|  |  |
| --- | --- |
|  |  |
| TRẦN VĂN LỊCH | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| **ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC**  **NGÀNH CÔNG NGHỆ THÔNG TIN** |
|  |
| **XÂY DỰNG WEBSITE BÁN HÀNG THỜI TRANG BẰNG**  **REACTJS VÀ NODEJS** |
|  |
|  |
| **CBHD:Ths. Vũ Duy Giang** |
| CÔNG NGHỆ THÔNG TIN | **Sinh viên: Trần Văn Lịch** |
| **Mã số sinh viên: 2019606523** |
|  |
|  |
|  |
| Hà Nội – Năm 2023 |
|  |

**LỜI NÓI ĐẦU**

Việc mua bán hàng online đang là một trong những nhu cầu hết sức thiết yếu hiện nay, đặc biệt là nhu cầu về mua sắm thời trang, do đó hệ thống hỗ trợ mua những món đồ thời trang online cũng là một trong những vấn đề đang được quan tâm. Việc xây dựng website bán hàng thời trang góp phần giúp người dùng dễ dàng tìm hiểu và đặt mua những món đồ thời trang của hang một cách dễ dàng.

Sau khi tìm hiểu và dưới sự hướng dẫn của thầy Vũ Duy Giang, em đã phân tích và thiết kế được **Website bán hàng thời trang bằng NodeJs và ReactJs**. Để giúp khách hàng đặt mua được những món đồ thời trang theo ý thích của mình, trước hết cần có một hệ thống tốt, một website có các dữ liệu chính xác, các đánh giá tốt của người dùng, …. Đó cũng là lý do mà em chọn đề tài này.

Để hoàn thành được đồ án tốt nghiệp này, em xin được gửi lời cảm ơn chân thành đến:

* Em xin cảm ơn các thầy/ cô trong Khoa Công nghệ Thông tin Trường Đại học Công Nghiệp Hà Nội đã tận tình giảng dạy và trang bị kiến thức cho em trong suốt thời gian em học tập tại trường.
* Thầy giáo hướng dẫn đề tài - **Thạc Sĩ Vũ Duy Giang**, Giảng viên Khoa Công nghệ Thông tin Trường Đại học Công Nghiệp Hà Nội – đã tận tuỵ hết lòng giúp đỡ, hướng dẫn, chỉ dẫn tận tình để giúp em hoàn thành được đồ án.
* Bạn bè đã góp ý và giúp đỡ, bên cạnh động viên và khích lệ tôi trong suốt thời gian tôi làm đồ án.

***Hà Nội, Ngày 09 tháng 05 năm 2023***

**MỤC LỤC**

[TÓM TẮT ĐỒ ÁN TỐT NGHIỆP 8](#_Toc134531676)

[CHƯƠNG 1: LỜI MỞ ĐẦU 9](#_Toc134531677)

[1.1. Tên đề tài 9](#_Toc134531678)

[1.2. Lý do chọn đề tài: 9](#_Toc134531679)

[1.3. Mục tiêu của đề tài 10](#_Toc134531680)

[1.4. Đối tượng và phạm vi 11](#_Toc134531681)

[1.5. Kết quả dự kiến đạt được 11](#_Toc134531682)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 12](#_Toc134531683)

[2.1. Cơ sở lý thuyết 12](#_Toc134531684)

[*2.1.1.* *ReactJs* 12](#_Toc134531685)

[*2.1.2.* *NodeJs* 13](#_Toc134531686)

[*2.1.3.* *Cơ sở dữ liệu MongoDB* 16](#_Toc134531687)

[2.2. Công cụ sử dụng 16](#_Toc134531688)

[CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG 17](#_Toc134531689)

[3.1. Phân tích thiết kế 17](#_Toc134531690)

[*3.1.1.* *Vai trò của người dùng* 17](#_Toc134531691)

[*3.1.2.* *Yêu cầu chức năng* 17](#_Toc134531692)

[*3.1.3.* *Yêu cầu phi chức năng* 18](#_Toc134531693)

[3.2. Biểu đồ usecase tổng quát 19](#_Toc134531694)

[*3.2.1.* *Biểu đồ usecase admin và khách hàng* 19](#_Toc134531695)

