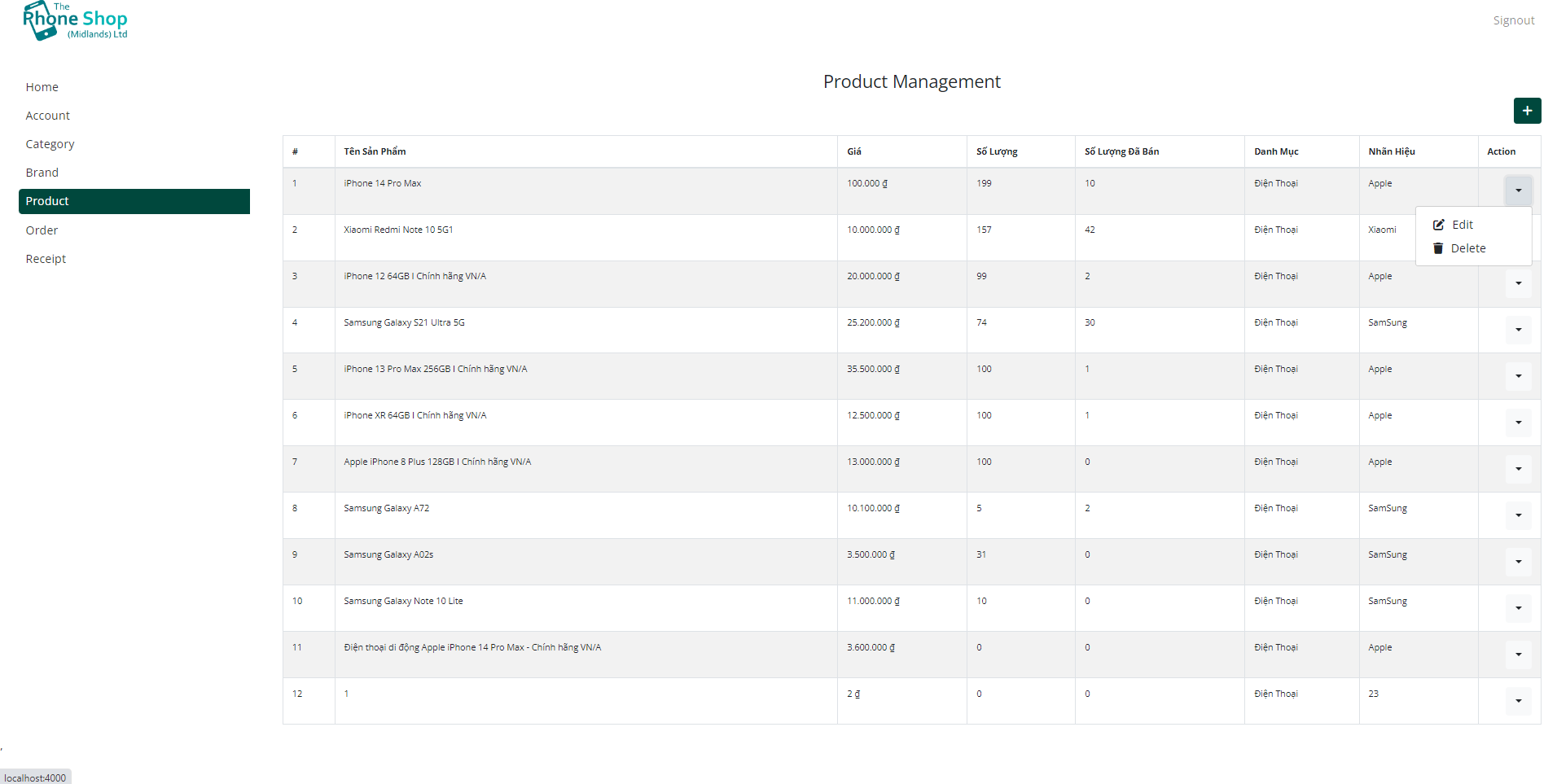
* + 1. Sửa Danh Mục



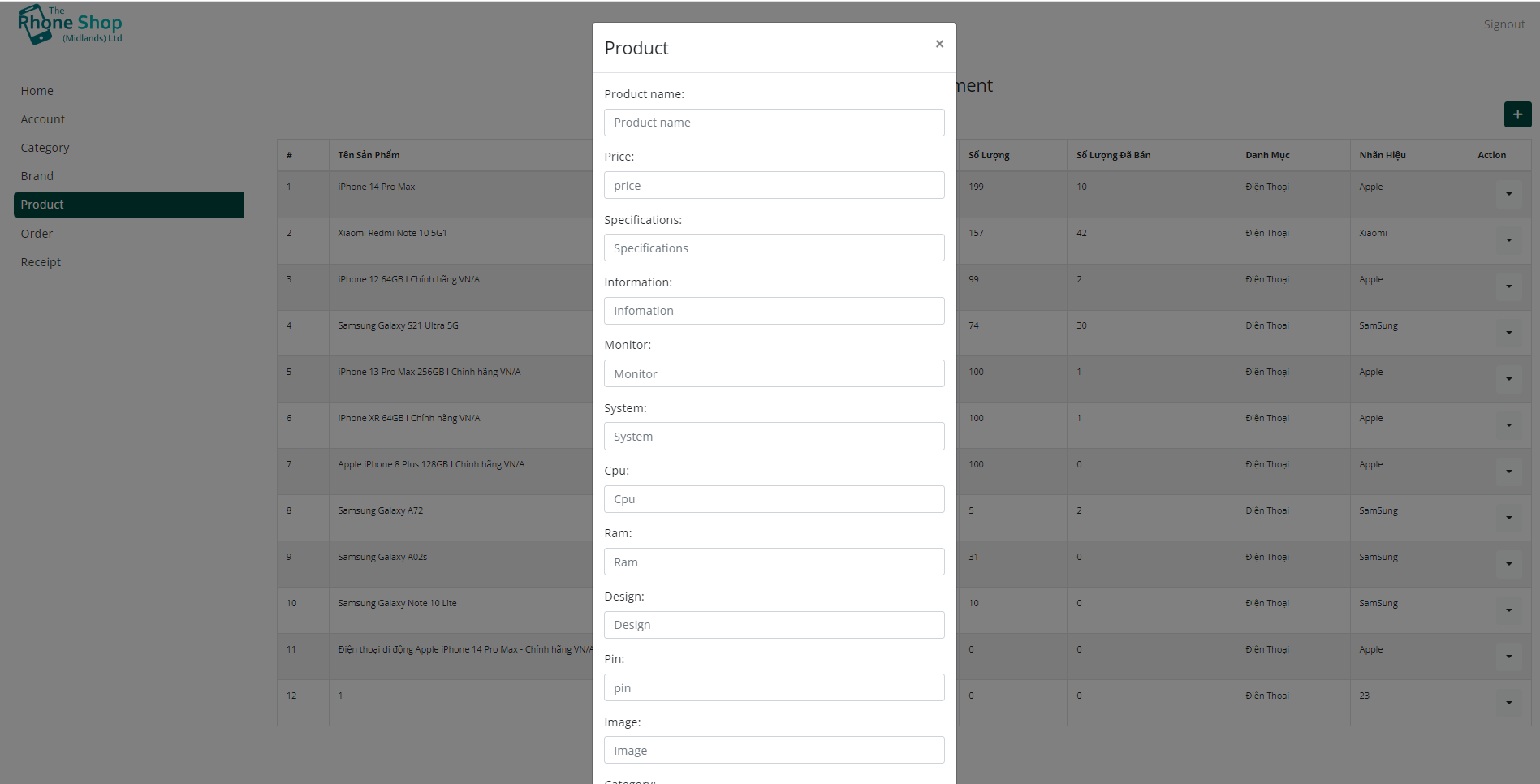
* + 1. Xóa Danh Mục



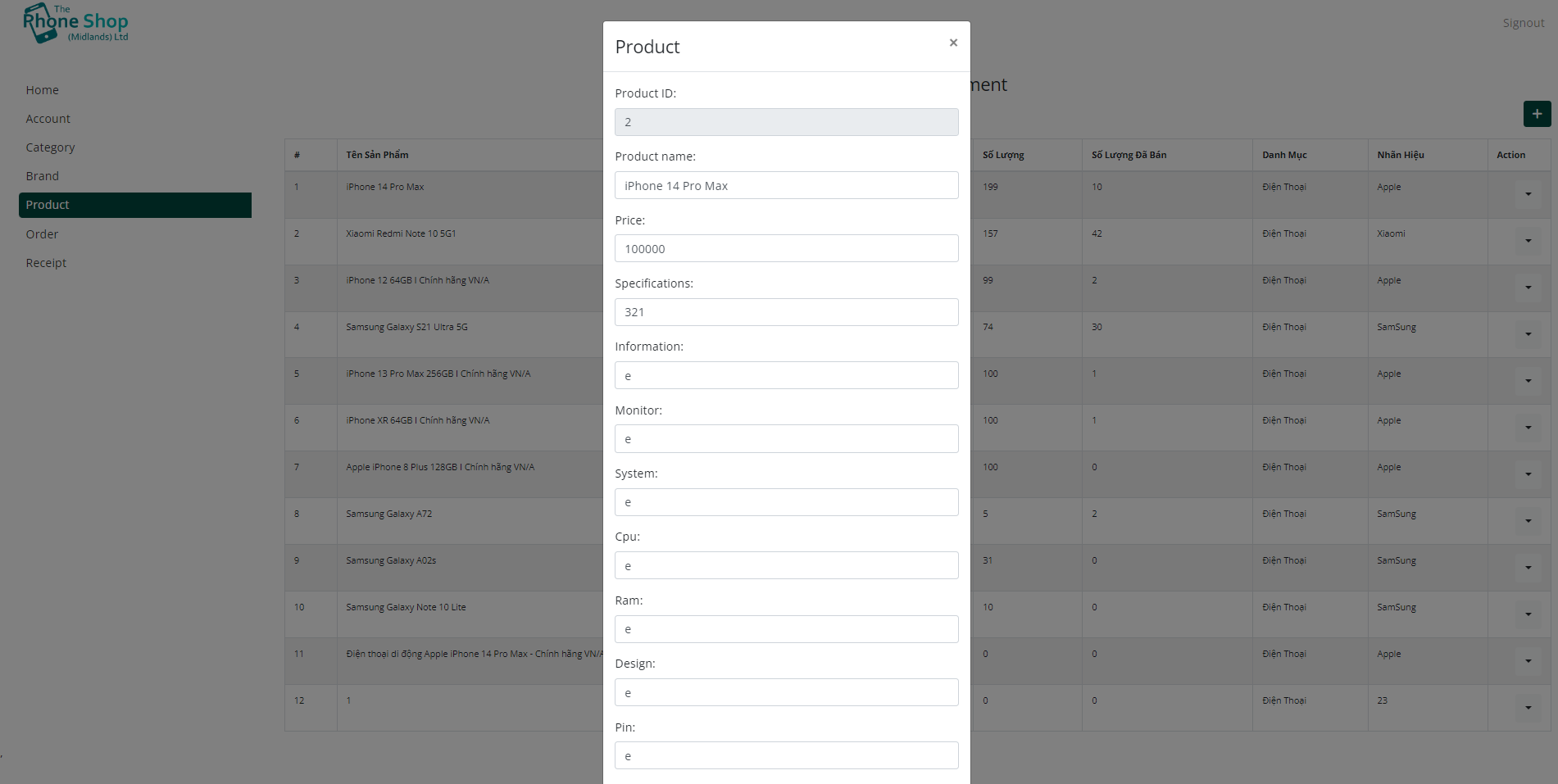
