 **TRƯỜNG ĐẠI HỌC SÀI GÒN**

KHOA CÔNG NGHỆ THÔNG TIN

-----🙞🙜🕮🙞🙜-----

****

**CÁC CÔNG NGHỆ LẬP TRÌNH HIỆN ĐẠI**  
  
**Quản Lý Website Bán Hàng**

**Giảng viên hướng dẫn: Trần Đình Nghĩa  
Sinh viên thực hiện:**

* 3114410008 - Lê Văn Bảo
* 3114410171 – Phan Tăng Minh Vũ
* 3114410109 – Triệu Hoàng Phúc
* 3114410102 - Từ Ái Như

Nhận xét của giảng viên

……………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………………………………………………………………………………….………………………………………………

Mục lục

[Chương 1 6](#_Toc532750435)

[I. Lý do chọn đề tài 6](#_Toc532750436)

[II. Các công nghệ lập trình được sử dụng 6](#_Toc532750437)

[1. Frontend 6](#_Toc532750438)

[2. Backend 6](#_Toc532750439)

[Chương 2 7](#_Toc532750440)

[I. Kiến thức về backend 7](#_Toc532750441)

[II. Kiến thức về frontend 7](#_Toc532750442)

[1. HTML 7](#_Toc532750443)

[2. CSS 8](#_Toc532750444)

[3. Javascript 8](#_Toc532750445)

[4. Framework ReactJS 9](#_Toc532750446)

[III. Kiến thức về SQL Server 10](#_Toc532750447)

[1. Khái niệm 10](#_Toc532750448)

[2. Hướng dẫn cài đặt 11](#_Toc532750449)

[IV. Kiến trúc website 14](#_Toc532750450)

[1. Kiến trúc client/server 14](#_Toc532750451)

[2. Mô hình Single Page App 15](#_Toc532750452)

[Chương 3 17](#_Toc532750453)

[I. Môi trường sử dụng cho backend 17](#_Toc532750454)

[1. Cài visual studio 17](#_Toc532750455)

[2. Cài PostMan 17](#_Toc532750456)

[3. SQL server để quản lý dữ liệu . 18](#_Toc532750457)

[II. Môi trường sử dụng cho fontend 18](#_Toc532750458)

[1. Cài đặt nodeJS 18](#_Toc532750459)

[2. Tạo project ReactJS 22](#_Toc532750460)

[Chương 4 23](#_Toc532750461)

[**I.** Lược đồ usecase 23](#_Toc532750462)

[II. Lược đồ activity 23](#_Toc532750463)

[III. Hướng dẫn cài đặt source code 35](#_Toc532750464)

[1. Cài đặt cho backend 35](#_Toc532750465)

[2. Cài đặt cho frontend 40](#_Toc532750466)

[IV. Hướng dẫn sử dụng chức năng 41](#_Toc532750467)

[1. Chức năng User 41](#_Toc532750468)

[2. Chức năng Admin 46](#_Toc532750469)

# Chương 1

**GIỚI THIỆU SƠ LƯỢC**

## Lý do chọn đề tài

Website ngày càng phát triển mạnh mẽ trong thời đại công nghệ 4.0 với những công nghệ hỗ trợ và cộng đồng ngày càng rộng lớn thì việc tạo ra một website sẽ nhanh chống và dễ bảo trì hơn.

Ngoài ra, website bán hàng sẽ giúp đẩy mạnh hoạt động bán hàng mọi lúc, mọi nơi. Khi có một website thì chỉ cần khách hàng có kết nối internet với các thiết bị truy cập thì khách hàng có thể truy cập vào website bán hàng của bạn mọi nơi vào mọi thời điểm. Khách hàng sẽ không phải đến trực tiếp cửa hàng hay mua hàng theo giờ giấc hoạt động của cửa hàng mà có thể mua hàng bất cứ lúc nào. Điều này giúp tăng tối đa cơ hội bán hàng, giúp thúc đẩy doanh số và thuận tiện cho khách hàng trong việc mua sắm, làm cho họ hài lòng hơn.

## Các công nghệ lập trình được sử dụng

### Frontend

* Sử dụng framework NodeJs, ReactJS.
* Cú pháp Javacript ES6.

### Backend

* Sử dụng công nghệ Entity Framework trong C# (EF 6)
* Database SQL SERVER Web Service sử dụng WebAPI của .N.et
* Chuẩn RESTful API.

# Chương 2

**KIẾN THỨC CẦN THIẾT**

## Kiến thức về backend

Để triển khai được hệ thống backend, ta cần phải nắm được kiến thức về ngôn ngữ lập trình .NET, có kiến thức về SQL Server . Biết sử dụng Entity Framework để có thể kết nối cơ sở dữ liệu SQL và .NET và thực hiện việc truy vấn cơ sở dữ liệu qua Entity Framework. Có kiến thức về API (HTTP Post, Get, Put, Patch, Delete) , kiến thức về kiểu dữ liệu trả về Json.

Cần biết sử dụng một số phần mềm hỗ trợ để thiết kế và triển khai api.

* POSTMAN : để test API.
* SQL SERVER : để quản lý , lưu trữ dữ liệu .
* VISUAL STUDIO 2017 : để viết code .

## Kiến thức về frontend

### HTML

HTML (tiếng Anh, viết tắt cho HyperText Markup Language, hay là "Ngôn ngữ Đánh dấu Siêu văn bản") là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web với các mẩu thông tin được trình bày trên World Wide Web.

Ví dụ:

|  |
| --- |
| <!DOCTYPE html> <html>  <head>  <title>Page Title</title>  </head>  <body>  <h1>This is a Heading</h1>  <p>This is a paragraph.</p>  </body> </html> |

Một trang web được bắt đầu bằng thẻ mở <html> và kết thúc bằng thẻ đóng </html>. Trong đó bao gồm 2 nội dung chính là thông tin header được khai báo trong cặp thẻ <head></head>, nội dung trang web được khai báo trong cặp thẻ <body></body>

Nội dung đặt trong cặp thẻ <head></head> bao gồm:

* + Tiêu đề trang web (title)
  + Phần mô tả trang web (description)
  + Tên tác giả (author)
  + Từ khóa liên quan đến nội dung trang web…

Nội dung trong cặp thẻ <body></body> được hiển thị trên trang web, gồm các cặp thẻ như là <p></p>, <div></div>, <table></table>, <span></span>, <a></a>….

### CSS

CSS (Cascading Style Sheet). Website được cấu tạo từ các thẻ html nhưng các thẻ html thì chỉ thể hiện được bộ khung của website. Để căn chỉnh, trình bày cho đẹp mặt thì ta cần sử dụng ngôn ngữ CSS. Đây là ngôn ngữ được dùng rấy nhiều trong lập trình web, thường đi cùng với ngôn ngữ html.

Chúng ta có 3 cách đẻ chèn CSS và trang HTML

* Chèn nội dung CSS vào cặp thẻ <style></style> trong phần <head></head> của trang web
* Chèn trực tiếp vào bên trong thẻ HTML
* Liên kết với một file .css bên ngoài

Trong thực tế, cách thứ 3 được các lập trình viên sử dụng nhiều nhất do tính tiện dụng và linh hoạt.

### Javascript

JavaScript hoặc JS sẽ giúp tăng tính tương tác trên website. Script này chạy trên các trình duyệt của người dùng thay vì trên server và thường sử dụng thư vuiên của bên thứ 3 nên có thể tăng thêm chức năng cho website mà không phải code từ đầu.

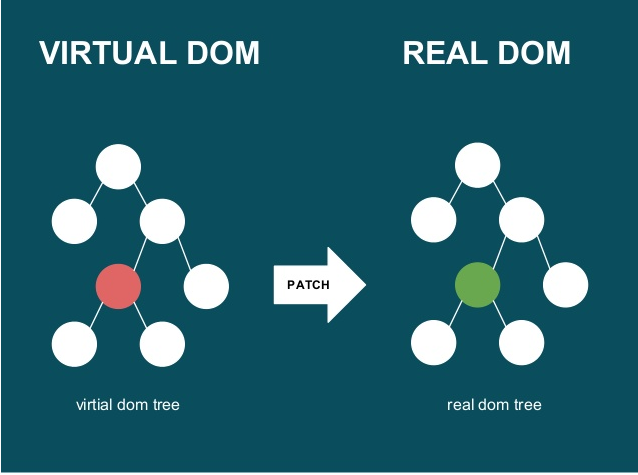
JavaScript thường được nhúng trực tiếp vào một trang web hoặc được tham chiếu qua file .js riêng. Nó là ngôn ngữ phía client, tức là script được tải về máy của khách truy cập và được xử lý tại đó thay vì phía server là xử lý trên server rồi mới đưa kết quả tới khách truy cập.

### Framework ReactJS

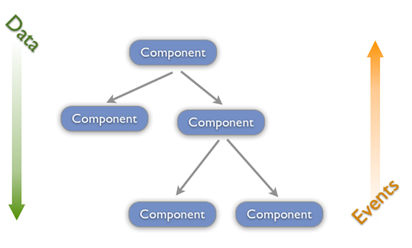
React là một thư viện UI phát triển tại Facebook để hỗ trợ việc xây dựng những thành phần (components) UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được. Với xu hướng Single Page Application.

React là một library cho phép nhúng code html trong code javascript nhờ vào JSX, bạn có thể dễ dàng lồng các đoạn HTML vào trong JS.Tích hợp giữa javascript và HTML vào trong JSX làm cho các component dễ hiểu hơn.

Virtual DOM: Công nghệ DOM ảo giúp tăng hiệu năng cho ứng dụng. Virtual DOM là một object Javascript, mỗi object chứa đầy đủ thông tin cần thiết để tạo ra một DOM, khi dữ liệu thay đổi nó sẽ tính toán sự thay đổi giữa object và tree thật, điều này sẽ giúp tối ưu hoá việc re-render DOM tree thật.



React sử dụng cơ chế one-way data binding – luồng dữ liệu 1 chiều. Dữ liệu được truyền từ parent đến child thông qua props. Luồng dữ liệu đơn giản giúp chúng ta dễ dàng kiểm soát cũng như sửa lỗi.



Component: React được xây dựng xung quanh các component. Trong React, chúng ta xây dựng trang web sử dụng những thành phần (component) nhỏ. Chúng ta có thể tái sử dụng một component ở nhiều nơi, với các trạng thái hoặc các thuộc tính khác nhau, trong một component lại có thể chứa thành phần khác. Mỗi component trong React có một trạng thái riêng, có thể thay đổi, và React sẽ thực hiện cập nhật component dựa trên những thay đổi của trạng thái.

Props: giúp các component tương tác với nhau, component nhận input gọi là props, và trả thuộc tính mô tả những gì component con sẽ render. Prop là bất biến.

State: thể hiện trạng thái của ứng dụng, khi state thay đồi thì component đồng thời render lại để cập nhật UI.

## Kiến thức về SQL Server

### Khái niệm

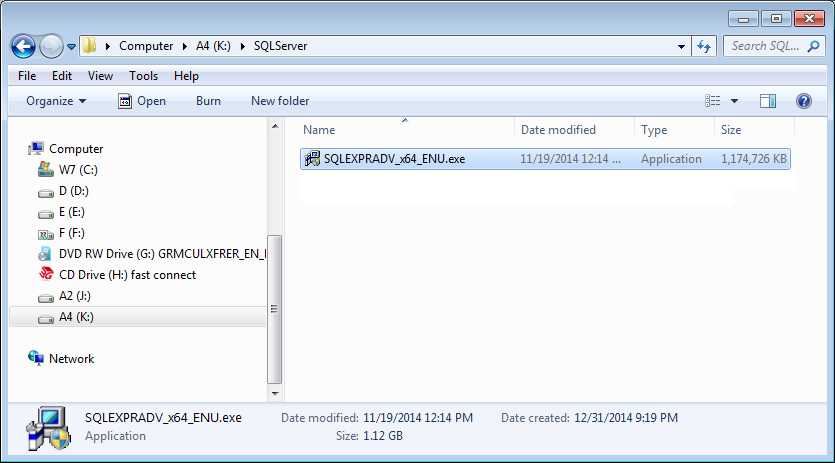
SQL là ngôn ngữ phi thủ tục, không yêu cầu cách thức truy cập cơ sở dữ liệu như thế nào. Tất cả các thông báo của SQL rất dễ dàng sử dụng và ít mắc lỗi.

SQL cung cấp các tập lệnh phong phú cho các công việc hỏi đáp dữ liệu như:

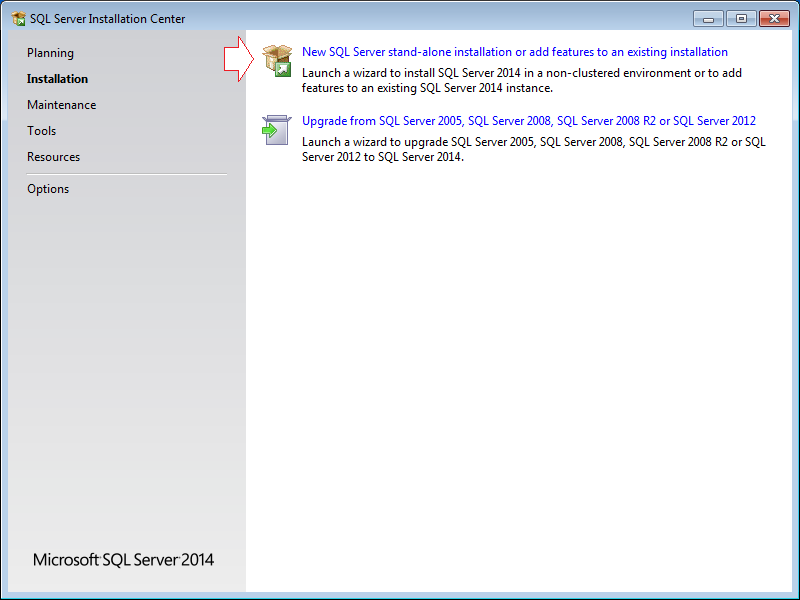
* Chèn, xóa và cập nhật các hàng trong 1 quan hệ
* Tạo thêm, xóa và sửa đổi các đối tượng trong của cơ sở dữ liệu.
* Điều khiển việc truy cấp tới cơ sở dữ liệu và các đối tượng của cơ sở dữ liệu để đảm bảo tính bảo mật, tính nhất quán và sự ràng buộc của cơ sở dữ liệu.
* Đối tượng của SQL server là các bảng dữ liệu với các cột và các hàng. Cột được gọi là trường dữ liệu và hàng là bản ghi của bảng. Cột dữ liệu và kiểu dữ liệu xác định tạo nên cấu trúc của bảng. Khi bảng được tổ chức thành một hệ thống cho một mục đích sử dụng cụ thể vào công việc nào đó sẽ trở thành một cơ sở dữ liệu.

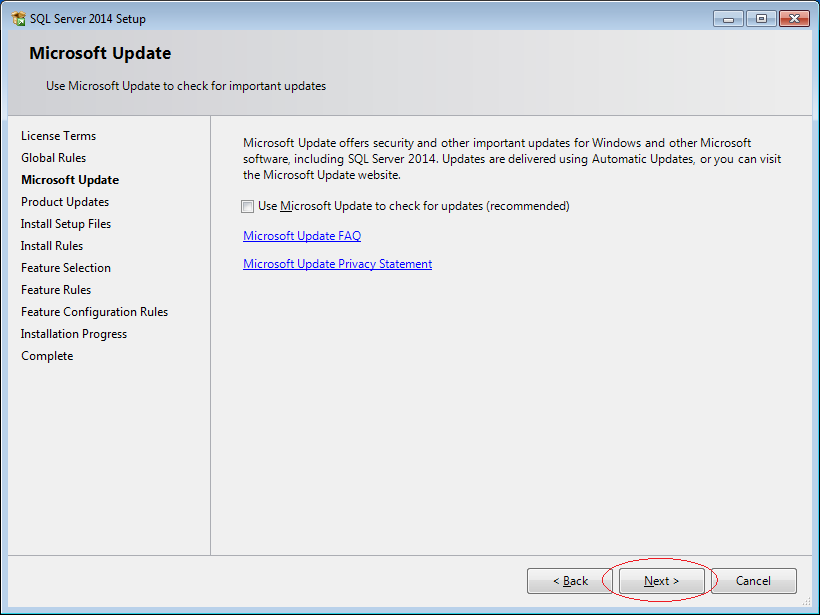
### Hướng dẫn cài đặt

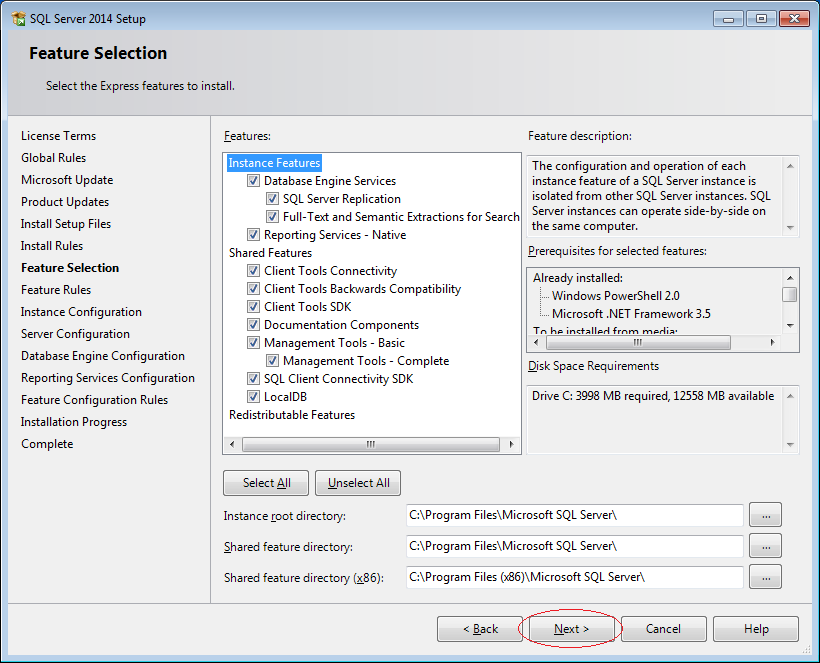
Bước 1: Truy cập link <https://www.microsoft.com/en-in/sql-server/sql-server-downloads> và download