[3.3. Đặc tả usecase 20](#_Toc134531696)

[*3.3.1.* *Use case đăng nhập* 20](#_Toc134531697)

[*3.3.2.* *Use case đăng ký* 24](#_Toc134531698)

[*3.3.3.* *Use case xem chi tiết sản phẩm:* 27](#_Toc134531699)

[*3.3.4.* *Use case quản lý sản phẩm:* 30](#_Toc134531700)

[*3.3.5.* *Use case quản lý tài khoản:* 37](#_Toc134531701)

[*3.3.6.* *Use case đặt hàng.* 43](#_Toc134531702)

[CHƯƠNG 4: THIẾT KẾ HỆ THỐNG 47](#_Toc134531703)

[4.1. Thiết kế cơ sở dữ liệu 47](#_Toc134531704)

[*4.1.1.* *Biểu đồ mô tả cơ sở dữ liệu* 47](#_Toc134531705)

[4.2. Thiết kế giao diện 47](#_Toc134531706)

[KẾT LUẬN 57](#_Toc134531707)

[TÀI LIỆU THAM KHẢO 59](#_Toc134531708)

**CÁC HÌNH VẼ SỬ DỤNG TRONG ĐỀ TÀI**

[*Hình 2. 1: NodeJs logo.* 13](#_Toc134531709)

[Đặc điểm của công nghệ : 13](#_Toc134531710)

[*Hình 3. 1: Biểu đồ usecase admin và khách hàng.* 19](#_Toc134531711)

[*Hình 3. 2: Biểu đồ usecase đăng nhập.* 19](#_Toc134531712)

[*Hình 3. 3: Biểu đồ trình tự đăng nhập.* 21](#_Toc134531713)

[*Hình 3. 4: Biểu đồ hoạt động đăng nhập.* 22](#_Toc134531714)

[*Hình 3. 5: Biểu đồ các lớp tham gia ca sử dụng đăng nhập.* 23](#_Toc134531715)

[*Hình 3. 6: Biểu đồ use case đăng ký.* 23](#_Toc134531716)

[*Hình 3. 7: Biểu đồ trình tự đăng ký.* 25](#_Toc134531717)

[*Hình 3. 8: Biểu đồ hoạt động đăng ký.* 25](#_Toc134531718)

[*Hình 3. 9: Biểu đồ use case xem chi tiết sản phẩm.* 26](#_Toc134531719)

[*Hình 3. 10: Biểu đồ trình tự xem chi tiết sản phẩm.* 27](#_Toc134531720)

[*Hình 3. 11: Biểu đồ hoạt động xem chi tiết sản phẩm.* 28](#_Toc134531721)

[*Hình 3. 12: Biểu đồ các lớp tham gia ca sử dụng xem chi tiết sản phẩm.* 28](#_Toc134531722)

[*Hình 3. 13: Biểu đồ use case quản lý sản phẩm.* 29](#_Toc134531723)

[*Hình 3. 14: Biểu đồ tuần tự thêm sản phẩm.* 32](#_Toc134531724)

[*Hình 3. 15: Biểu đồ tuần tự sửa sản phẩm.* 33](#_Toc134531725)

[*Hình 3. 16: Biểu đồ tuần tự xóa sản phẩm.* 33](#_Toc134531726)

[*Hình 3. 17: Biểu đồ tuần tự tìm kiếm sản phẩm.* 34](#_Toc134531727)

[*Hình 3. 18: Biểu đồ hoạt động quản lý sản phẩm.* 35](#_Toc134531728)

[*Hình 3. 19: Biểu đồ các lớp tham gia vào ca sử dụng quản lý sản phẩm.* 35](#_Toc134531729)

[*Hình 3. 20: Biểu đồ use case quản lý tài khoản.* 36](#_Toc134531730)

[*Hình 3. 22: Biểu đồ tuần tự thêm tài khoản.* 38](#_Toc134531731)

[*Hình 3. 23: Biểu đồ tuần tự sửa tài khoản.* 39](#_Toc134531732)

[*Hình 3. 24: Biểu đồ tuần tự xóa tài khoản.* 39](#_Toc134531733)

[*Hình 3. 25: Biểu đồ hoạt động quản lý tài khoản.* 40](#_Toc134531734)