* + 1. Quản lý sản phẩm



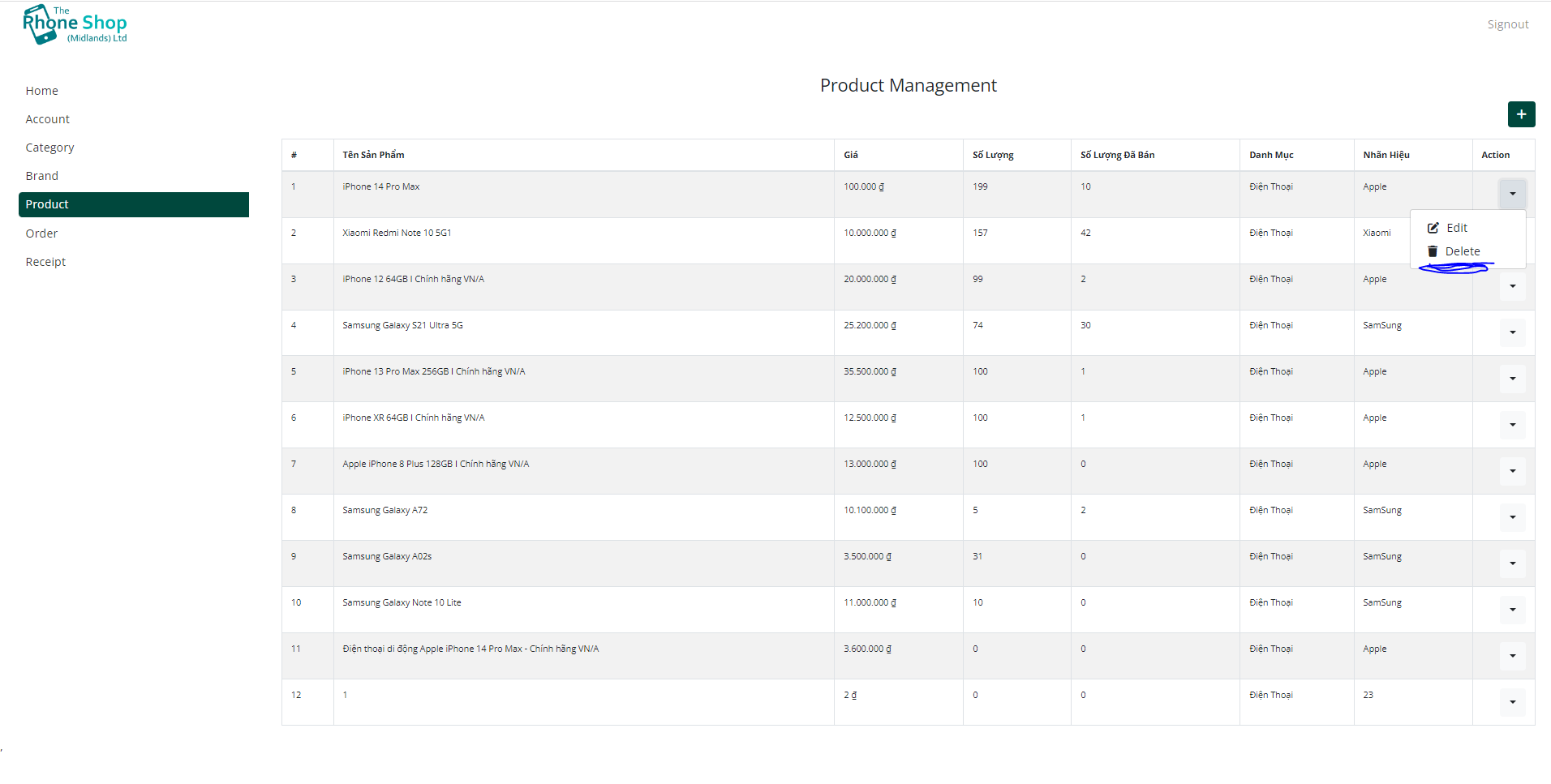
* + 1. Thêm mới sản phẩm



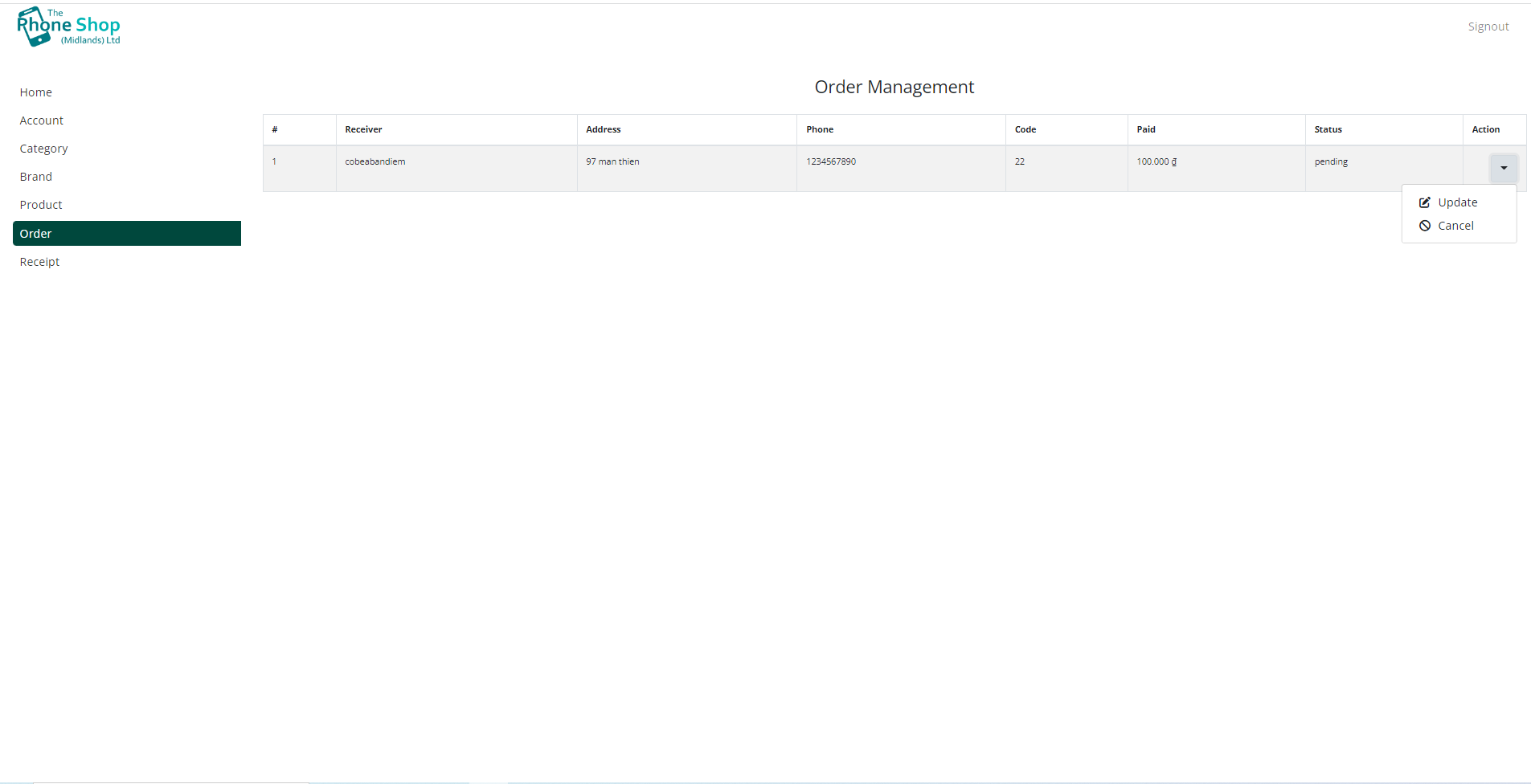
* + 1. Sửa Sản Phẩm



* + 1. Xóa Sản Phẩm



* + 1. Chi tiết sản phẩm
    2. Quản lý đơn hàng



* + 1. Chi tiết đơn hàng
    2. Xác nhận đơn hàng
    3. Tạo mới Page giao diện
  1. Giao diện người dùng
     1. Xem sản phẩm theo giá
     2. Xem sản phẩm theo danh mục
     3. Đăng nhập người dùng
     4. Đăng ký người dùng
     5. Chi tiết sản phẩm
     6. Giỏ hàng
     7. Thông tin thanh toán
     8. Xác nhận địa chỉ
     9. Thêm mới địa chỉ
     10. Chọn phương thức thanh toán
     11. Xem đơn hàng đã đặt
     12. Xem chi tiết đơn hàng