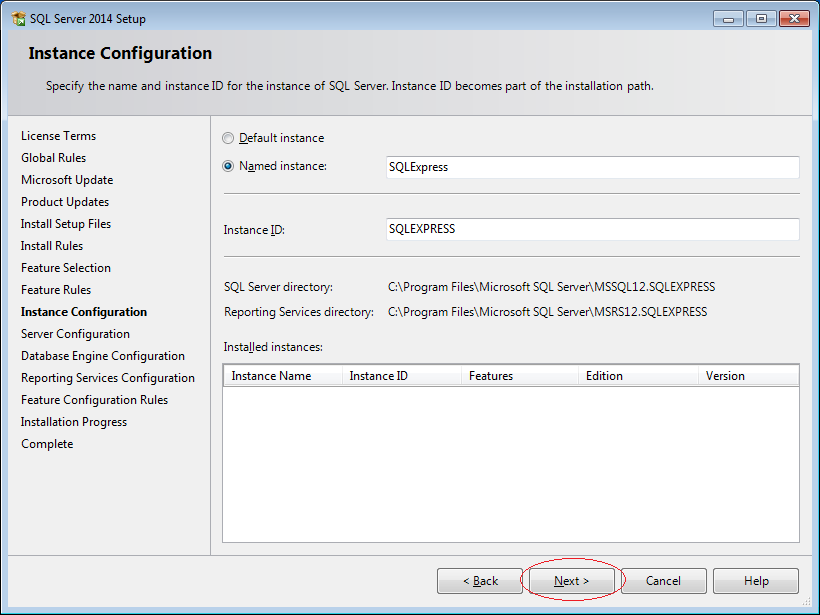


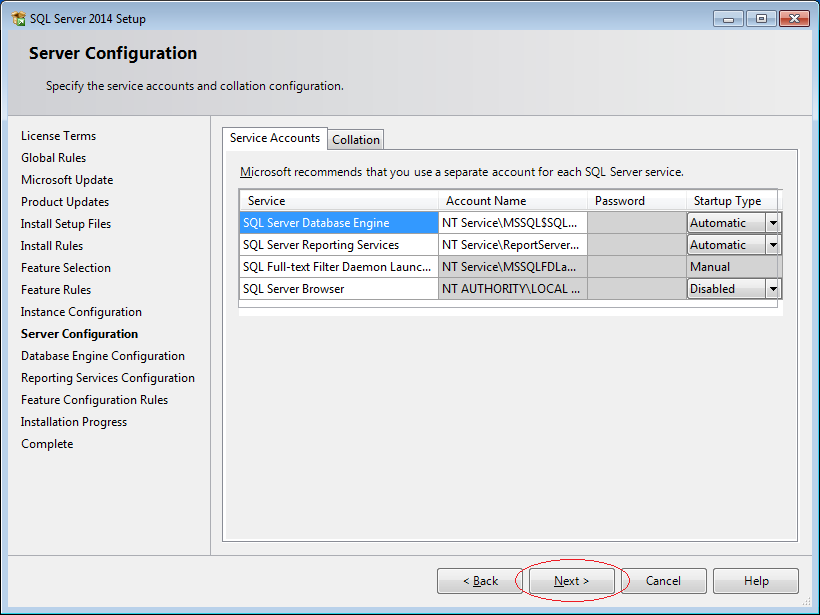
Bước 2: Tiến hành cài đặt

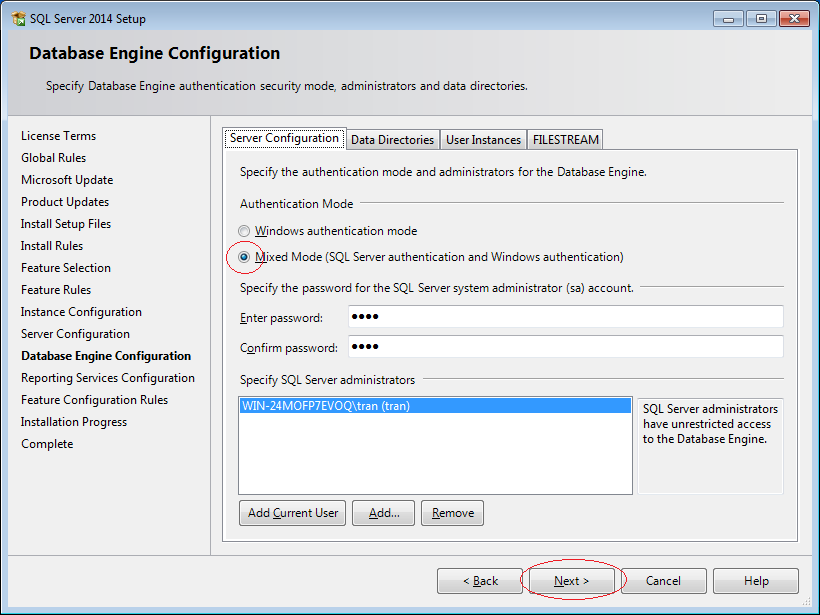


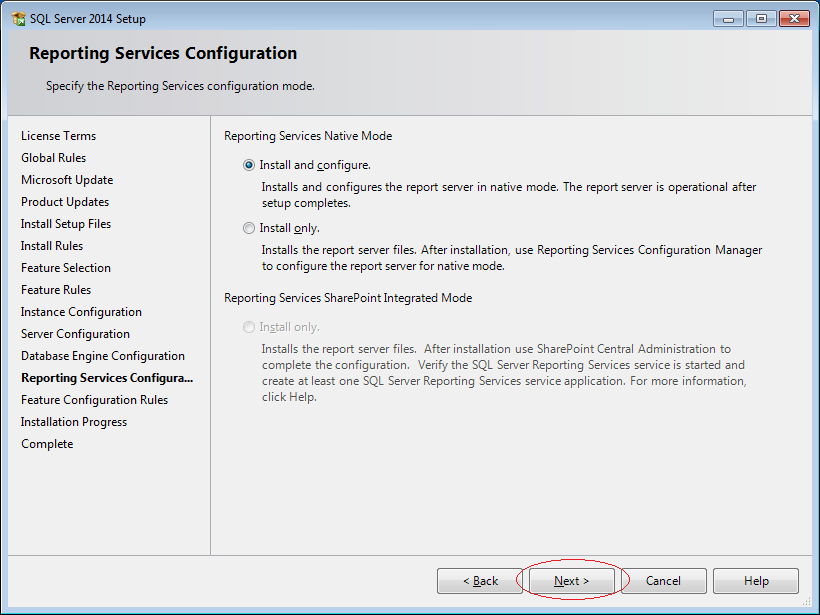


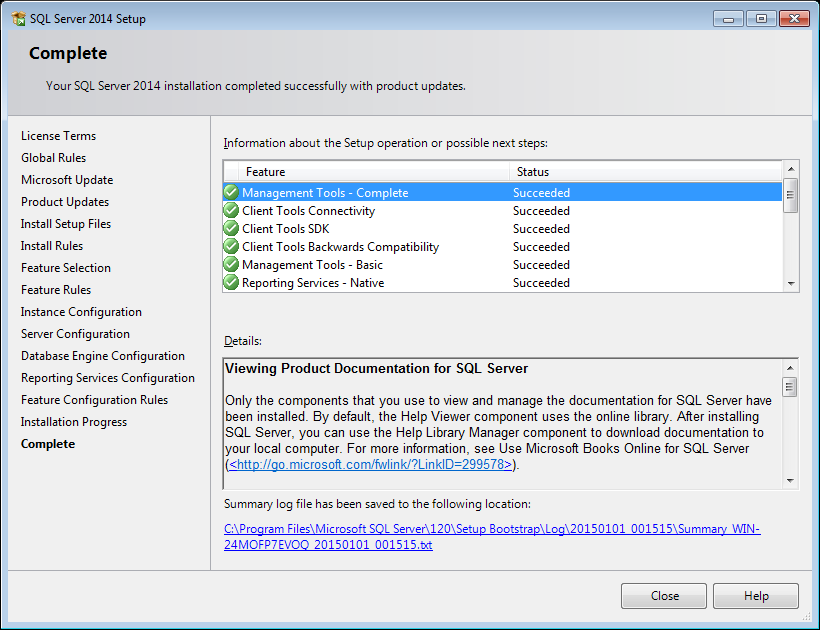








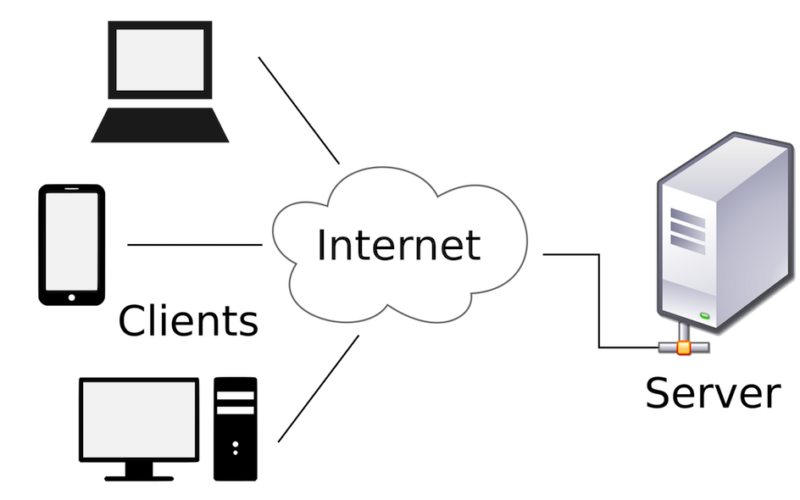




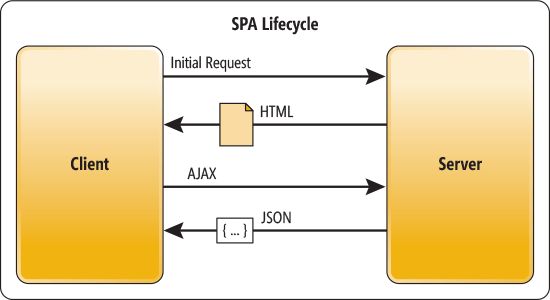
## Kiến trúc website

### Kiến trúc client/server

Là kiến trúc nổi tiếng trong mạng máy tính, hầu hết các website hoạt động dựa trên kiến trúc này. Trong đó Client là máy khách gửi yêu cầu đến máy Server. Tại đây thì Server lắng nghe các yêu cầu từ máy Client, nhận thông tin từ Client sau đó xử lý, trả kết quả về cho máy Client



### Mô hình Single Page App



Single page app xử lý dựa trên nền các library, framework java script và ***phương thức truyền dữ liệu chính là dùng ajax***.

**Browser (Client site):**

* + - Yêu cầu technique: phải hỗ trợ java script, các library java script dùng ajax là chính.
    - Hoạt động: mỗi lần load page lần đầu khá lâu, vì load master page và khởi tạo các Object model (lưu trữ data vào object java script). Nhưng sẽ không tốn thời gian load lại khi chuyển sang các page khác vì mọi thứ đều được các library hoặc framework sẽ render thành view, còn data sẽ được truyền thông qua ajax và được lưu vào các object!

**Web server (Server site):**

* + - Mỗi lần nhận response thì Server chỉ xử lý data, bussiness logic và gửi data theo kiểu json object dữ liệu rất nhẹ về client site.

# Chương 3

**THIẾT LẬP MÔI TRƯỜNG**

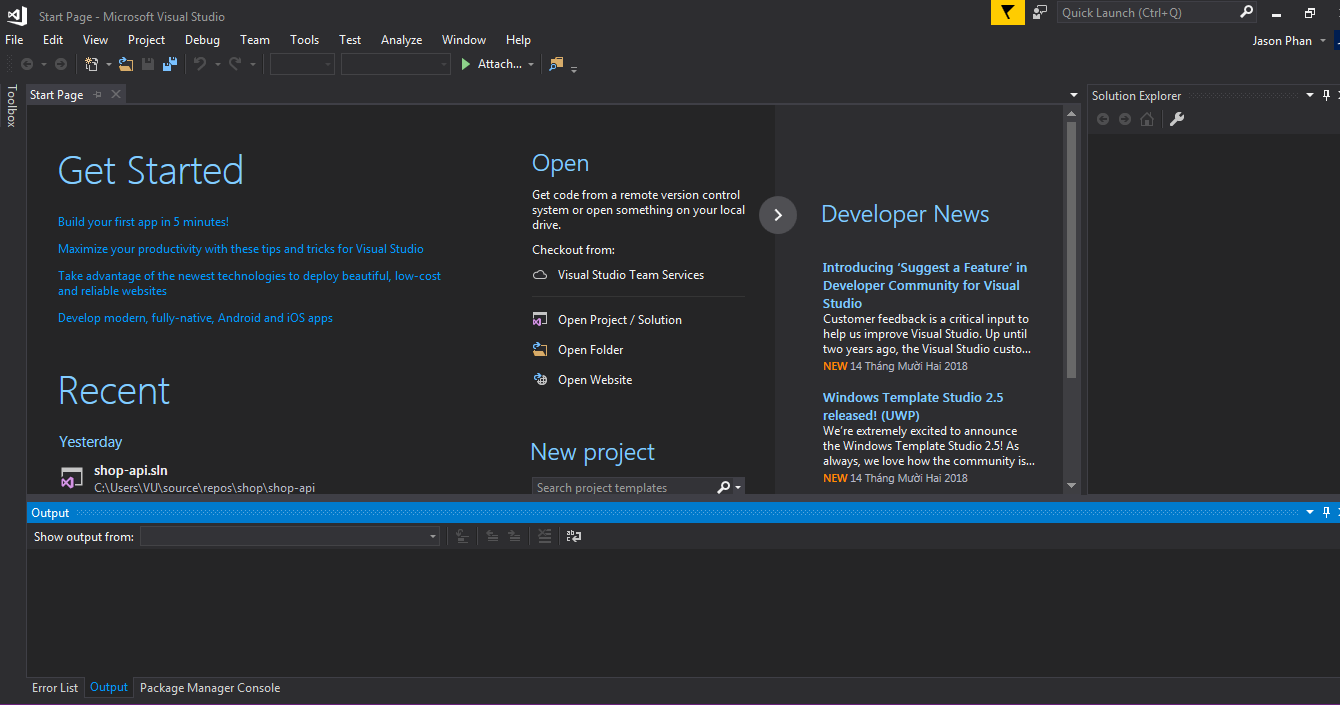
## Môi trường sử dụng cho backend

### Cài visual studio

Máy tính phải có Visual Studio để lập trình API

Có thể sử dụng Visual Studio Community 2017 , download trên trang :

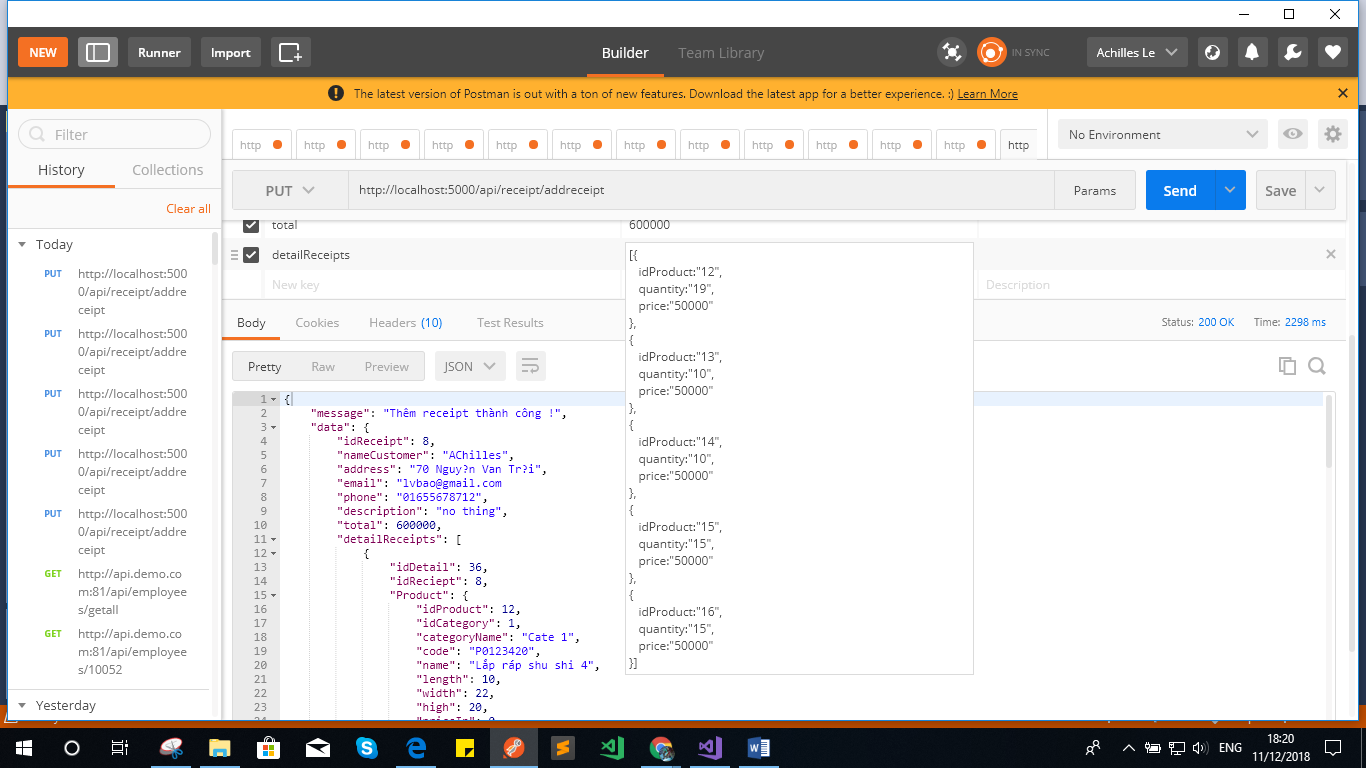
<https://visualstudio.microsoft.com/downloads/>.



### Cài PostMan

Phần mềm Postman để test API, download tại trang:

https://www.getpostman.com/



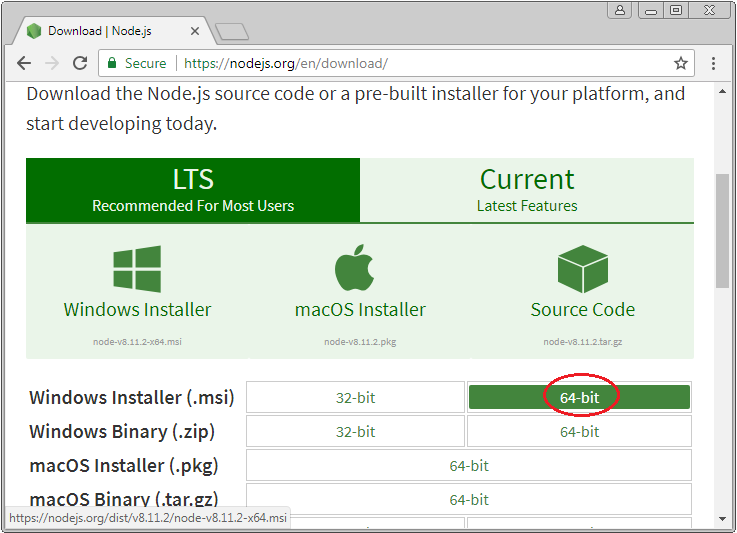
### SQL server để quản lý dữ liệu

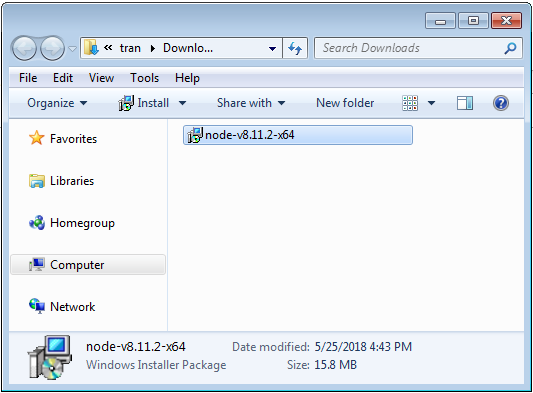
Được đề cập ở chương 2 , kiến thức về sqlserver

## Môi trường sử dụng cho fontend

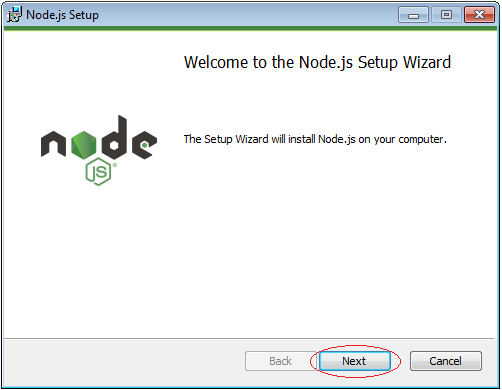
### Cài đặt nodeJS

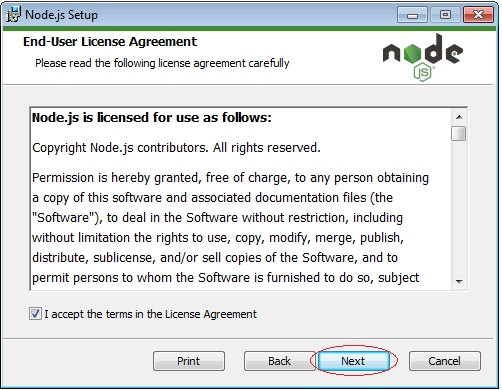
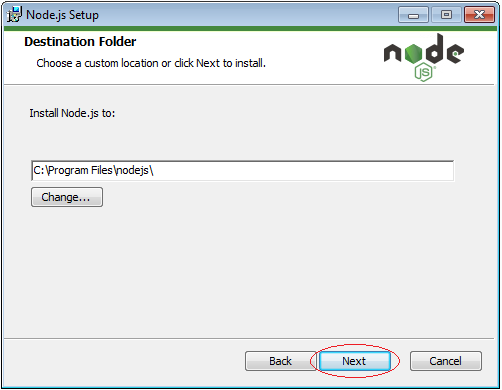
Bước 1: Truy cập <https://nodejs.org/en/download/>

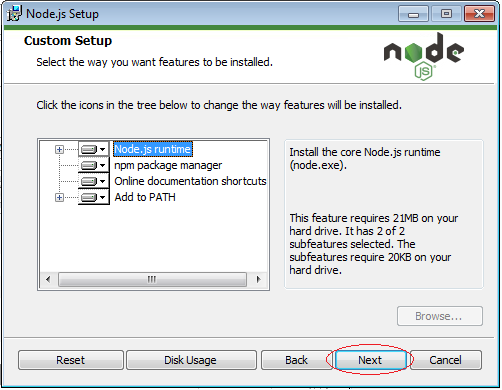


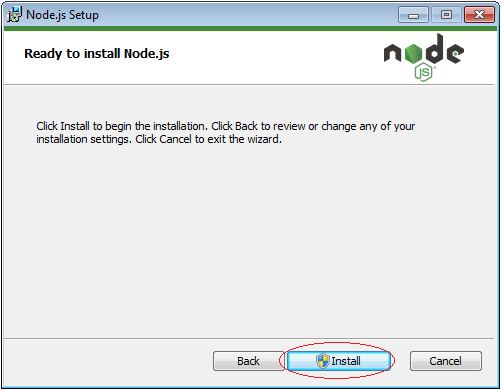


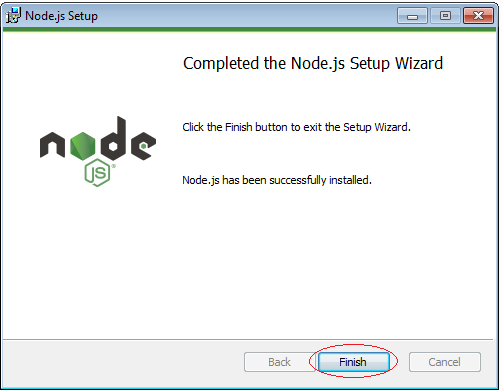
Bước 2: Tiến hành cài đặt



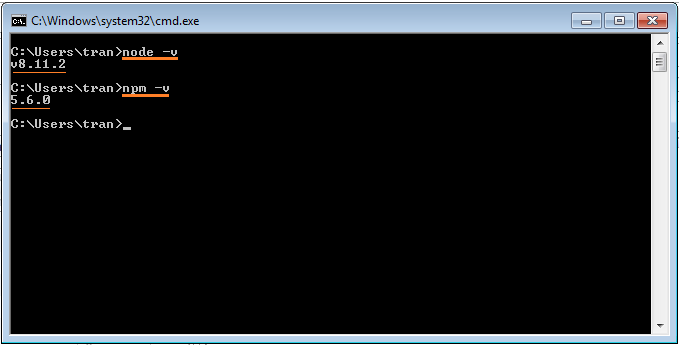






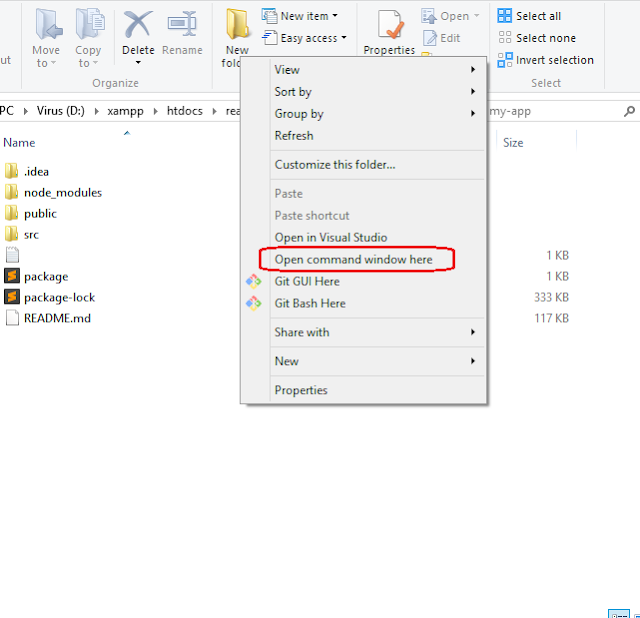
Bước 3: Kiểm tra cấu hình:

Mở **cửa sổ CMD** và thực thi các lệnh sau để **kiểm tra phiên bản** của **NodeJS và NPM:**

****

### Tạo project ReactJS

Bước 1:Tạo 1 folder để chứa project React, vào trong folder đè shift nhấn chuột phải và chọn như hình.