[*Hình 3. 26: Biểu đồ lớp tham gia vào ca sử dụng quản lý tài khoản.* 41](#_Toc134531735)

[*Hình 3. 27: Biểu đồ use case mua hàng.* 42](#_Toc134531736)

[*Hình 3. 28: Biểu đồ trình tự đặt hàng.* 44](#_Toc134531737)

[*Hình 3. 29: Biểu đồ hoạt động đặt hàng.* 44](#_Toc134531738)

[*Hình 3. 30: Biểu đồ các lớp tham gia ca sử dụng đặt hàng..* 45](#_Toc134531739)

[*Hình 4. 1: Biểu đồ mô tả cấu trúc database.* 46](#_Toc134531740)

[*Hình 4. 2: Giao diện trang đăng nhập.* 47](#_Toc134531741)

[*Hình 4. 3: Giao diện trang đăng ký.* 47](#_Toc134531742)

[*Hình 4. 4: Giao diện cửa hàng.* 49](#_Toc134531743)

[*Hình 4. 5: Giao diện chi tiết sản phẩm.* 49](#_Toc134531744)

[*Hình 4 .6: Giao diện đặt hàng* 50](#_Toc134531745)

[*Hình 4 .7: Giao diện sửa đổi thông tin nhận hàng.* 50](#_Toc134531746)

[*Hình 4. 8: Giao diện trang chủ admin.* 51](#_Toc134531747)

[*Hình 4. 9: Giao diện quản lý tài khoản/ quản trị viên* 52](#_Toc134531748)

[*Hình 4. 10: Giao diện trang quản lý danh mục sản phẩm.* 53](#_Toc134531749)

[*Hình 4. 11: Giao diện trang quản lý sản phẩm.* 53](#_Toc134531750)

[*Hình 4. 12: Giao diện thêm mới sản phẩm.* 54](#_Toc134531751)

[*Hình 4. 13: Giao diện thêm mới danh mục.* 54](#_Toc134531752)

[*Hình 4. 14: Giao diện quản lý hoá đơn.* 55](#_Toc134531753)

TÓM TẮT ĐỒ ÁN TỐT NGHIỆP

Tên đề tài: “Xây dựng website bán hàng thời trang bằng NodeJs và ReactJs”.

Hiện nay, với sự phát triển không ngừng của khoa học công nghệ cũng như sự phát triển vượt bậc của ngành công nghệ thông tin. Nền kinh tế thế giới nói chung và nền kinh tế Việt Nam nói riêng đang dần chuyển từ nền kinh tế từ sản xuất công nghiệp nặng sang nền kinh tế sản xuất công nghiệp không khói. Trong đó công nghệ thông tin đóng vai trò vô cùng quan trọng trong cuộc sống sinh hoạt sản xuất con người. Mỗi người có thể sắm sản phẩm thông minh như điện thoại cảm ứng, máy tính... Nền kinh tế dần phát triển các doanh nghiệp công nghệ thông tin cũng đóng vai trò vô cùng quan trọng cho nền phát triển kinh thế của mỗi quốc gia.

Giờ đây, Internet đã trở thành dịch vụ phổ biến và thiết yếu và có ảnh hưởng sâu rộng tới thói quen, sinh hoạt, giải trí của nhiều người. Cùng với sự phát triển nhanh chóng của Internet thì các hình thức mua và bán hàng hóa cho mọi người ngày càng đa dạng và phát triển hơn. Các ứng dụng, các website ngày càng trở nên phổ biến. Trước nhu cầu đó, cùng với yêu cầu đề tài, em em xin được nhận đề tài “Xây dựng website bán hàng thời trang bằng NodeJs và ReactJs” làm đồ án tốt nghiệp của mình. Em đã tiến hành tìm hiểu, lên ý tưởng, phân tích, thiết kế xây dựng website dựa trên nền tảng công nghệ NodeJs và ReactJs, hỗ trợ bán hàng trực tuyến qua website. Đồ án này bao gồm 4 chương:

Chương 1: Tổng quan.

Chương 2: Cơ sở lý thuyết và công cụ sử dụng.

Chương 3: Phân tích thiết kế hệ thống mức logic.