****

Bước 2: Tiếp theo gõ các dòng lệnh sau để tạo project:

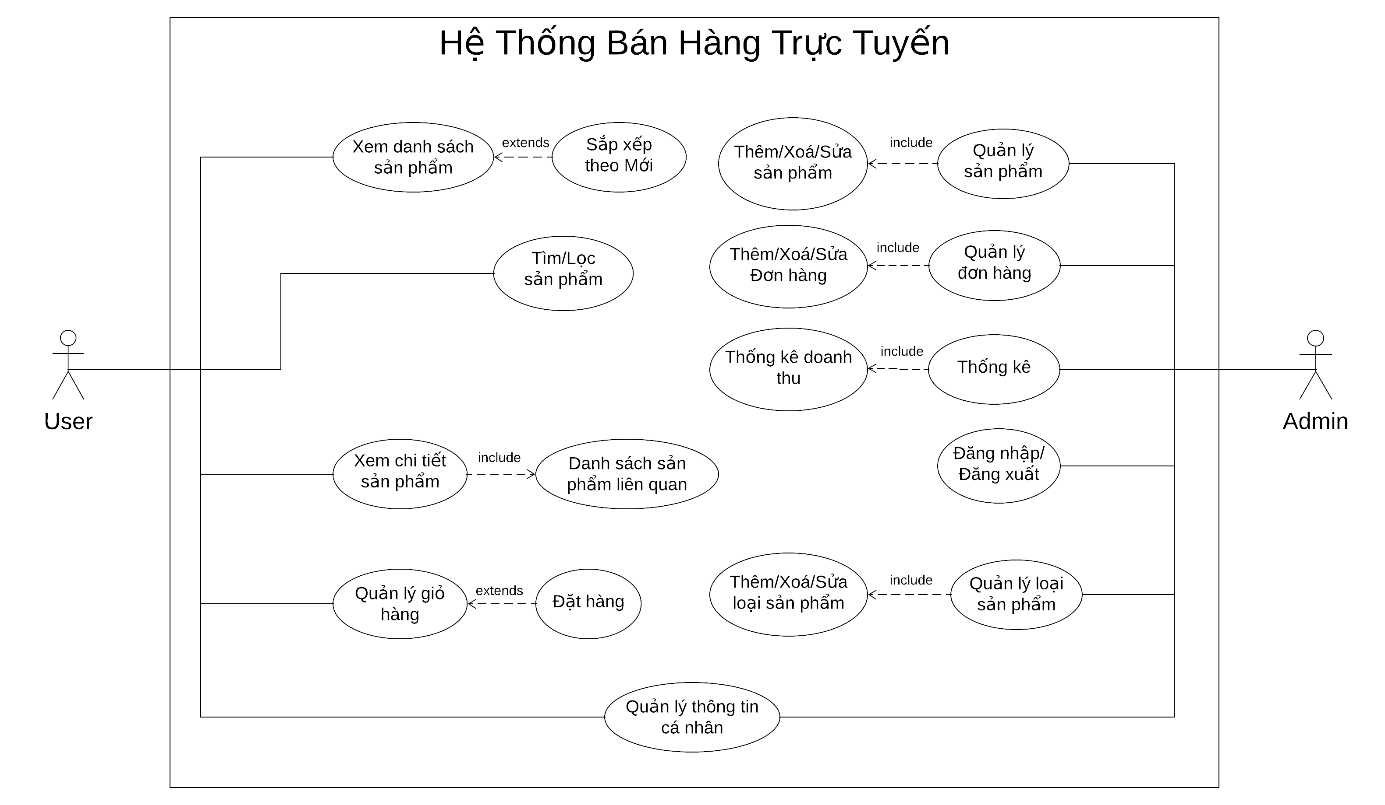
npm install -g create-react-app

create-react-app my-app

# Chương 4

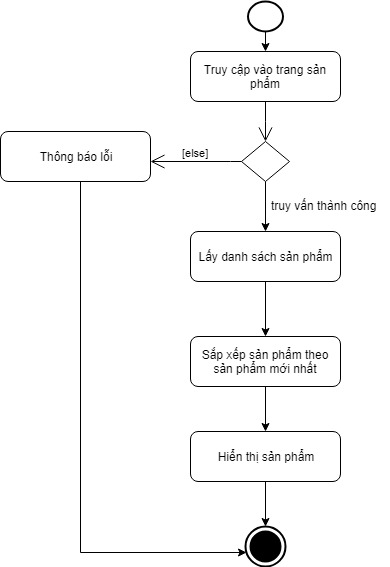
**TRIỂN KHAI HỆ THỐNG**

## Lược đồ usecase

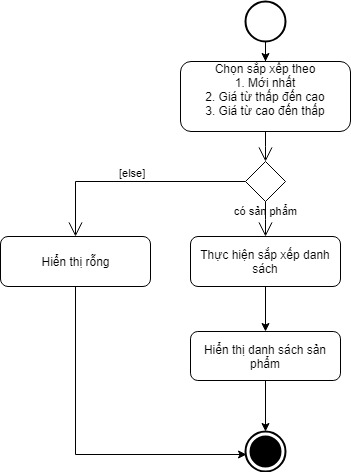


## Lược đồ activity

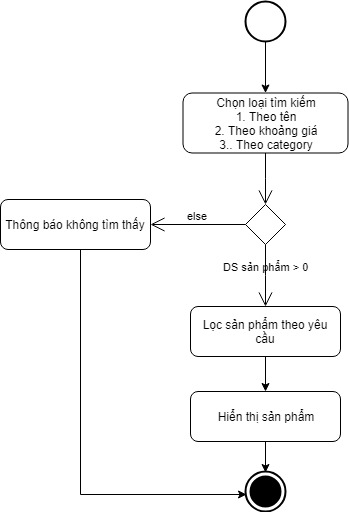
* Xem danh sách sản phẩm



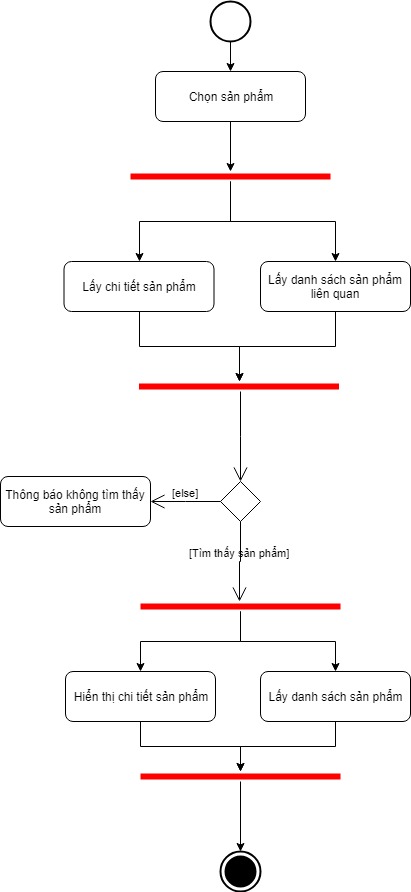
* Sắp xếp sản phẩm



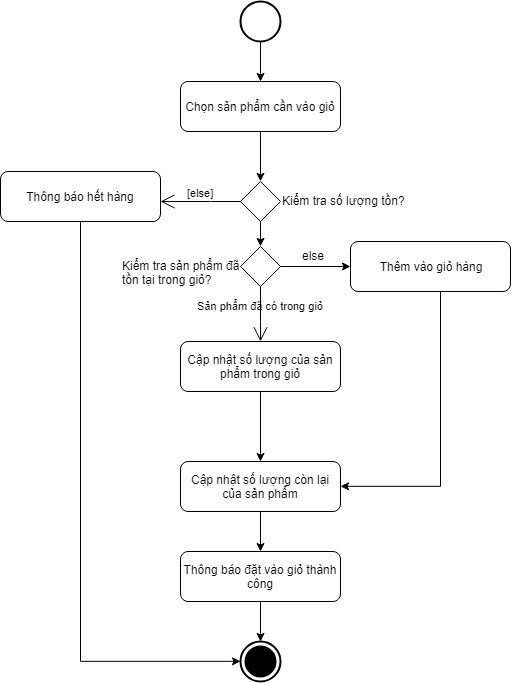
* Tìm kiếm/ Lọc sản phẩm



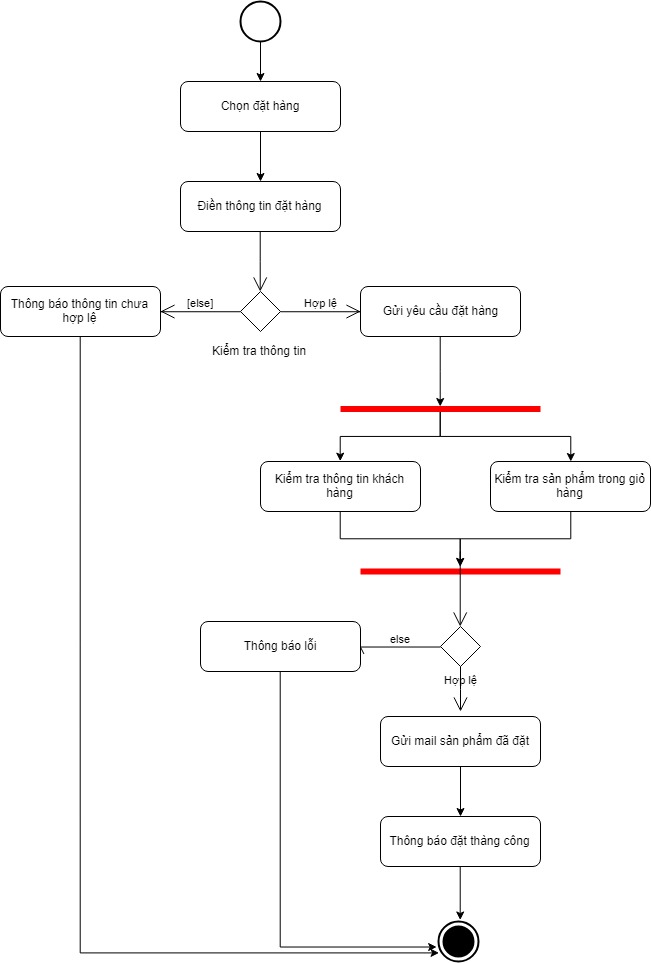
* Xem chi tiết sản phẩm



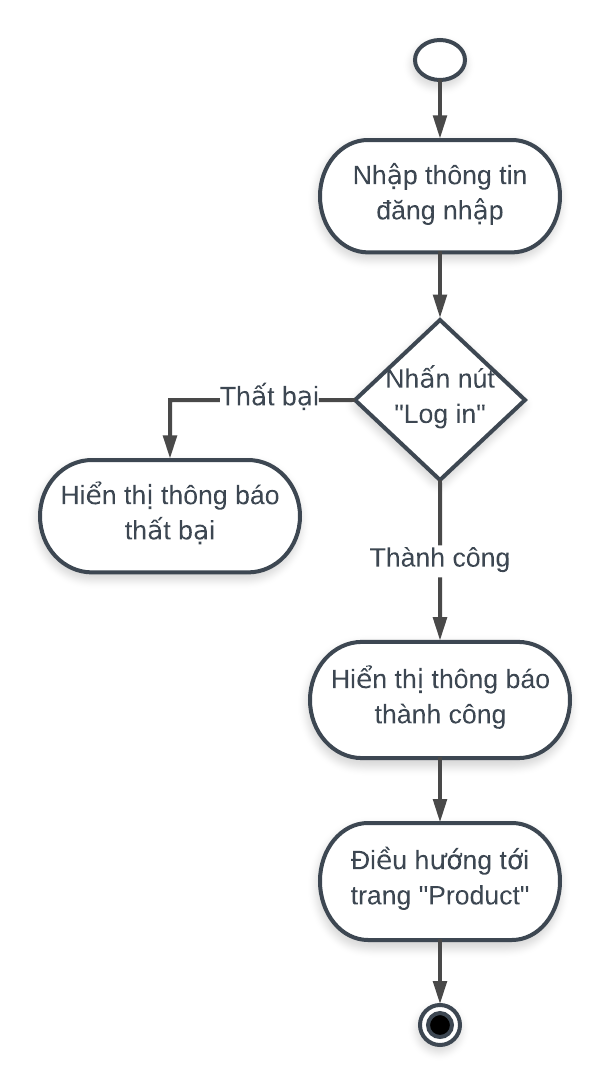
* Quản lý giỏ hàng



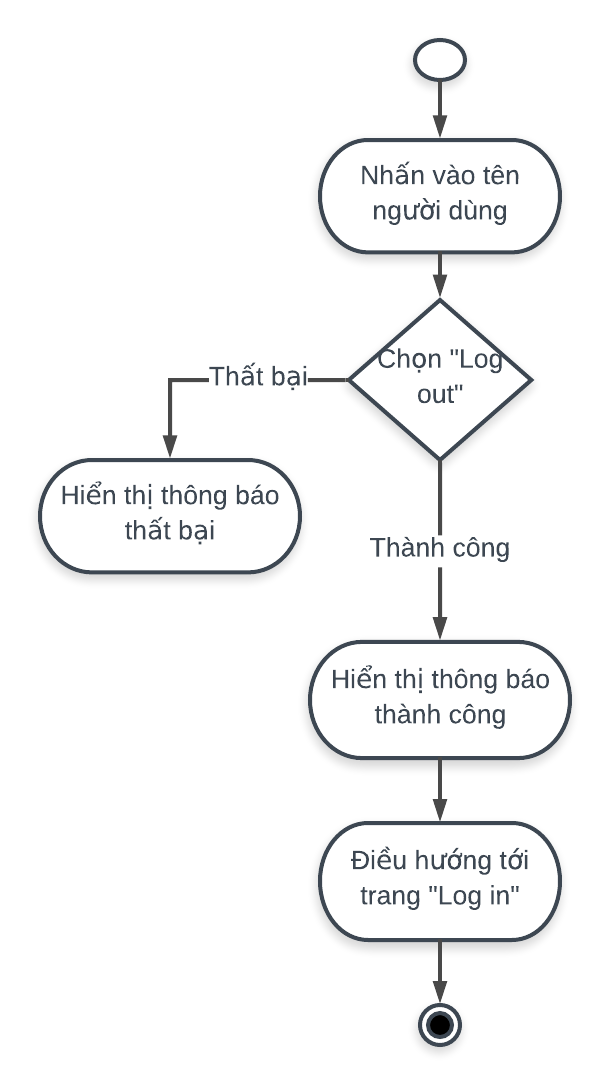
* Đặt hàng



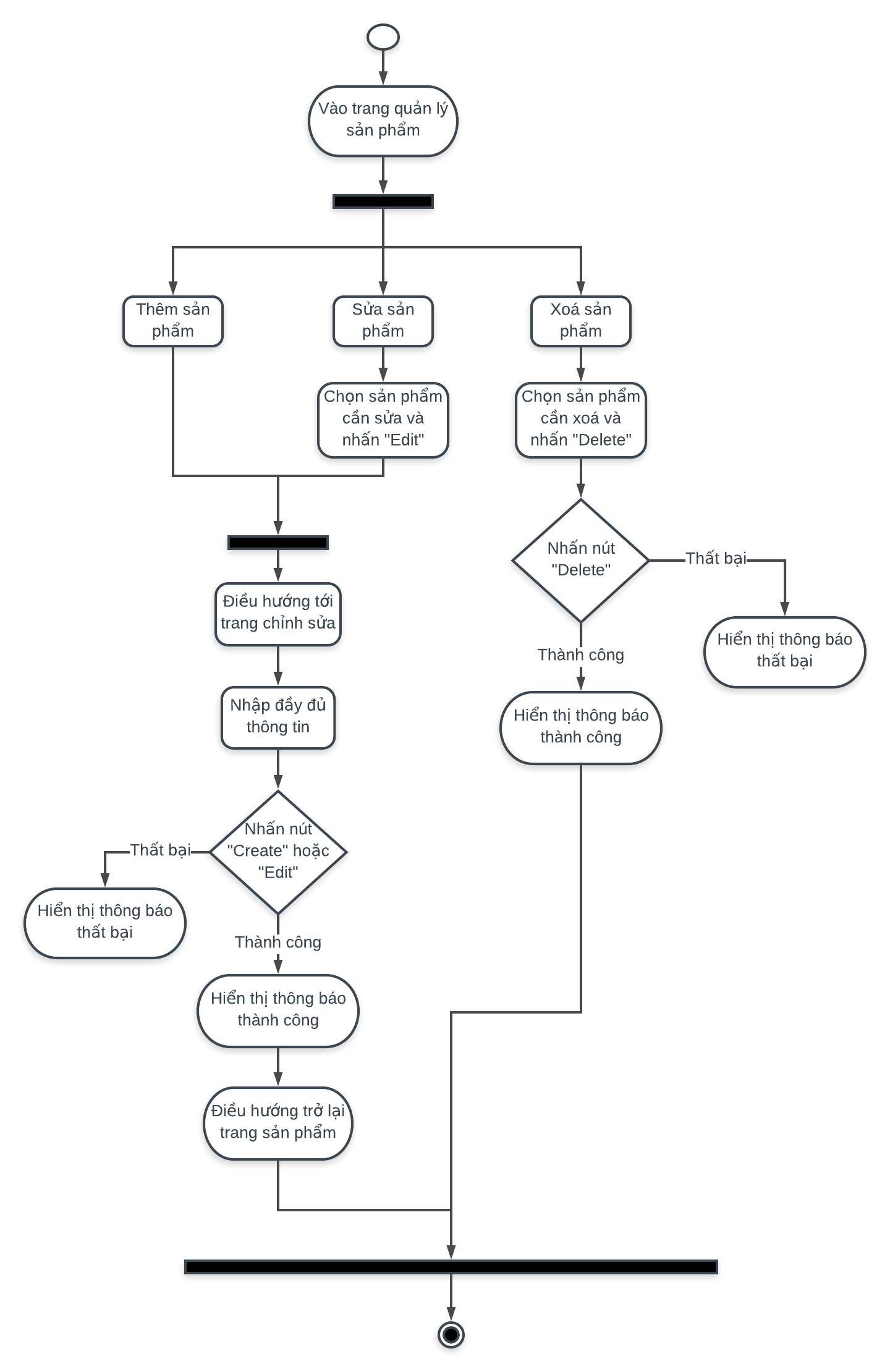
* Đăng nhập



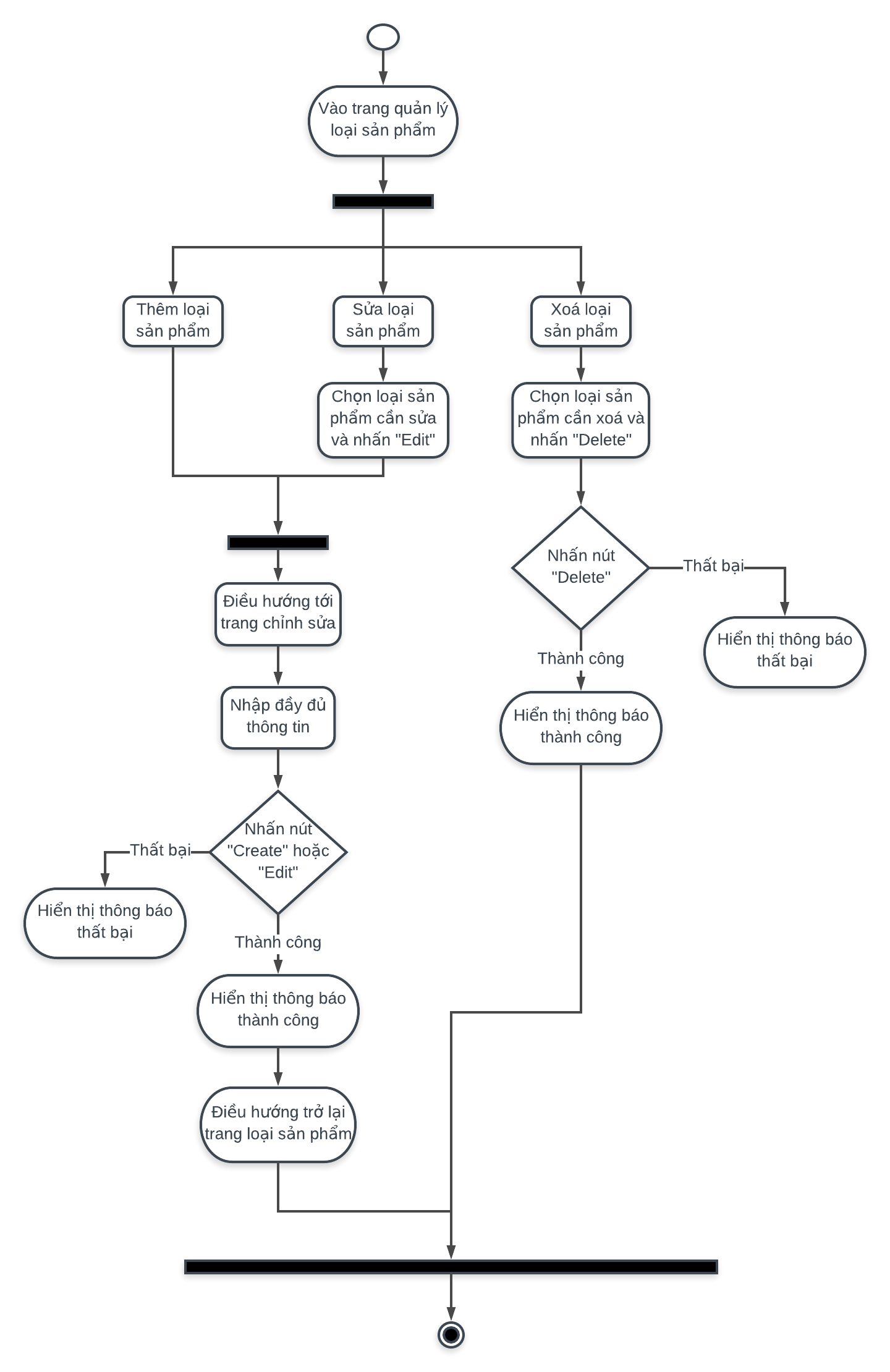
* Đăng xuất



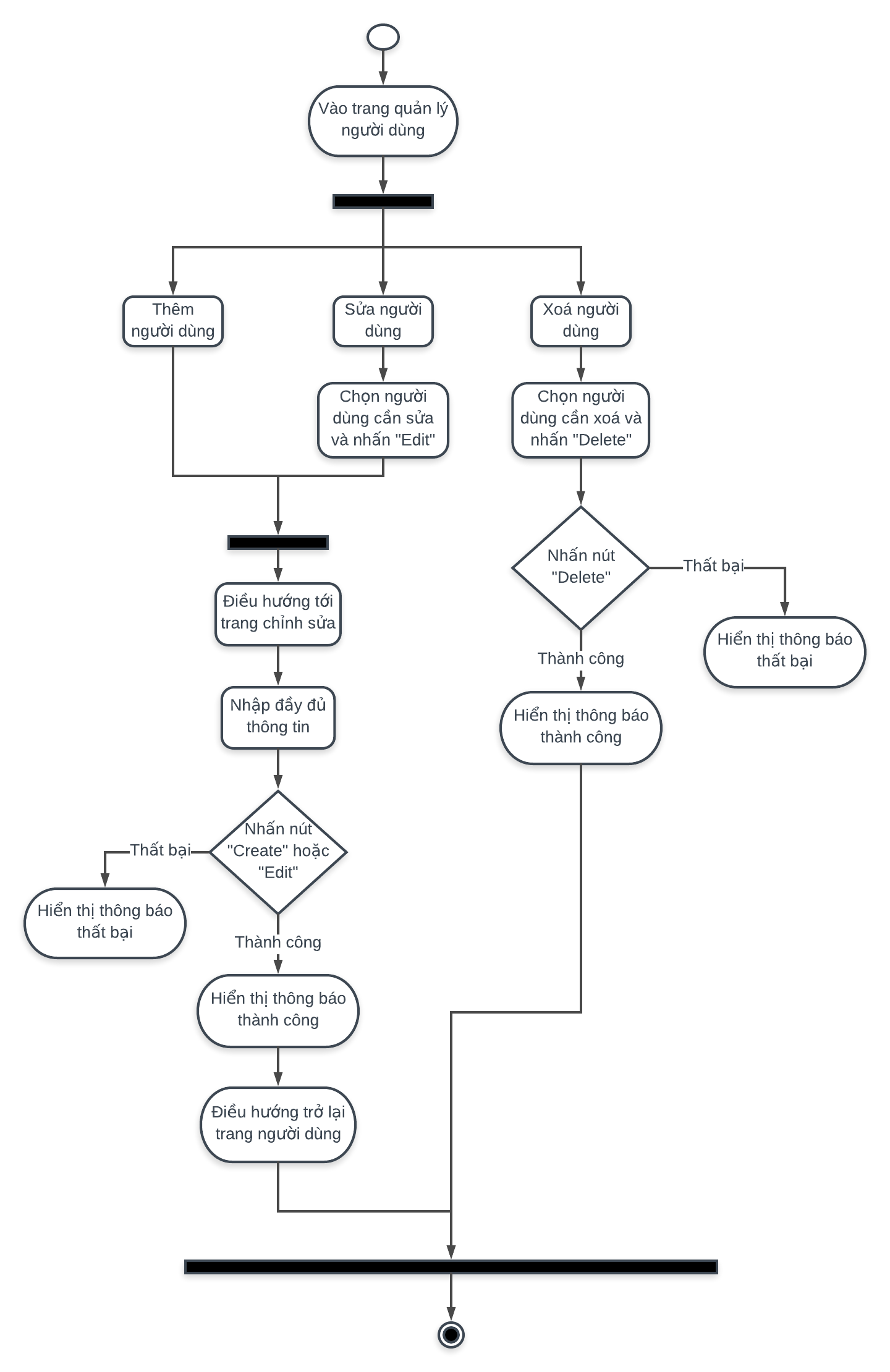
* Quản lý sản phẩm



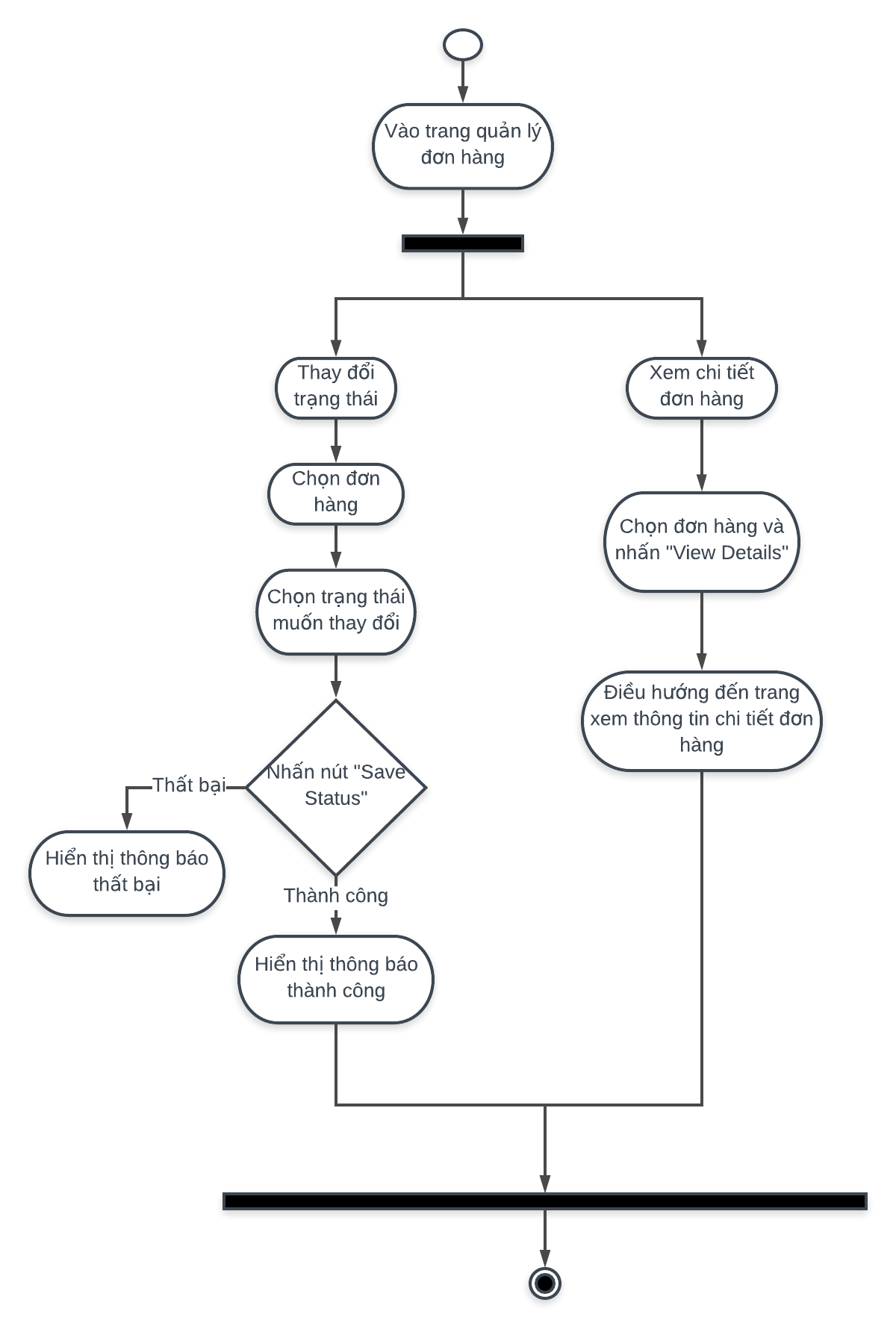
* Quản lý loại sản phẩm



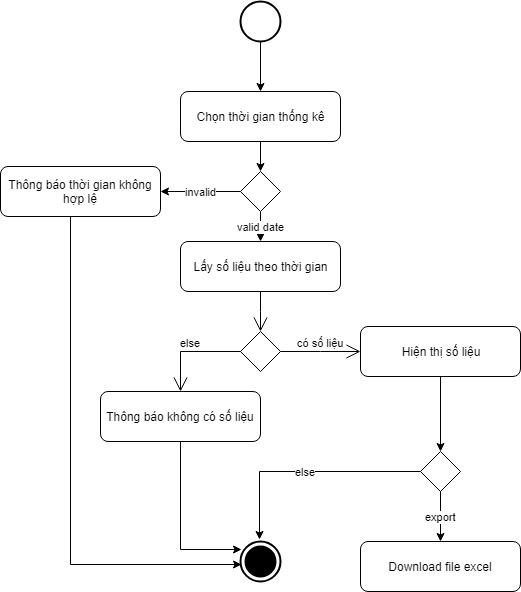
* Quản lý người dùng



* Quản lý đơn hàng



* Thống kê

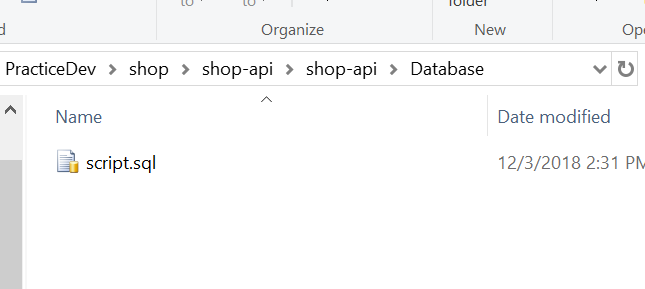


## Hướng dẫn cài đặt source code

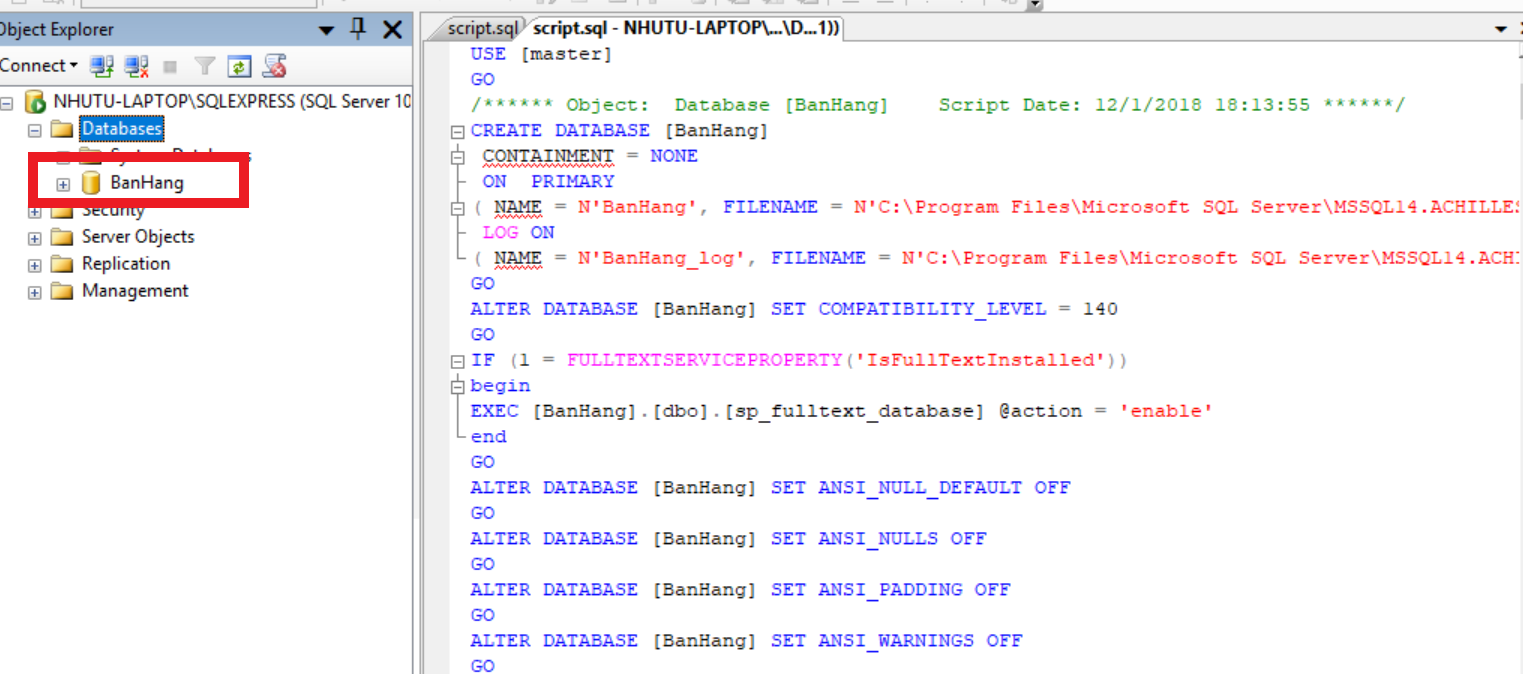
### Cài đặt cho backend

Cài đặt database – SQL Server:

Bước 1: Vào thư mục ‘shop\shop-api\shop-api\Database’ sau đó mở file script.sql

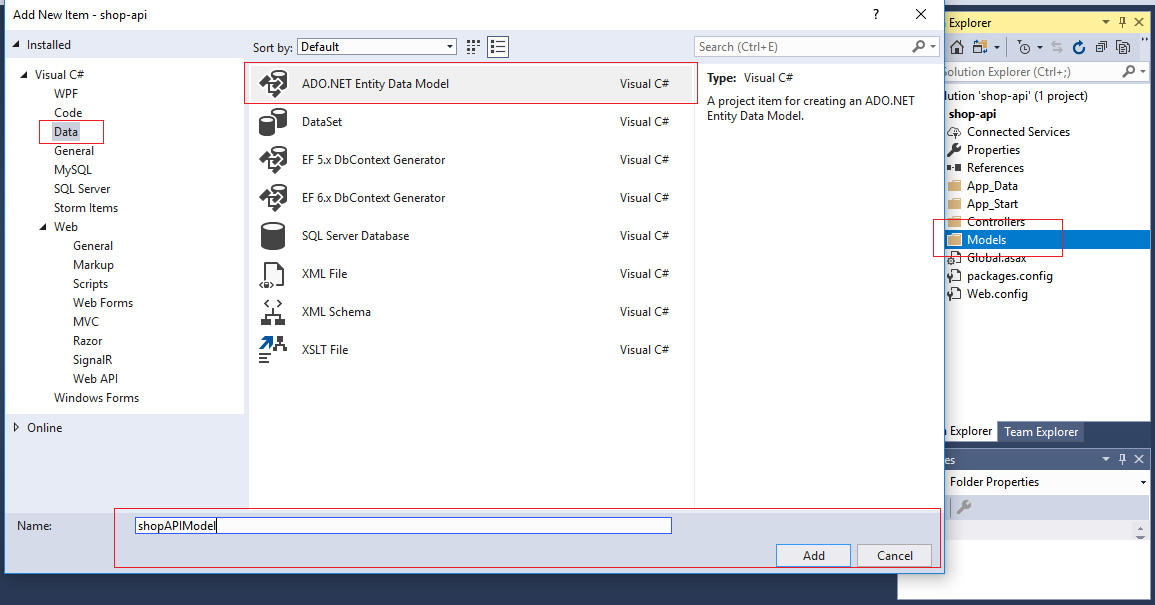


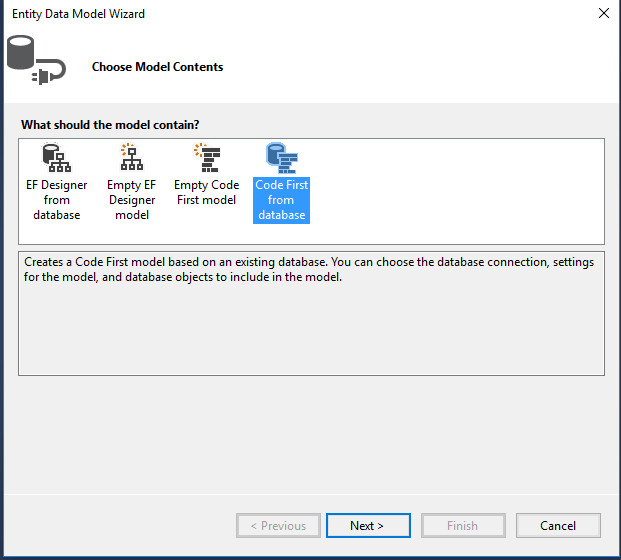
Bước 2: Thực thi các câu lệnh trong file ‘script.sql’ và ta sẽ có database tên ‘BanHang’



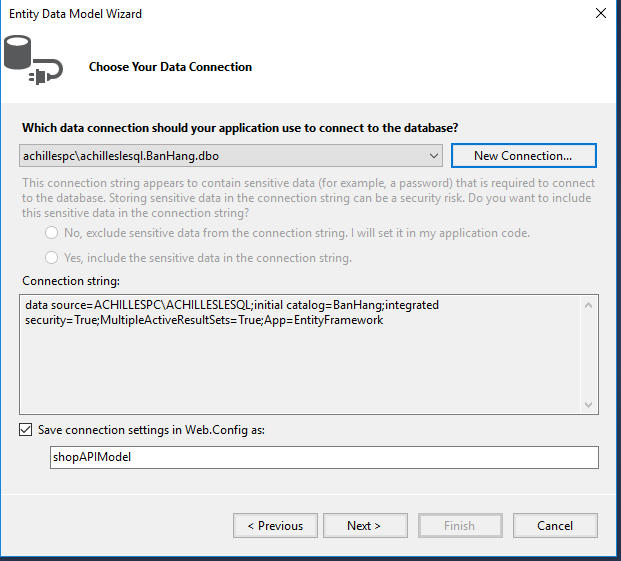
Mở visual studio lên , tạo thư mục project tên “shop-api”

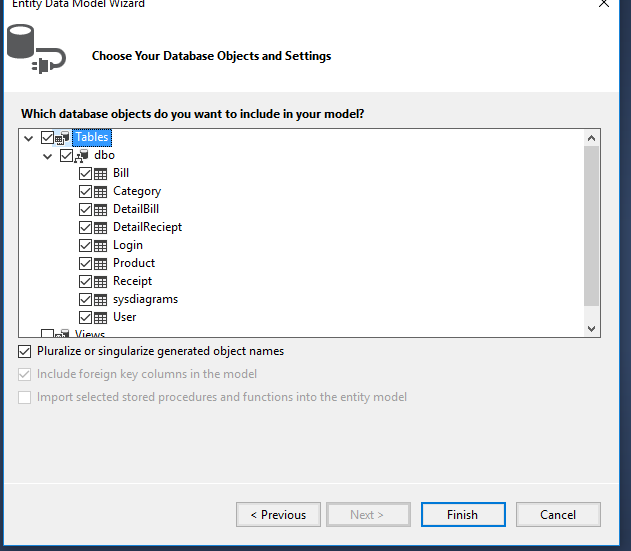
Click chuột phải vào folder models chọn “add new item” .