Chương 4: Thiết kế hệ thống.

CHƯƠNG 1: LỜI MỞ ĐẦU

## Tên đề tài

Xây dựng website bán hàng thời trang bằng NodeJs và ReactJs.

## Lý do chọn đề tài:

Bán hàng trực tuyến là một hình thức kinh doanh các sản phẩm hay dịch vụ trên mạng internet. Tận dụng ưu thế của mạng internet để tạo ra quá trình mua bán trao đổi hàng hóa. Ưu điểm của bán hàng online đó chính là không cần phải phụ thuộc vào cửa hàng, mặt bằng, nhân viên. Chỉ cần bạn xây dựng một kênh thông tin bán hàng thông qua mạng internet để tiếp cận khách hàng. Chính vì thế, thiết kế một website bán hàng trực tuyến là yêu cầu cần thiết cho hình thức kinh doanh này. Website được coi như là một cửa hàng, giúp cho bạn có thể trưng bày các sản phẩm kèm thông tin giới thiệu. Ngoài ra, website còn tích hợp các chức năng giúp cho việc mua bán diễn ra một cách nhanh chóng và tiện lợi nhất.

Việc bán hàng thủ công thông qua các cửa hàng mà người mua phải đến tận nơi không còn hiệu quả, việc quảng cáo cũng như quảng bá những sản phẩm không đến được với nhiều người sử dụng. Việc sử dụng Internet đã đưa ra một bước ngoặt mới cho định hướng phát triển ngành Công nghệ thông tin của nước ta, cùng với nhu cầu sử dụng máy tính để trao đổi và cập nhật thông tin ngày càng tăng. Từ đó việc đưa ra thông tin nhanh chóng tiết kiệm là vô cùng cấp thiết. Bán hàng qua mạng là một giải pháp tối ưu trong việc phân phối thông tin vào mục đích thương mại. Việc bán hàng qua mạng giúp giải quyết được những vấn đề tồn tại của bán hàng thủ công nó không những giúp cho sản phẩm được biết đến nhiều hơn người mua sản phẩm không phải đến tận nơi mua mà thông qua website có thể đặt mua sản phẩm và sản phẩm có thể chuyển về đến tận nơi yêu cầu, tiết kiệm được thời gian.

Ngày nay, bạn có thể mua cho mình tất cả các mặt hàng thời trang một cách đơn giản, dễ dàng. Do đó xây dựng website bán hàng là một vấn đề thực tế, ứng dụng được và có tiềm năng phát triển trong tương lai và nhận thấy được nhu cầu “Thuận mua vừa bán". Với lý do đó, được sự hướng dẫn tận tình của thầy giáo **ThS. Vũ Duy Giang**, em đã chọn đề tài **“Xây dựng Website bán hàng thời trang bằng ReactJs và NodeJs.”** sử dụng ReactJs, NodeJs và MongoDB làm nên website bán hàng này.

## Mục tiêu của đề tài

Đề tài: Xây dựng Website bán hàng thời trang đáp ứng được những mục tiêu:

+ Hiểu biết những kiến thức cần thiết về xây dựng giao diện trang website phân tích thiết kế hệ thống, cơ sở dữ liệu.

+Nắm được các kiến thức về NodeJs và ReactJs.

+ Lợi ích đầu tiên khi bạn sở hữu một website bán hàng đó chính là việc tiếp cận được những khách hàng tiềm năng. Dù khách hàng có ở bất cứ đâu trên thế giới, chỉ cần có mạng internet là có thể tìm thấy website bán hàng của bạn, biết đến sản phẩm và thương hiệu của công ty bạn. Từ đó, chắc chắn rằng lượng khách hàng của bạn sẽ tăng lên đáng kể.

+ Tiết kiệm được thời gian cho khách hàng, khách hàng không cần phải đến tận cửa hàng mới mua được hàng mà họ có thể thông qua website xem và mua hàng. Hơn nữa mặt hàng khách mua còn có thể được chuyển về địa chỉ yêu cầu đúng thời gian khách hàng mong muốn, tiết kiệm được cả chi phí đi lại cho khách hàng.