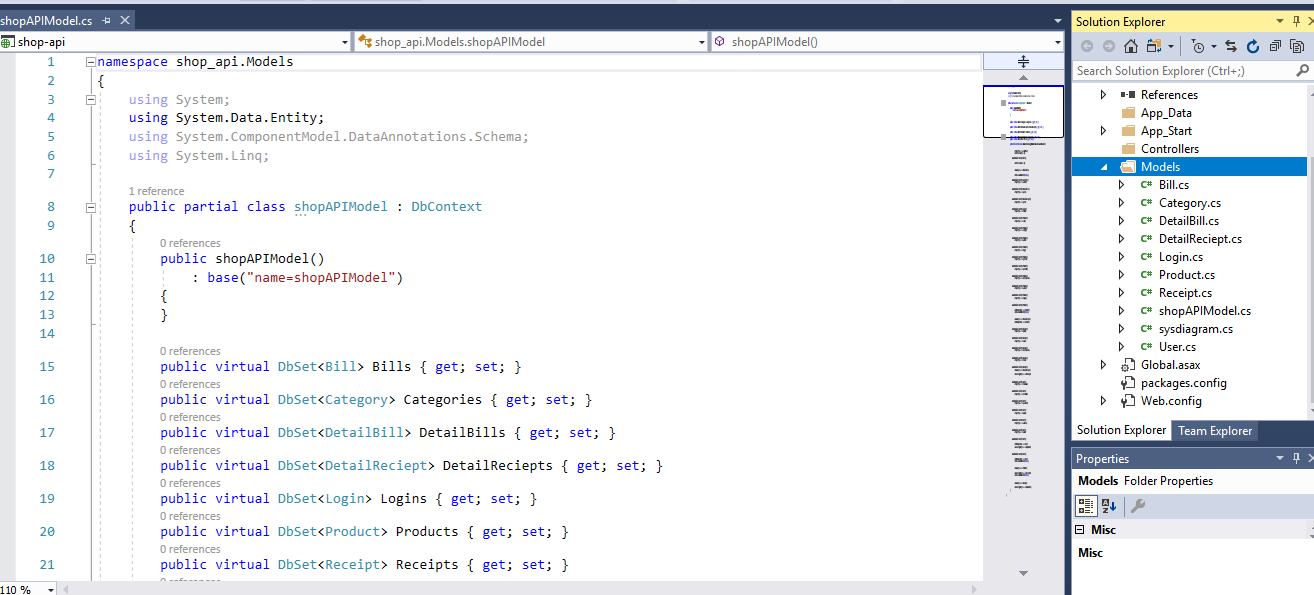












Sau các bước như thế này chúng ta đã có thể kết nối dữ liệu với database thông qua class shopAPIModel.

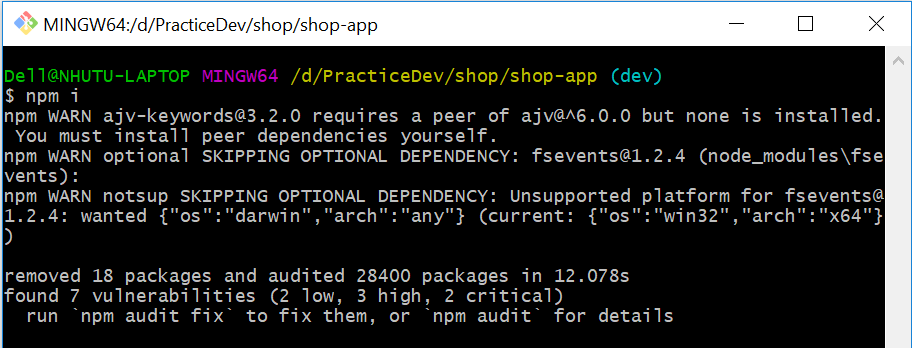
### Cài đặt cho frontend

Bước 1: Mở thư mục ‘shop-app’

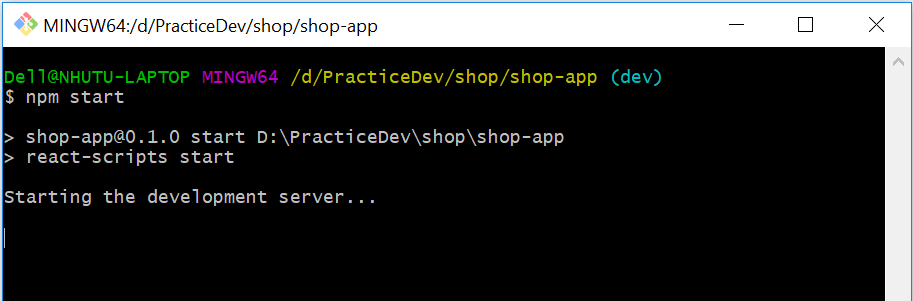


Bước 2: Chạy ‘npm i’ trong thư mục ‘shop-app’ bằng cửa sổ cmd hoặc git bash(nếu có)

Dòng lệnh này dùng để cài đặt những gói thư viện có trong file **package.json**



Bước 3: Chạy tiếp dòng lệnh ‘npm start’ để khởi động website với url: http://localhost:3000

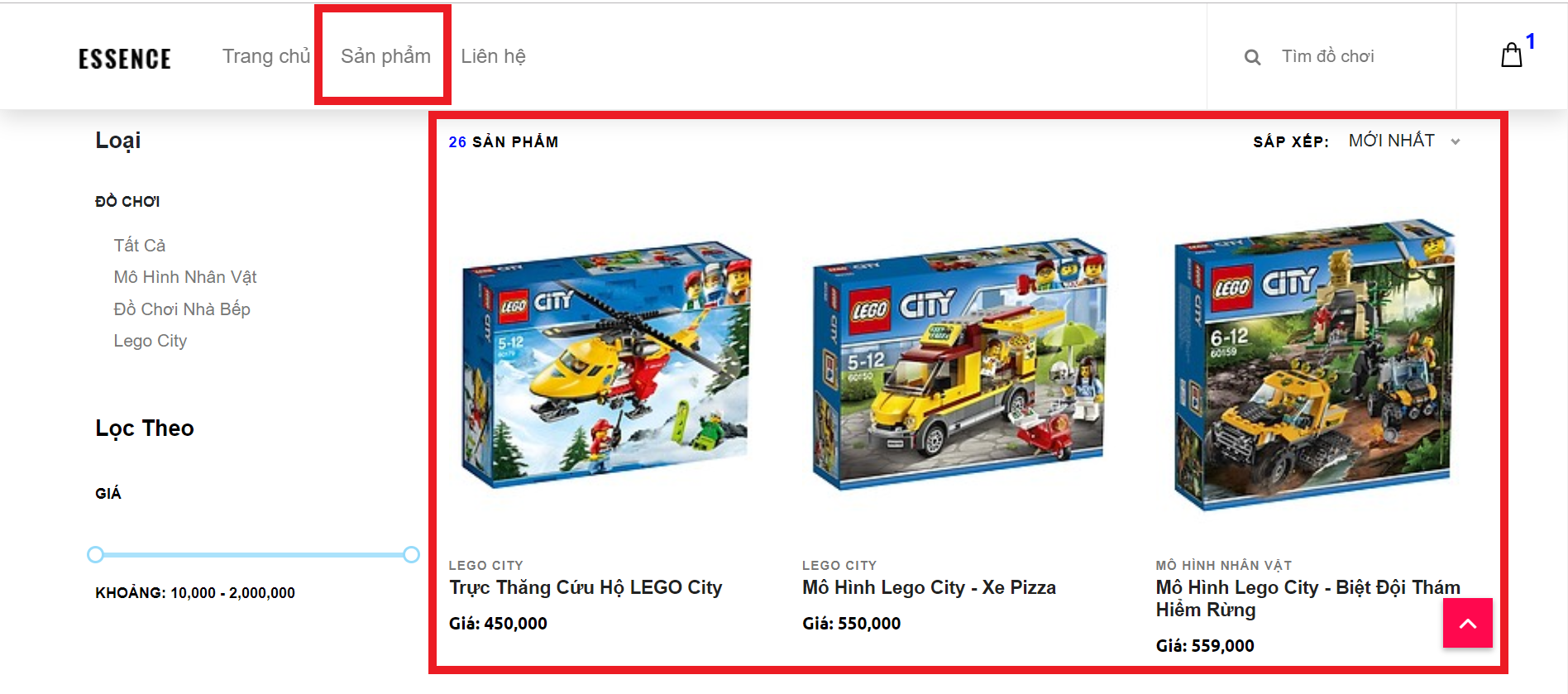


## Hướng dẫn sử dụng chức năng

### Chức năng User

#### Xem danh sách sản phẩm:

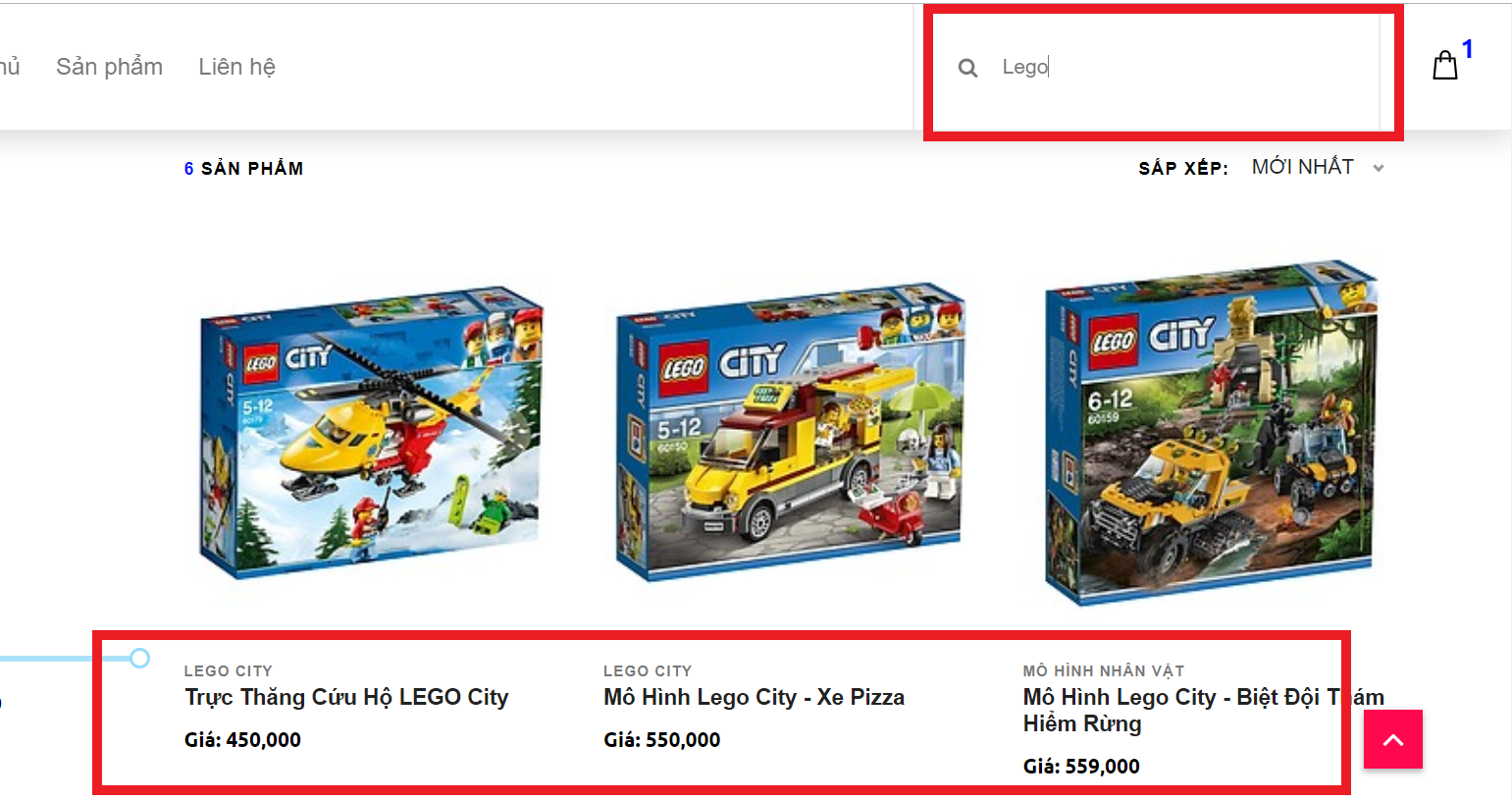
* + - Để xem sản phẩm, người dùng vào link <http://localhost:3000/san-pham> hoặc click vào thanh menu



#### Tìm kiếm/ lọc sản phẩm (truy cập vào link <http://localhost:3000/san-pham>)

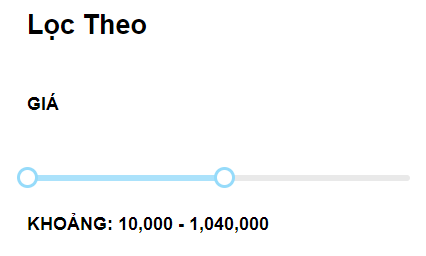
* + - * Tìm kiếm theo từ khóa

Nhập tên cần tìm vào ô tìm kiếm phía trên phải



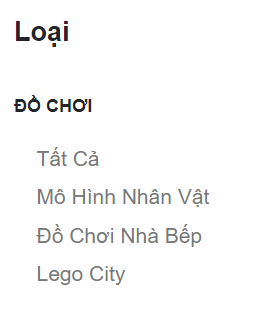
* + - * Lọc theo giá

Có khoảng giá 10000 – 2000000. Người dùng có thể kéo giá trong khoảng đó



* + - * Lọc theo loại sản phẩm

Lựa chọn loại sản phẩm mong muốn



#### Xem chi tiết sản phẩm:

Để xem chi tiết người dùng sẽ click vào tên sản phẩm muốn xem



Hoặc truy cập link [http://localhost:3000/chi-tiet?id={mã](http://localhost:3000/chi-tiet?id=%7bmã) sản phẩm}

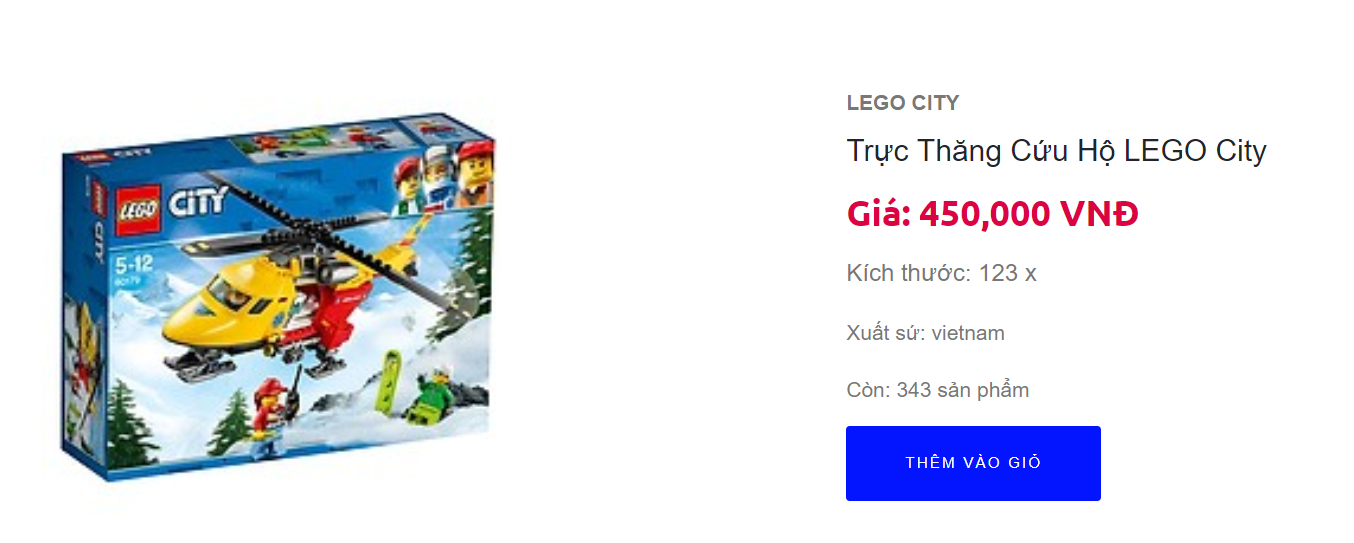
#### Quản lý giỏ hàng:

* + - Thêm giỏ hàng

Trong danh sách sản phẩm khi rê vào 1 sản phẩm sẽ có nút thêm giỏ hàng

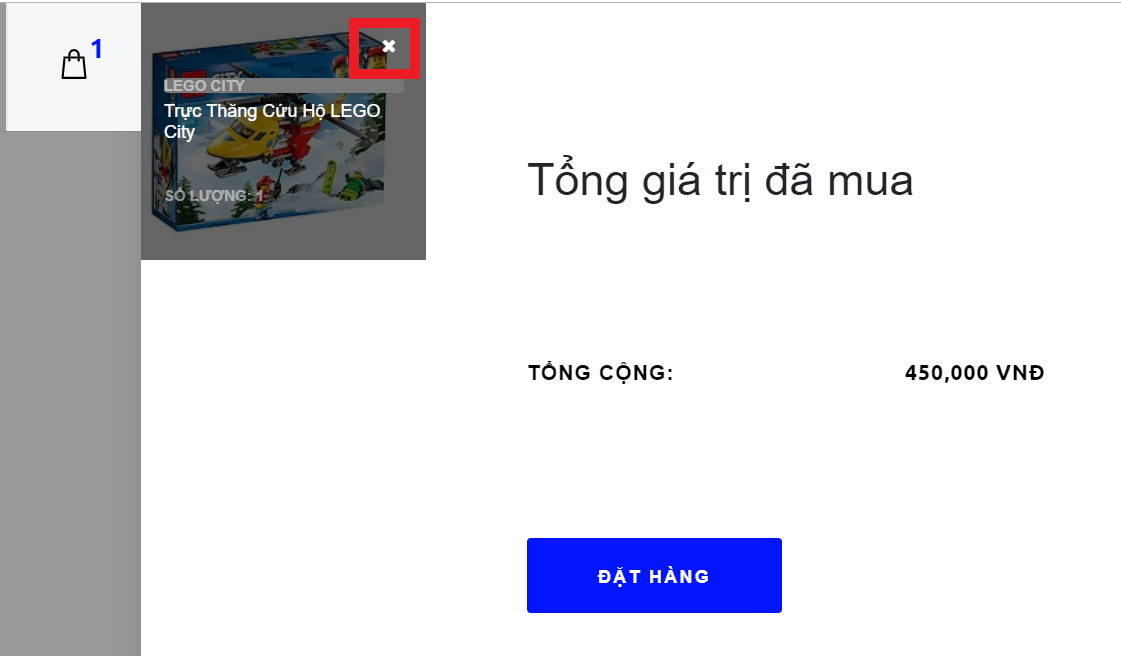
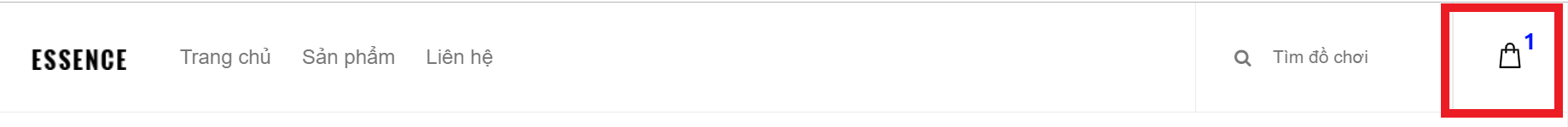


Trong trang chi tiết sản phẩm có nút thêm vào giỏ hàng



* + - Xóa sản phẩm trong giỏ hàng

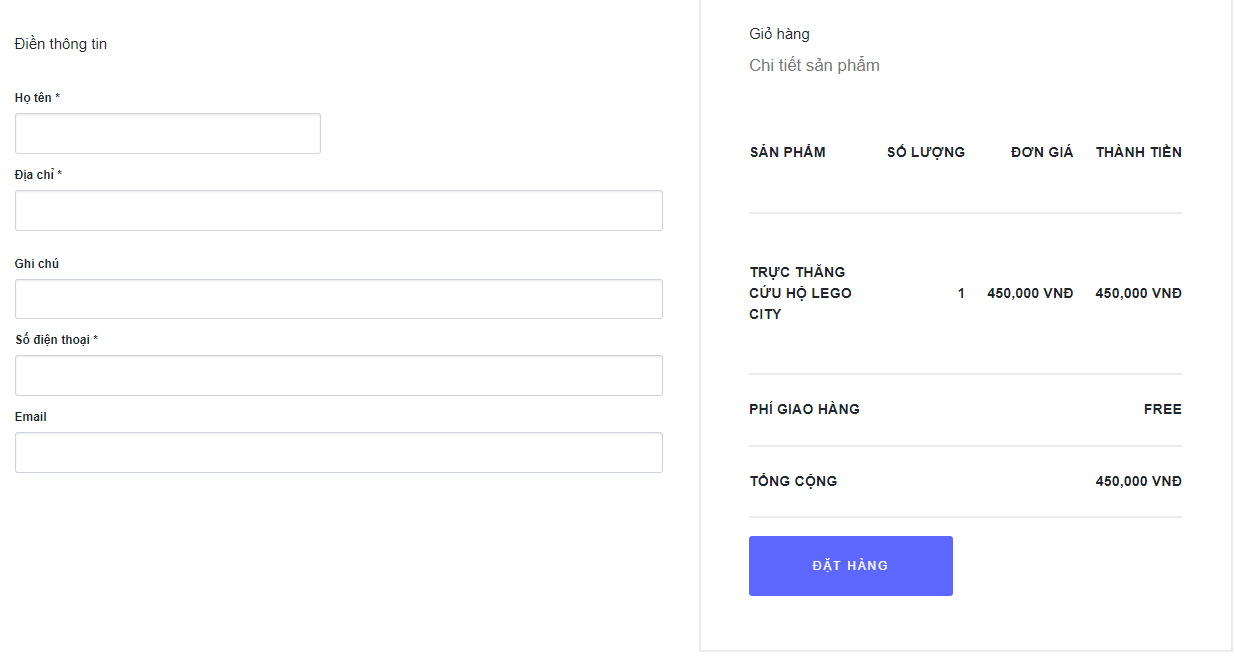
Để xóa sản phẩm người dùng click vào giỏ hàng và click vào dấu (x) tại sản phẩm muốn xóa

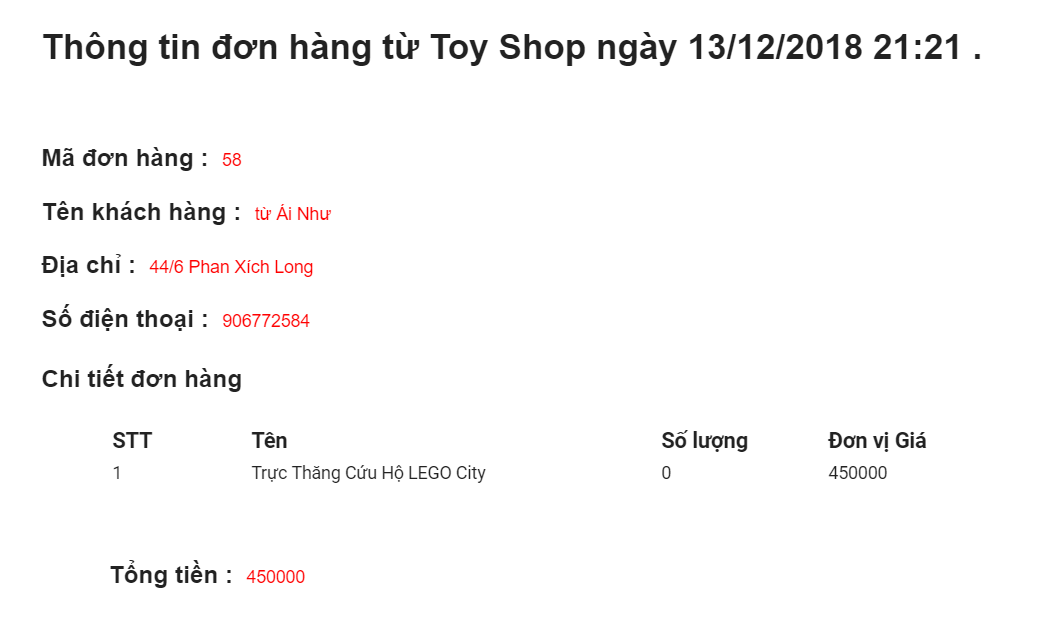


#### Đặt hàng:

* + - Để đặt hàng người dùng cần mở giỏ hàng và click vào nút đặt hàng
    - Điền thông tin và click vào nút đặt hàng
    - Check lại đơn hàng thông qua mail đã điền





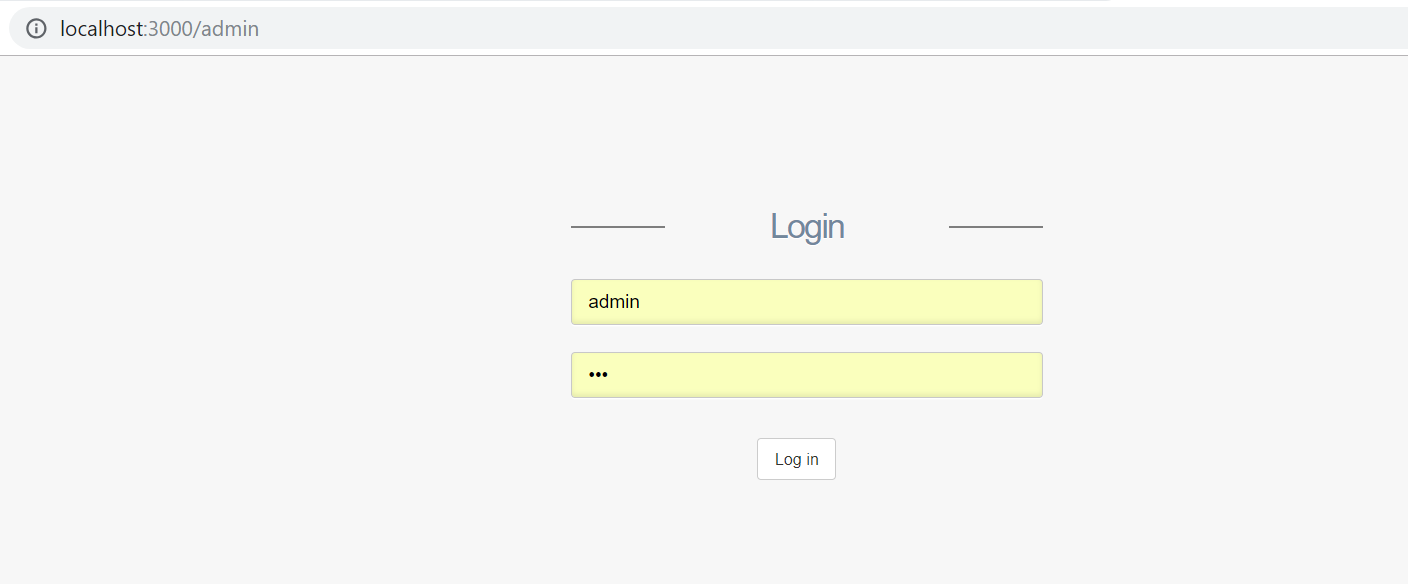


### Chức năng Admin

#### Đăng nhập – Đăng xuất:

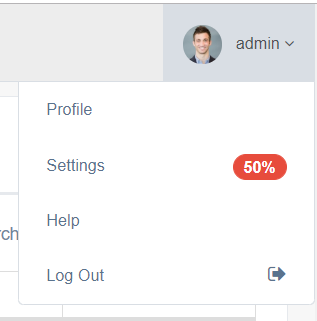
Để đăng nhập, người dùng vào link <http://localhost:3000/admin>

Gõ username/password : admin/123



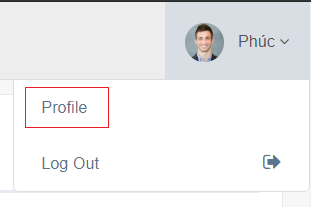
Để đăng xuất, người dùng vào link bất kỳ của trang admin <http://localhost:3000/admin>

Nhấn vào tên người dùng và chọn Log out

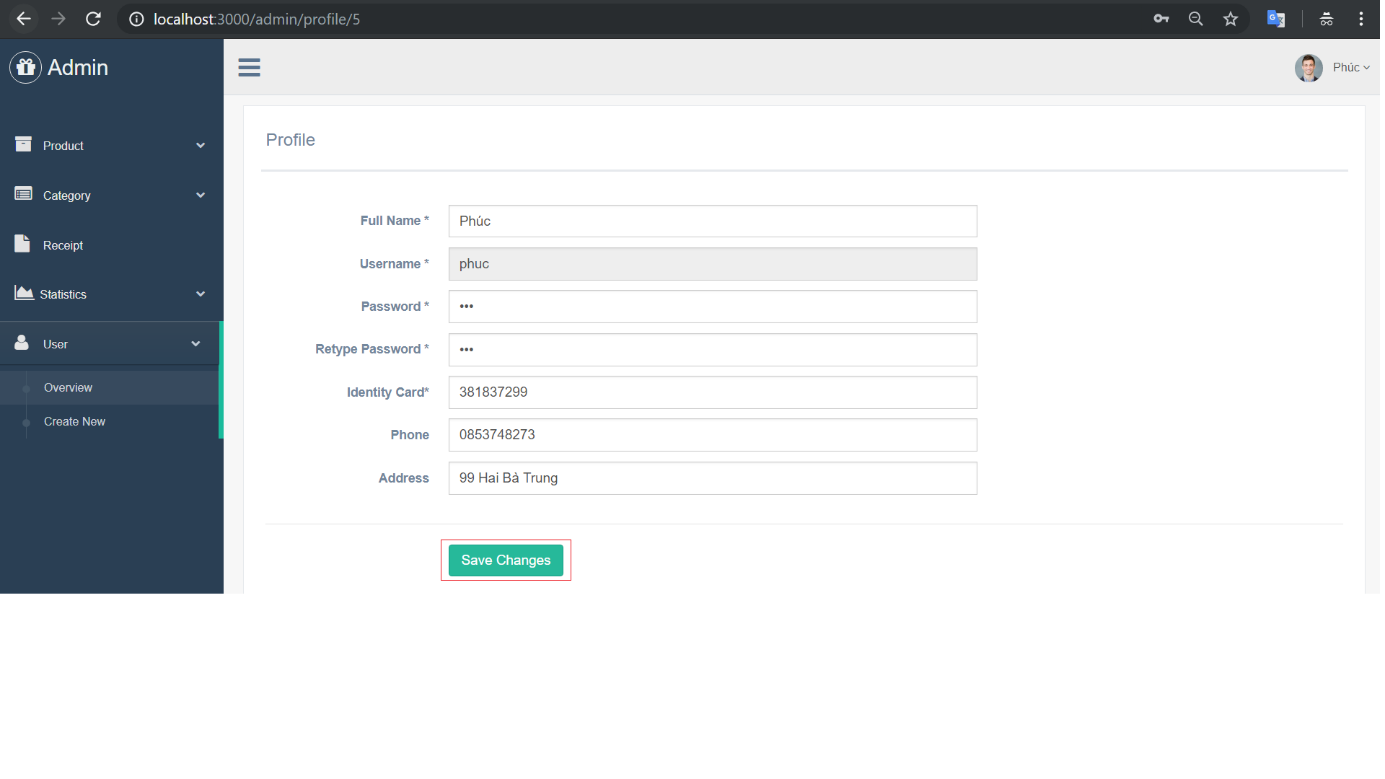


#### Cập nhật thông tin cá nhân:

Để cập nhật thông tin cá nhân, người dùng nhấn vào tên mình và chọn “Profile”.

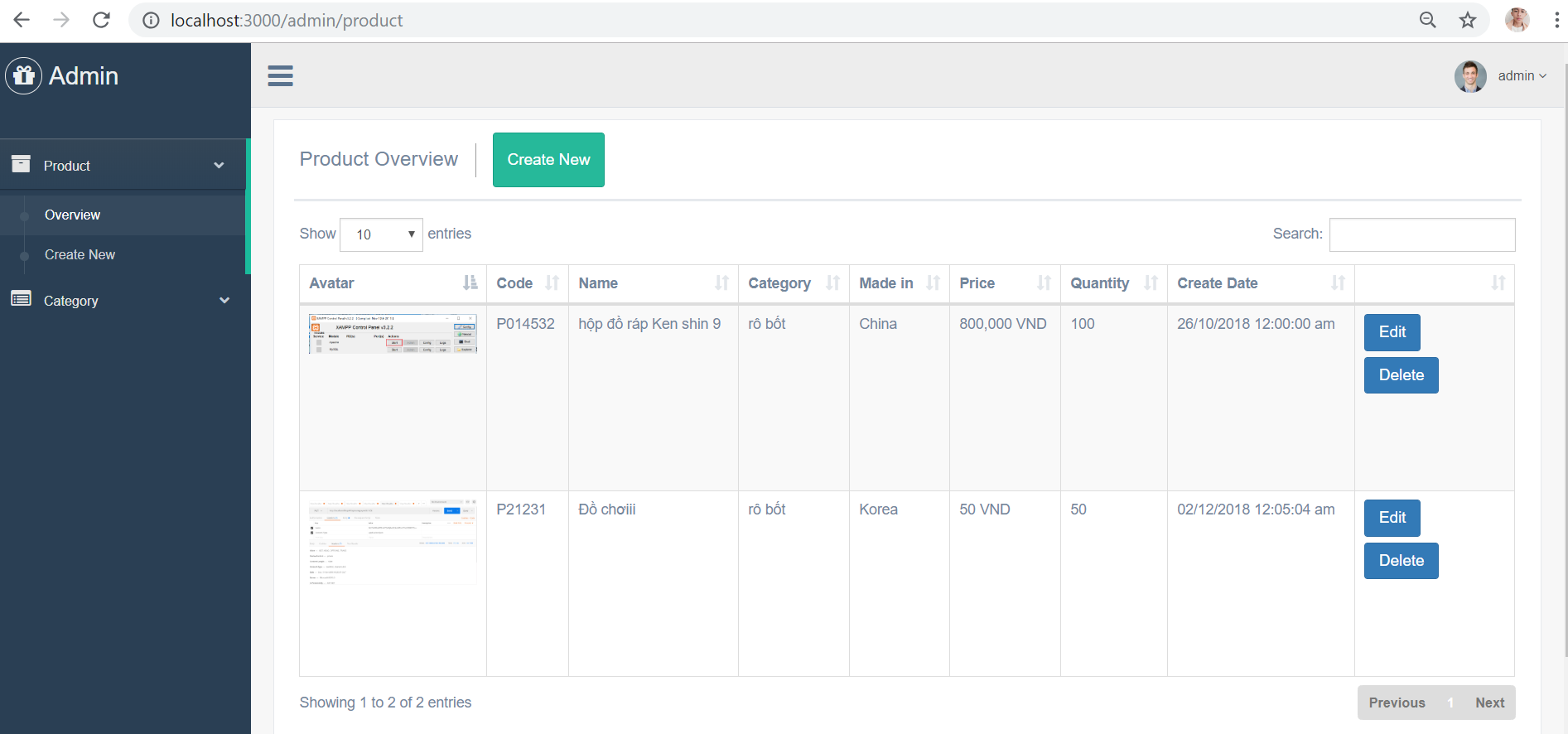


Sau đó điền đầy đủ thông tin cần sửa và nhấn nút “Save changes” để hệ thống tiến hành lưu thông tin mới.



#### Quản lý sản phẩm

Để quản lý sản phẩm và xem toàn bộ sản phẩm, người dùng vào link <http://localhost:3000/admin/product>

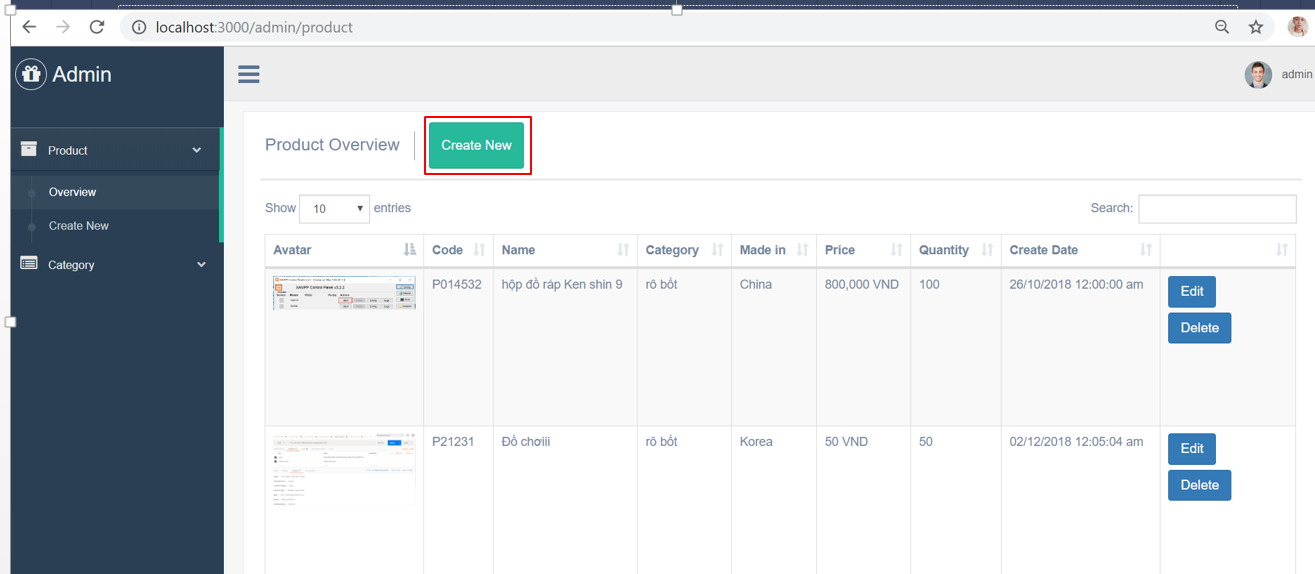


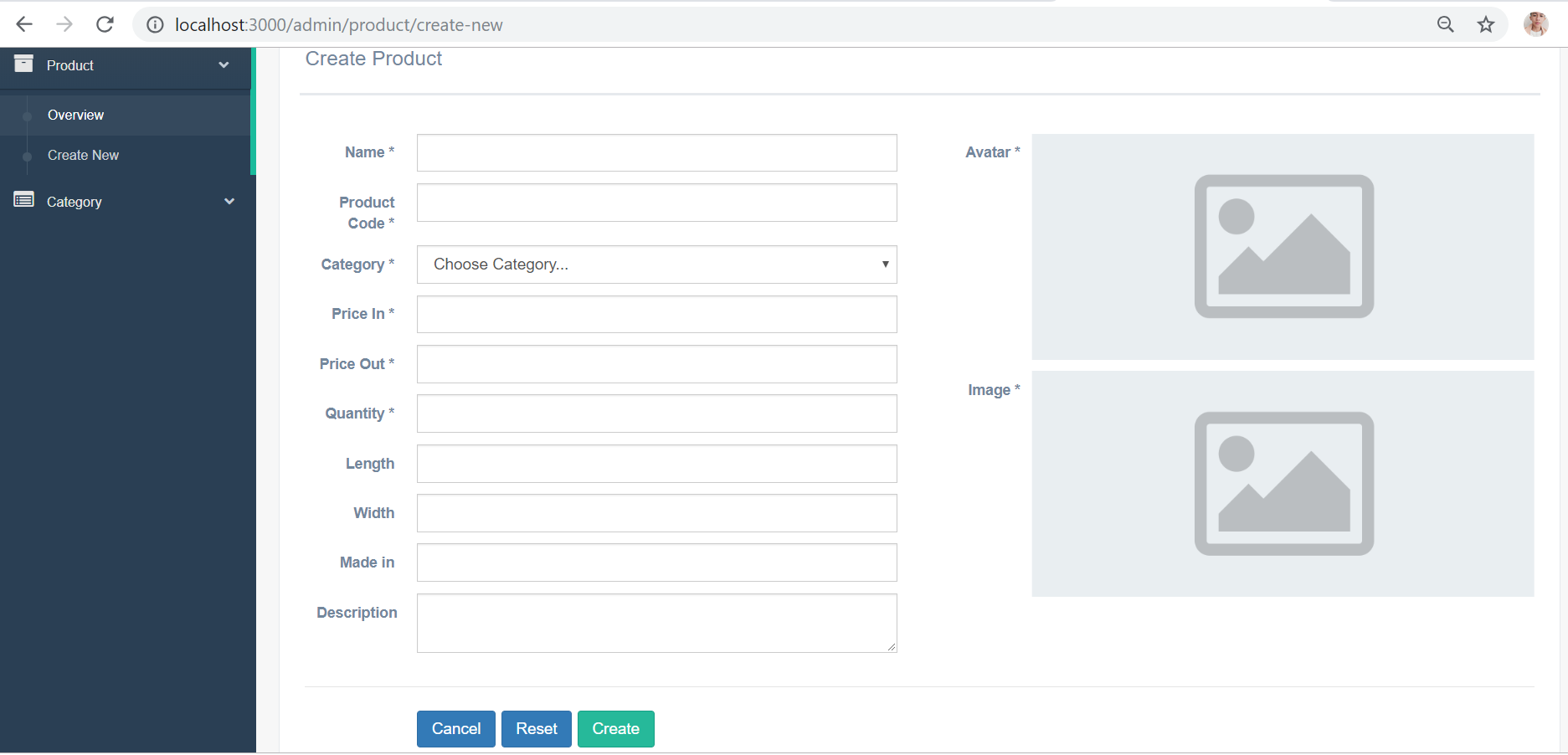
* + - * Thêm sản phẩm

Để thêm sản phẩm, người dùng vào link <http://localhost:3000/admin/product/create-new> .

Hoặc người dùng có thể vào link <http://localhost:3000/admin/product> và nhấn Create New.