+ Thông tin sản phẩm được cập nhật nhanh chính xác qua từng ngày đến khách hàng**.**

**+** Xây dựng và quảng bá thương hiệu cho hãng.

## Đối tượng và phạm vi

Bất cứ ai có nhu cầu mua hàng thời trang và nhà quản lý trên nền tảng Internet.

## Kết quả dự kiến đạt được

- Website bao gồm những mục chính và chức năng chính:

+ Có thể đăng ký, đăng nhập tài khoản cá nhân để quản lý trang website.

+ Thay đổi mật khẩu, thông tin cá nhân của tài khoản.

+ Phân quyền giữa người quản trị và khách hàng.

+ Hình ảnh đẹp, background nhẹ nhàng dễ nhìn.

+ Thêm, sửa, xóa, tìm kiếm sản phẩm.

+ Có chức năng giỏ hàng và lưu trữ lại các đơn hàng.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## Cơ sở lý thuyết

### *ReactJs*

**[React](https://reactjs.org/" \t "_blank)** là thư viện [**JavaScript**](https://www.hostinger.vn/huong-dan/javascript-la-gi/) phổ biến nhất để xây dựng giao diện người dùng (UI). Nó cho tốc độ phản hồi tuyệt vời khi user nhập liệu bằng cách sử dụng phương pháp mới để render trang web.

* Được phát triển bởi [**Facebook**](http://facebook.com/). Nó được ra mắt như một công cụ JavaScript mã nguồn mở vào năm 2013.
* Mục tiêu của react chính là đơn giản để phát triển. Tất cả trạng thái đều được tập trung tại một thời điểm, bằng cách chia giao diện người dùng thành tập hợp các thành phần (components)
* React được sử dụng để xây dựng single-page web applications.

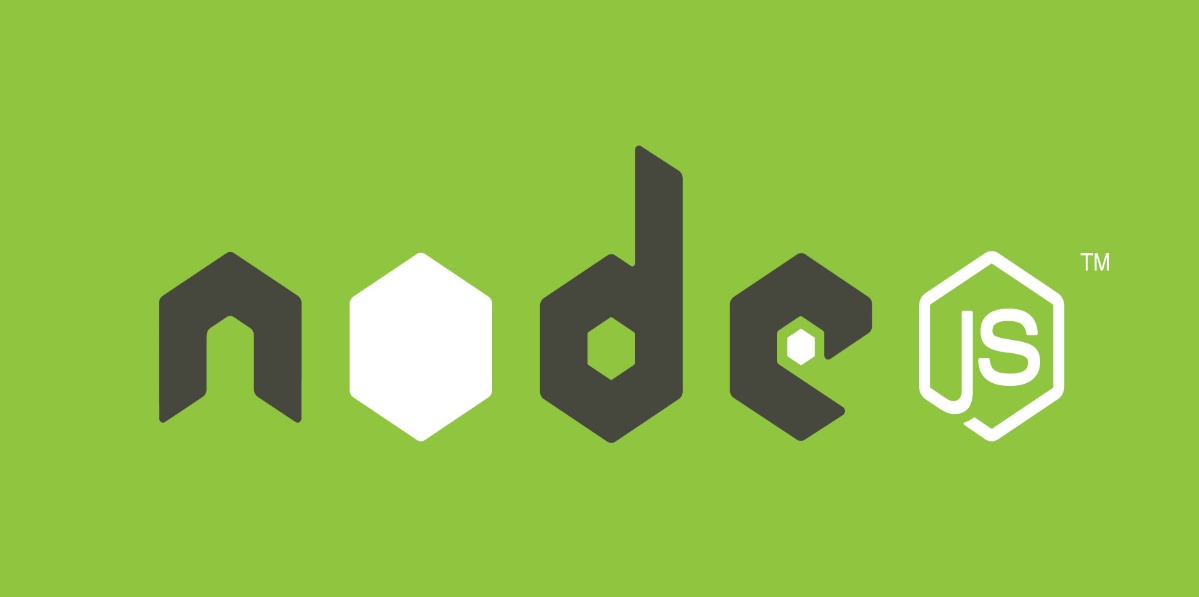
Đặc điểm của công nghệ:

* **JSX**
* Trong React, thay vì thường xuyên sử dụng JavaScript để thiết kế bố cục trang web thì sẽ dùng JSX. JSX được đánh giá là sử dụng đơn giản hơn JavaScript và cho phép trích dẫn HTML cũng như việc sử dụng các cú pháp thẻ HTML để render các subcomponent. JSX tối ưu hóa code khi biên soạn, vì vậy nó chạy nhanh hơn so với code JavaScript tương đương.
* **Single-way data flow (Luồng dữ liệu một chiều)**
* ReactJS không có những module chuyên dụng để xử lý data, vì vậy ReactJS chia nhỏ view thành các component nhỏ có mỗi quan hệ chặt chẽ với nhau. Tại sao chúng ta phải quan tâm tới cấu trúc và mối quan hệ giữa các component trong ReactJS? Câu trả lời chính là luồng truyền dữ liệu trong ReactJS: Luồng dữ liệu một chiều từ cha xuống con. Việc ReactJS sử dụng one-way data flow có thể gây ra một chút khó khăn cho những người muốn tìm hiểu và ứng dụng vào trong các dự án. Tuy nhiên, cơ chế này sẽ phát huy được ưu điểm của mình khi cấu trúc cũng như chức năng của view trở nên phức tạp thì ReactJS sẽ phát huy được vai trò của mình.
* **Virtual DOM**
* Những Framework sử dụng Virtual-DOM như ReactJS khi Virtual-DOM thay đổi, chúng ta không cần thao tác trực tiếp với DOM trên View mà vẫn phản ánh được sự thay đổi đó. Do Virtual-DOM vừa đóng vai trò là Model, vừa đóng vai trò là View nên mọi sự thay đổi trên Model đã kéo theo sự thay đổi trên View và ngược lại. Có nghĩa là mặc dù chúng ta không tác động trực tiếp vào các phần tử DOM ở View nhưng vẫn thực hiện được cơ chế Data-binding. Điều này làm cho tốc độ ứng dụng tăng lên đáng kể – môt lợi thế không thể tuyệt vời hơn khi sử dụng Virtula-DOM.

### *NodeJs*

**Lịch sử phát triển :**

* Nodejs là một nền tảng (Platform) phát triển độc lập được xây dựng ở trên Javascript Runtime của Chrome mà chúng ta có thể xây dựng được các ứng dụng mạng một cách nhanh chóng và dễ dàng mở rộng.
* Nodejs được xây dựng và phát triển từ năm 2009, bảo trợ bởi công ty Joyent, trụ sở tại California, Hoa Kỳ. Dù sao thì chúng ta cũng nên biết qua một chút chút lịch sử của thứ mà chúng ta đang học một chút chứ nhỉ?
* Phần Core bên dưới của Nodejs được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao.
* Nodejs tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực.
* Nodejs áp dụng cho các sản phẩm có lượng truy cập lớn, cần mở rộng nhanh, cần đổi mới công nghệ, hoặc tạo ra các dự án Startup nhanh nhất có thể.



*Hình 2. 1: NodeJs logo.*

Đặc điểm của công nghệ :

Nodejs sở hữu khá nhiều đặc điểm cơ bản, nổi trội và ưu việt. Hiểu thêm về các đặc điểm này sẽ giúp bạn có hiểu biết vững vàng hơn về Nodejs, đồng thời đưa ra quyết định sử dụng nền tảng này chính xác.

* **Không đồng bộ:** Đặc điểm đầu tiên của Nodejs là tính bất đồng bộ. Node.js không cần đợi [API](https://bizfly.vn/techblog/api-la-gi.html) trả dữ liệu về, vậy nên mọi APIs nằm trong thư viện Node.js đều không được đồng bộ, hiểu đơn giản là chúng không hề blocking (khóa). Server có cơ chế riêng để gửi thông báo và nhận phản hồi về các hoạt động của Node.js và API đã gọi.
* **Tốc độ nhanh:** Với phần core phía dưới lập trình gần như toàn bộ bằng ngôn ngữ C++, kết hợp với V8 Javascript Engine mà Google Chrome cung cấp, tốc độ vận hành, thực hiện code của thư viện Node.js diễn ra rất nhanh.
* **Đơn giản – Hiệu quả:** Tiến trình vận hành của Node.js đơn giản song lại mang đến hiệu năng cao nhờ ứng dụng mô hình single thread và các sự kiện lặp. Một loạt cơ chế sự kiện cho phép server trả về phản hồi bằng cách không block, đồng thời tăng hiệu quả sử dụng. Các luồng đơn cung cấp dịch vụ cho nhiều request hơn hẳn Server truyền thống.
* **Không đệm**: Nền tảng Node.js không có vùng đệm, tức không cung cấp khả năng lưu trữ dữ liệu buffer.
* Có giấy phép: Đây là nền tảng đã được cấp giấy phép, phát hành dựa trên MIT License.