Điền đầy đủ các thông tin và nhấn nút Create



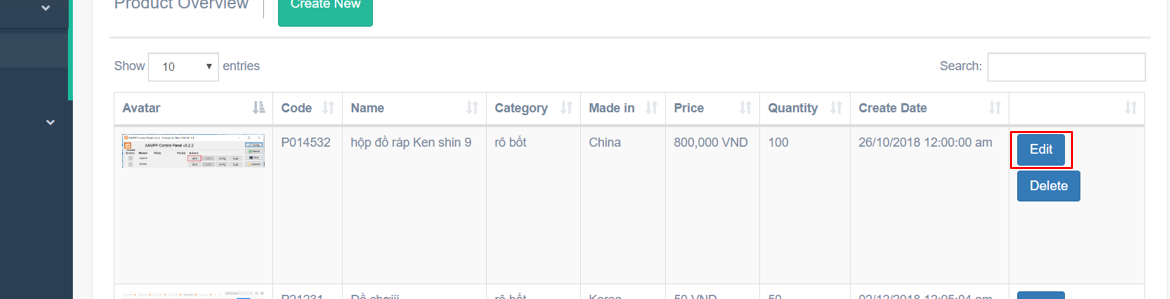


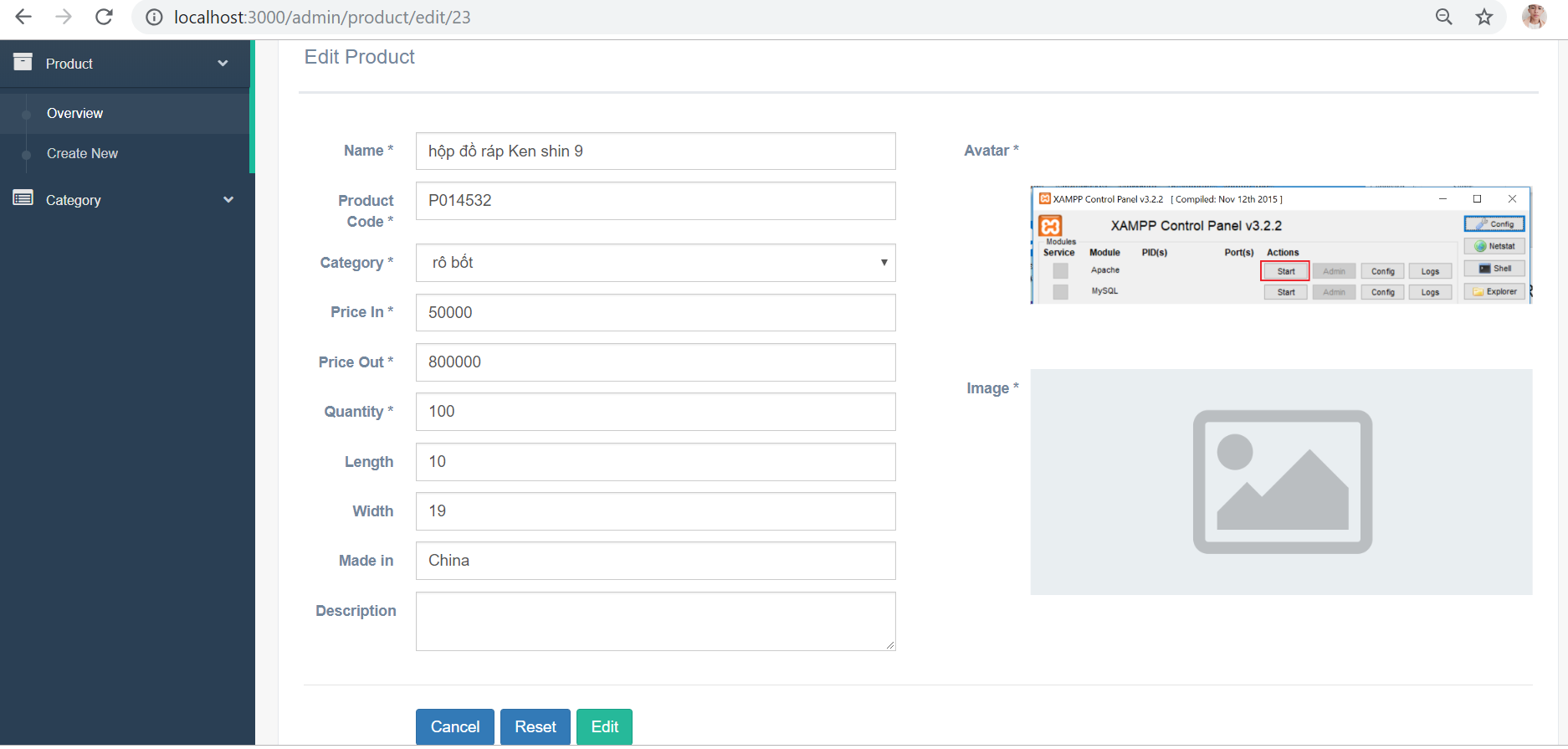
* + - * Sửa sản phẩm

Để sửa sản phẩm, người dùng vào link <http://localhost:3000/admin/product/edit/1> với 1 là id của sản phẩm.

Hoặc người dùng có thể vào link <http://localhost:3000/admin/product> chọn sản phẩm muốn sửa và nhấn edit.

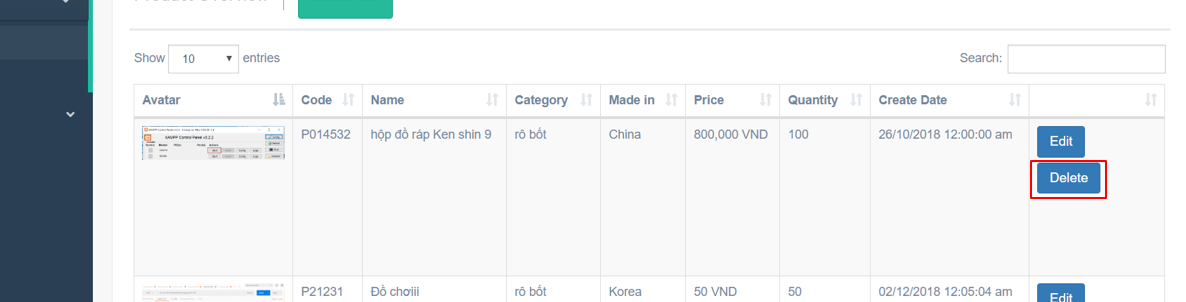
Sau đó điền đầy đủ các thông tin muốn sửa và nhấn nút Edit.





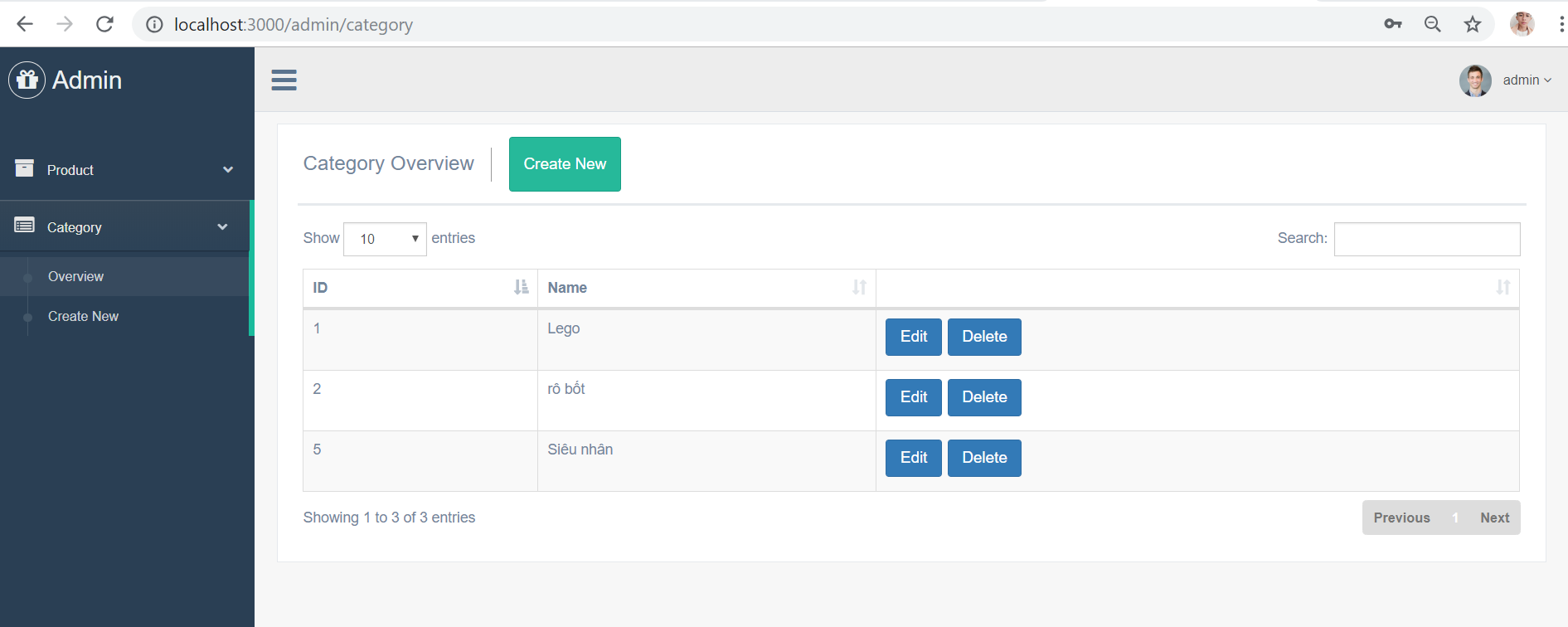
* + - * Xóa sản phẩm

Để xoá sản phẩm, người dùng vào link <http://localhost:3000/admin/product> chọn sản phẩm muốn xoá và nhấn delete.



#### Quản lý loại sản phẩm:

Để quản lý loại sản phẩm và xem toàn bộ loại sản phẩm, người dùng vào link <http://localhost:3000/admin/category>

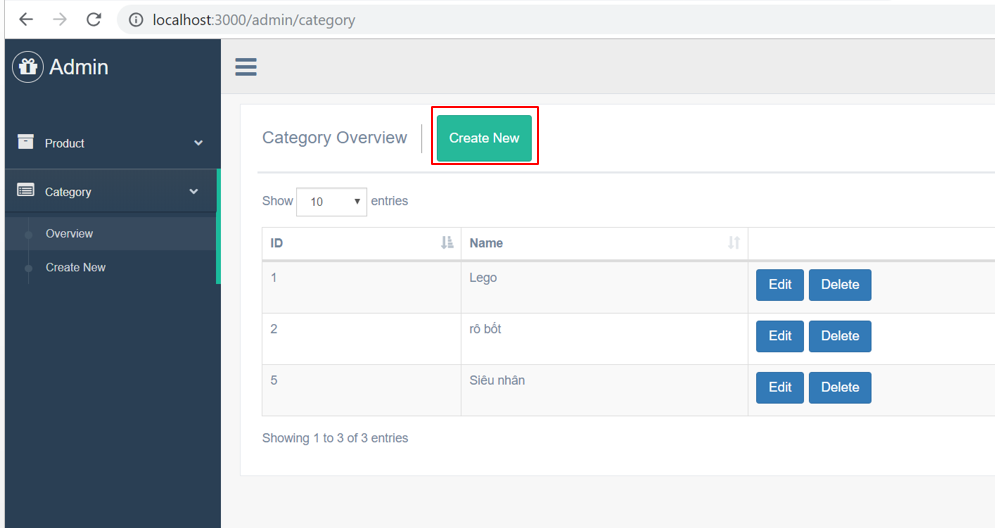


* + - * Thêm loại sản phẩm

Để thêm loại sản phẩm, người dùng vào link <http://localhost:3000/admin/category/create-new> .

Hoặc người dùng có thể vào link <http://localhost:3000/admin/category> và nhấn Create New.

Điền đầy đủ các thông tin và nhấn nút Create.



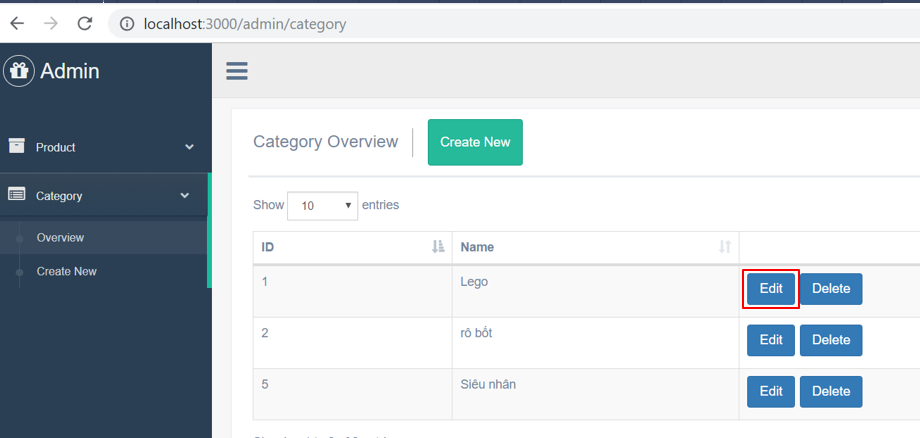


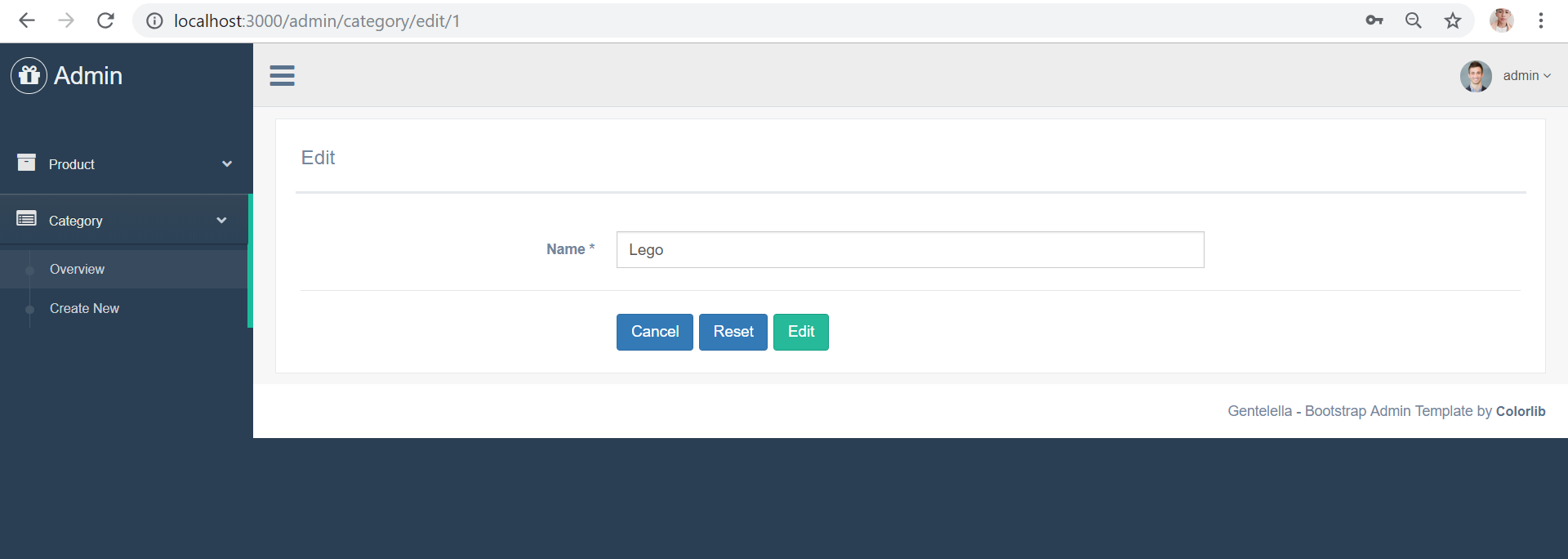
* + - * Sửa loại sản phẩm

Để sửa loại sản phẩm, người dùng vào link <http://localhost:3000/admin/category/edit/1> với 1 là id của loại sản phẩm.

Hoặc người dùng có thể vào link <http://localhost:3000/admin/category> chọn loại sản phẩm muốn sửa và nhấn edit.

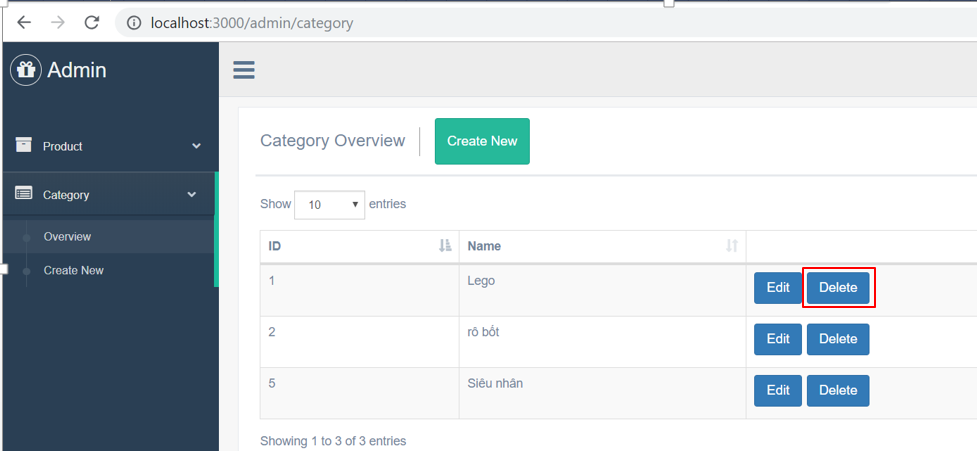
Sau đó điền đầy đủ các thông tin muốn sửa và nhấn nút Edit.





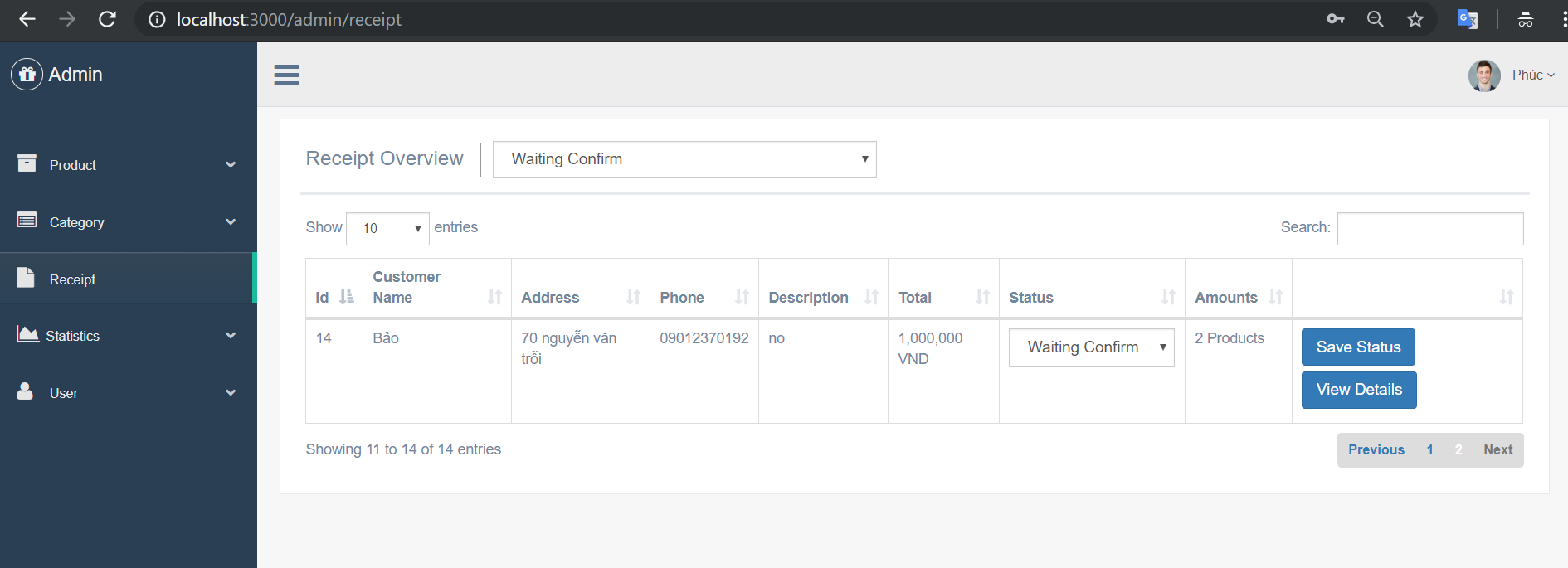
* + - * Xóa loại sản phẩm

Để xoá loại sản phẩm, người dùng vào link <http://localhost:3000/admin/category> chọn loại sản phẩm muốn xoá và nhấn delete.

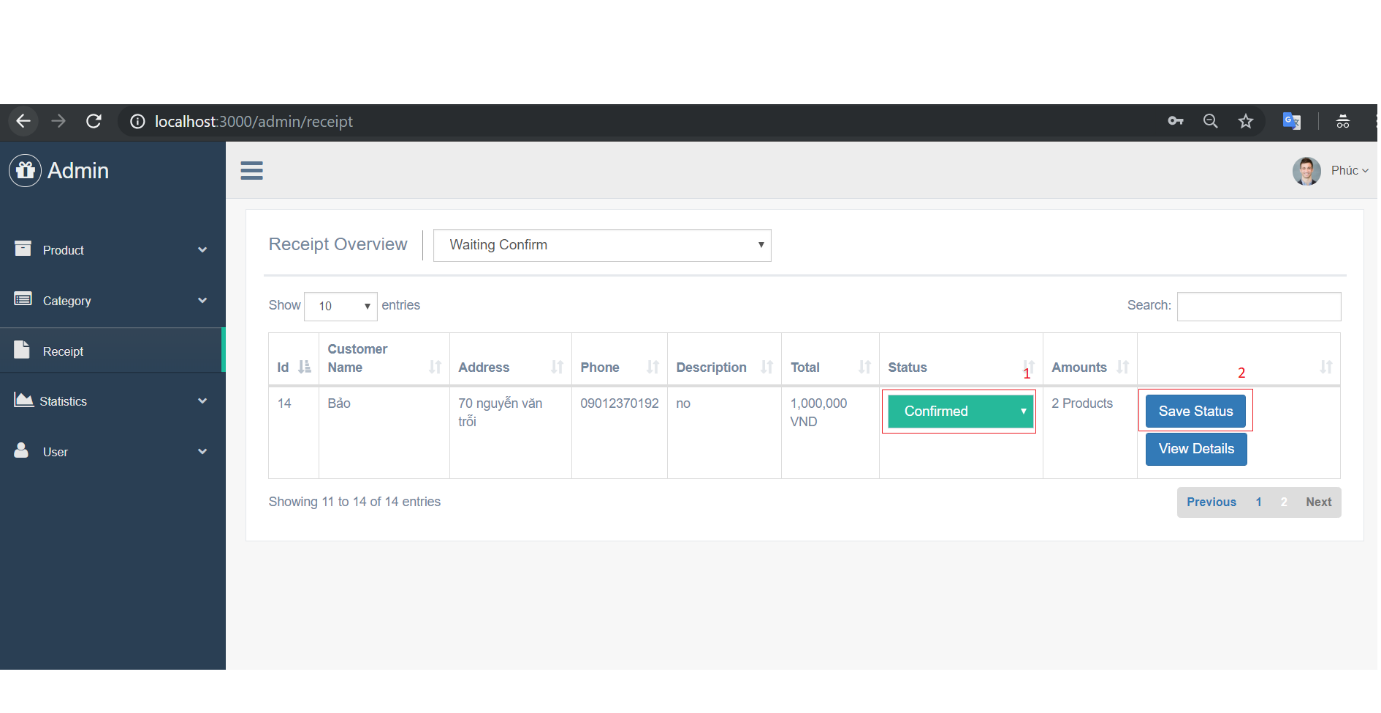


#### Quản lý đơn hàng:

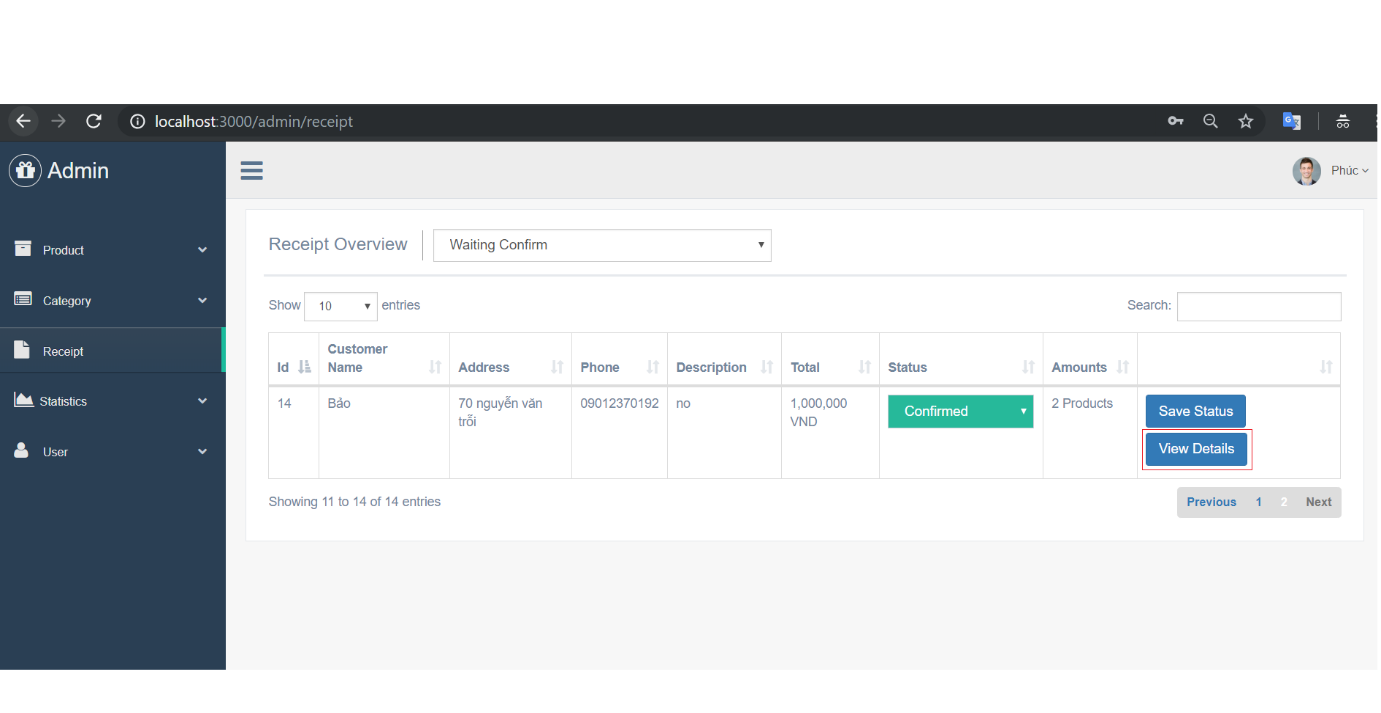
Để xem quản lý đơn hàng, người dùng vào đường link <http://localhost:3000/admin/receipt> .



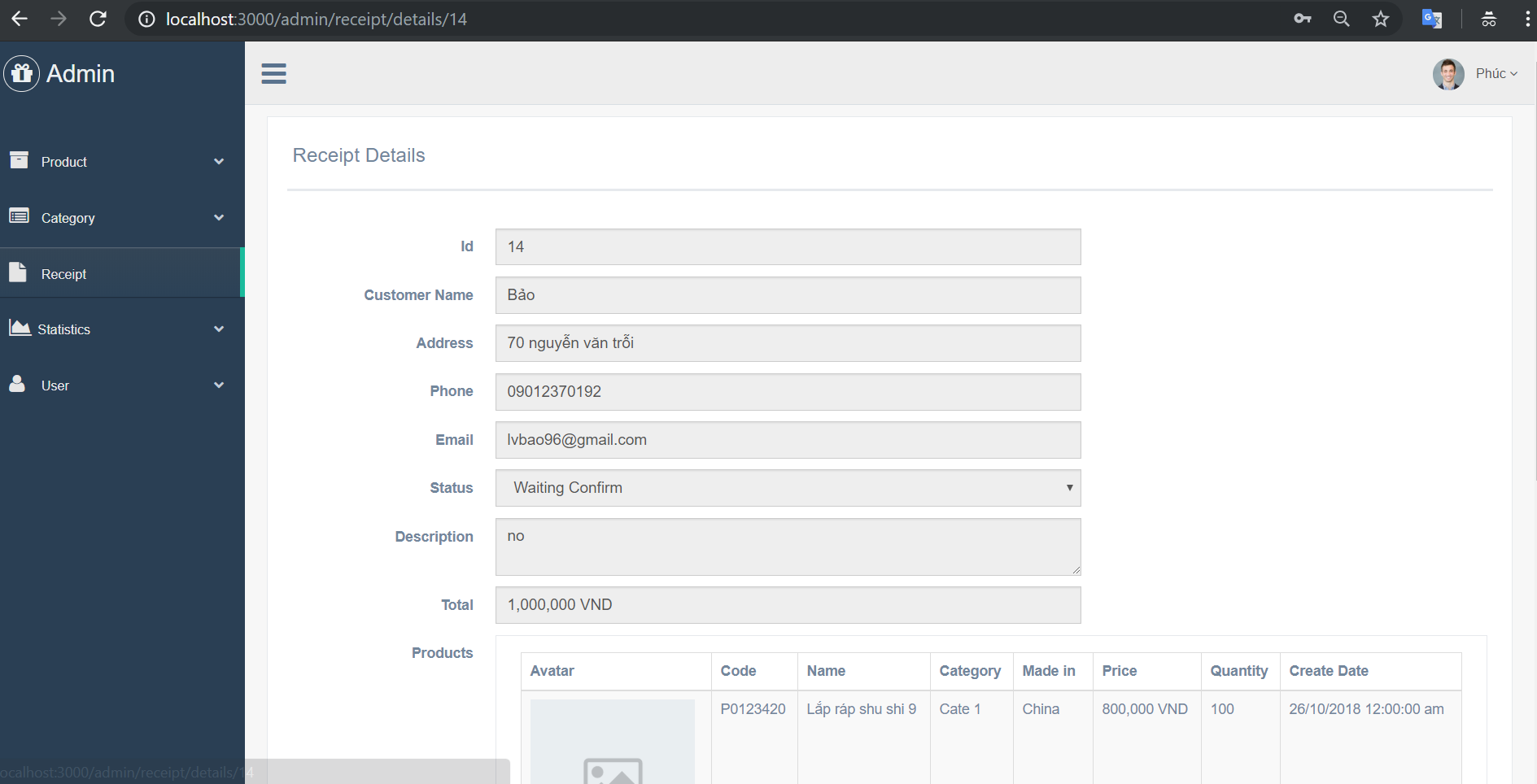
Sau đó, để thay đổi trạng thái đơn hàng người dùng chọn trạng thái ở cột “Status” và nhấn nút “Save Status” để thay đổi trạng thái.



Còn để xem đầy đủ các thông tin của đơn hàng, người nhấn vào nút “View Details”.

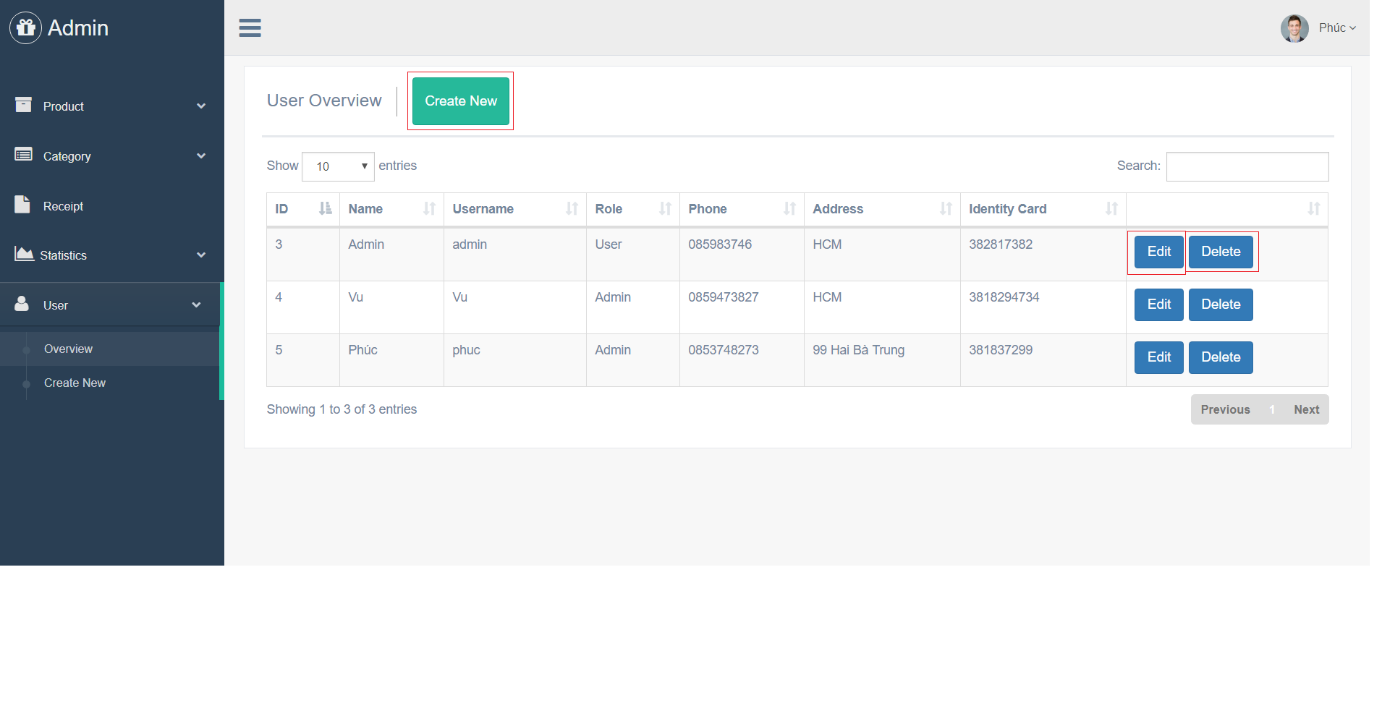


Màn hình thông tin chi tiết đơn hàng.

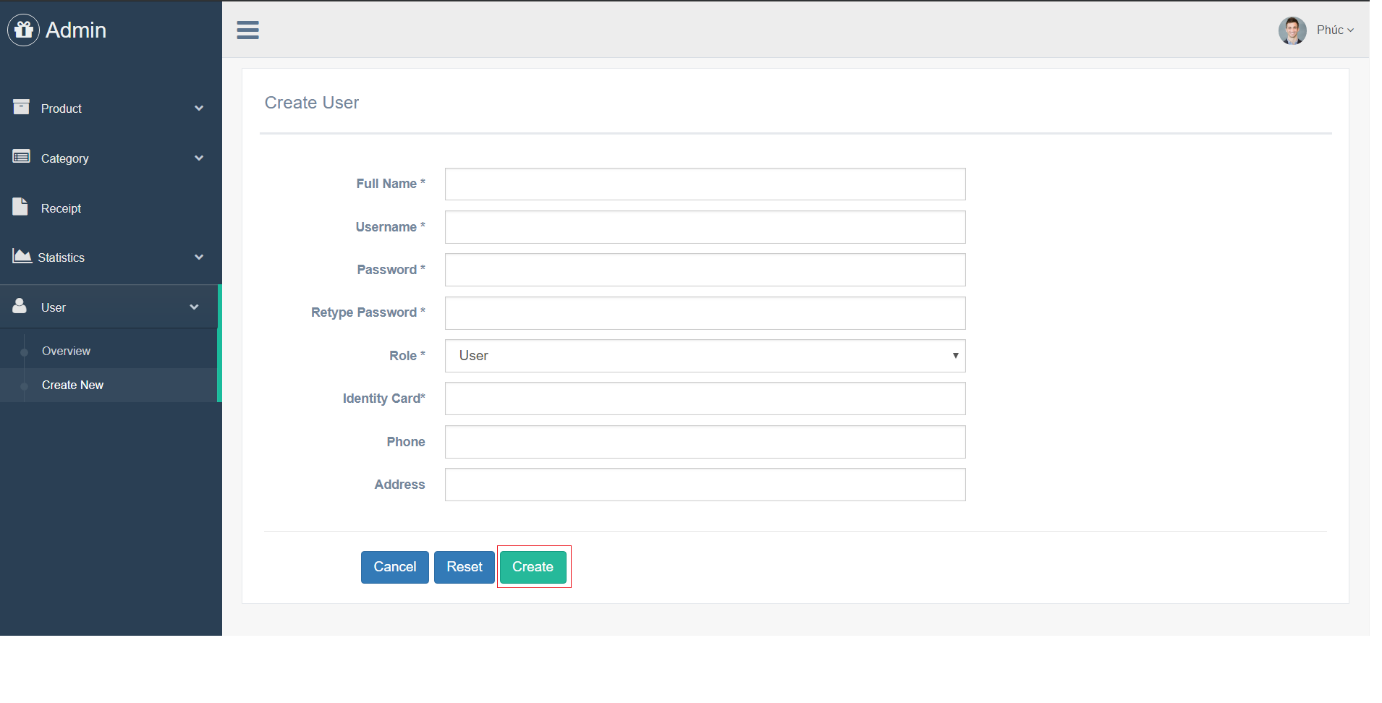


#### Quản lý người dùng:

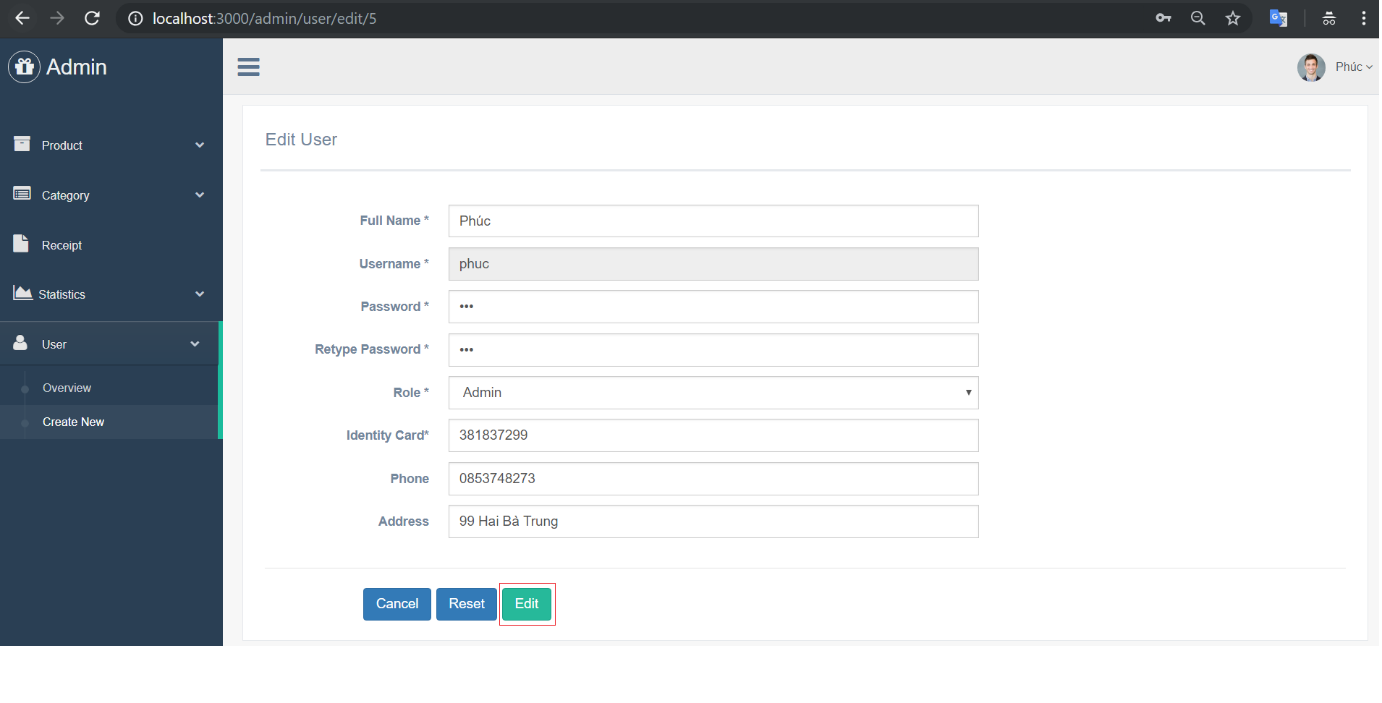
Để quản lý người dùng, người dùng phải đăng nhập bằng tài khoản có quyền admin và vào đường link <http://localhost:3000/admin/user> .



Để thêm người dùng, người dùng nhấn vào nút “Create New” , điền đầy đủ thông tin và nhấn nút “Create”.



Để sửa người dùng, người dùng chọn người muốn sửa nhấn nút “Edit”, sau đó điền đầy đủ thông tin cần sửa và nhấn nút “Edit” để lưu lại.



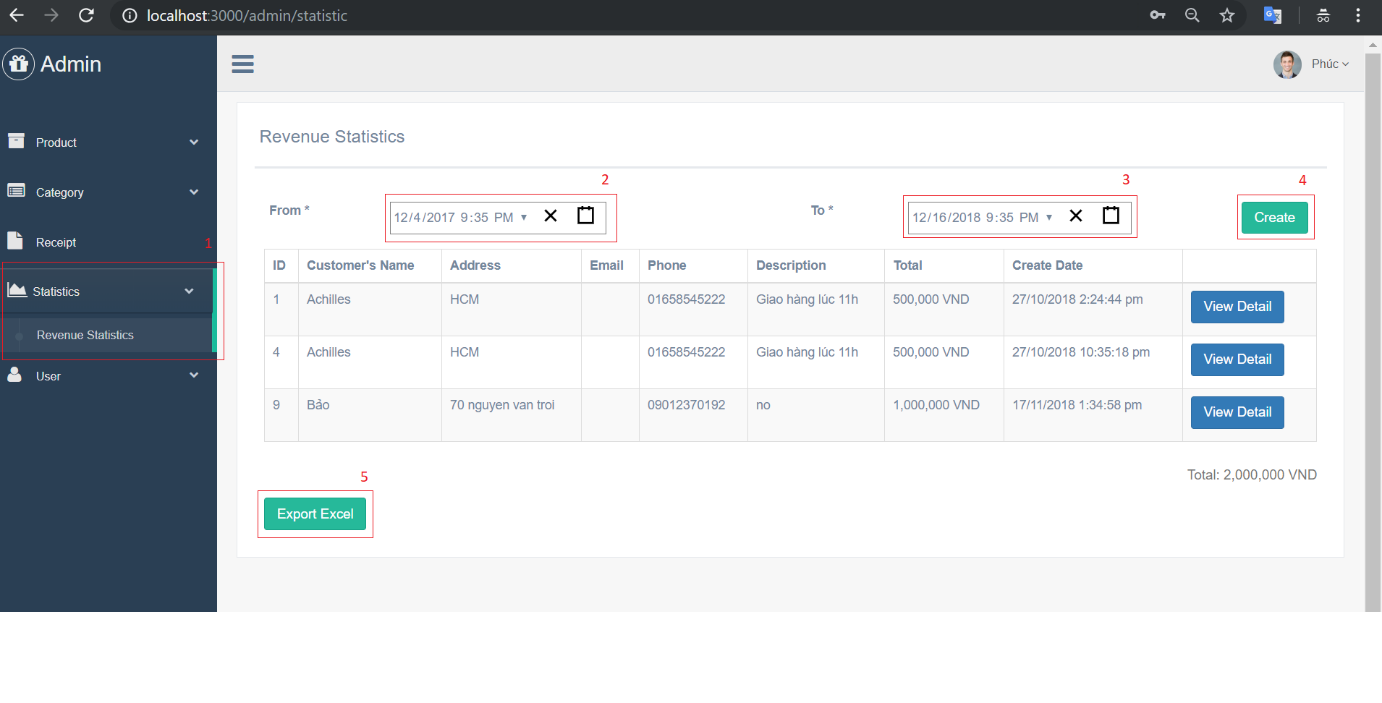
Để xoá người dùng, người dùng chọn người muốn xoá và nhấn nút “Delete” để hệ thống tiến hành xoá.

#### Thống kê doanh thu theo thời gian:

Để thống kê doanh thu, người dùng truy cập vào link <http://localhost:3000/admin/statistic> hoặc chọn vào chức năng “Statistics” -> “Revenue Statistics” trên thanh điều hướng.

Lựa chọn thời gian thống kê và nhấn nút “Create” để hệ thống tiến hành lấy thông tin.

Ngoài ra, người dùng còn có thể nhấn nút “View Detail” để xem thông tin chi tiết của đơn hàng như phần Quản lý đơn hàng và còn có thể xuất những thông tin của đơn hàng ra file excel bằng cách nhấn nút “Export Excel”.



TÀI LIỆU THAM KHẢO