Là một nền tảng phát triển ứng dụng mạnh mẽ, [Nodejs](https://vi.wikipedia.org/wiki/Node.js) có thể thỏa mãn mọi nhu cầu lập trình, phát triển ứng dụng. Các chuyên gia trong lĩnh vực lập trình, công nghệ khuyên dùng Nodejs khi phát triển các ứng dụng như Websocket server, Fast File Upload Client, Ad Server, Cloud Services, RESTful API, Any Real-time Data Application,…

Ngoài ra, Nodejs còn có thể:

* Hỗ trợ thực hiện các thao tác với tập tin như tạo, mở, đóng, đọc, ghi, xóa,… ngay trong lúc truy cập vào máy chủ.
* Tạo nội dung cho website động.
* Thu thập dữ liệu theo yêu cầu.
* Truy vấn, thêm, xóa, chỉnh sửa dữ liệu có trong hệ cơ sở MongoDB, Microsoft SQL Server, PostgreSQL và MySQL.

Với độ phổ biến của Nodejs trong thiết kế web hiện nay thì tìm hiểu Nodejs là gì trước khi tạo lập website là điều thiết yếu.

### *Cơ sở dữ liệu MongoDB*

* **MongoDB** là một database hướng tài liệu (document), một dạng NoSQL database. Vì thế, MongoDB sẽ tránh cấu trúc table-based của relational database để thích ứng với các tài liệu như JSON có một schema rất linh hoạt gọi là BSON. [**MongoDB**](https://topdev.vn/viec-lam-it/mongodb-kt90) sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ các các kích cỡ và các document khác nhau. Các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh.
* **Ưu điểm của MongoDB**
* Dữ liệu lưu trữ phi cấu trúc, không có tính ràng buộc, toàn vẹn nên tính sẵn sàng cao, hiệu suất lớn và dễ dàng mở rộng lưu trữ.
* Dữ liệu được caching (ghi đệm) lên RAM, hạn chế truy cập vào ổ cứng nên tốc độ đọc và ghi cao.

## Công cụ sử dụng

- Visual Studio Code: Phục vụ mục đích code ReactJs và NodeJs.

- Postman: Test api.

- MongoDB: lưu ảnh và dữ liệu.

CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG

## Phân tích thiết kế

### *Vai trò của người dùng*

- Hệ thống chia làm 2 nhóm người dùng chính:

+ Khách hàng.

+ Người quản trị.

- Mô tả nhóm người dùng.

+ Khách hàng (thành viên của trang web): Những khách hàng đã có tài khoản đăng nhập trên trang web. Khách hàng này khi mua hàng trên trang web thì thông tin của khách hàng đã được lưu lại trên trang web và khi người dùng muốn mua hàng chỉ cần đăng nhập bằng tài khoản thông tin của khách hàng sẽ hiển thị và khách hàng chỉ cần tiến hành mua.

+ Người quản trị: Là người quản lý trang web và được truy cập vào tất cả chức năng của hệ thống.

### *Yêu cầu chức năng*

- Thành viên của trang web:

+ Đăng nhập, đăng xuất.

+ Xem thông tin sản phẩm.

+ Đặt mua hàng.

+ Chat với admin.

- Quản trị viên:

+ Quản lý khách hàng.

+ Quản lý danh mục sản phẩm.

+ Quản lý sản phẩm.

+ Quản lý hóa đơn.

+ Quản lý tài khoản.

+ Giao tiếp tư vấn khách hàng.

### *Yêu cầu phi chức năng*

+ Giao diện trang web được thiết kế dễ sử dụng và có tính thẩm mỹ cao.

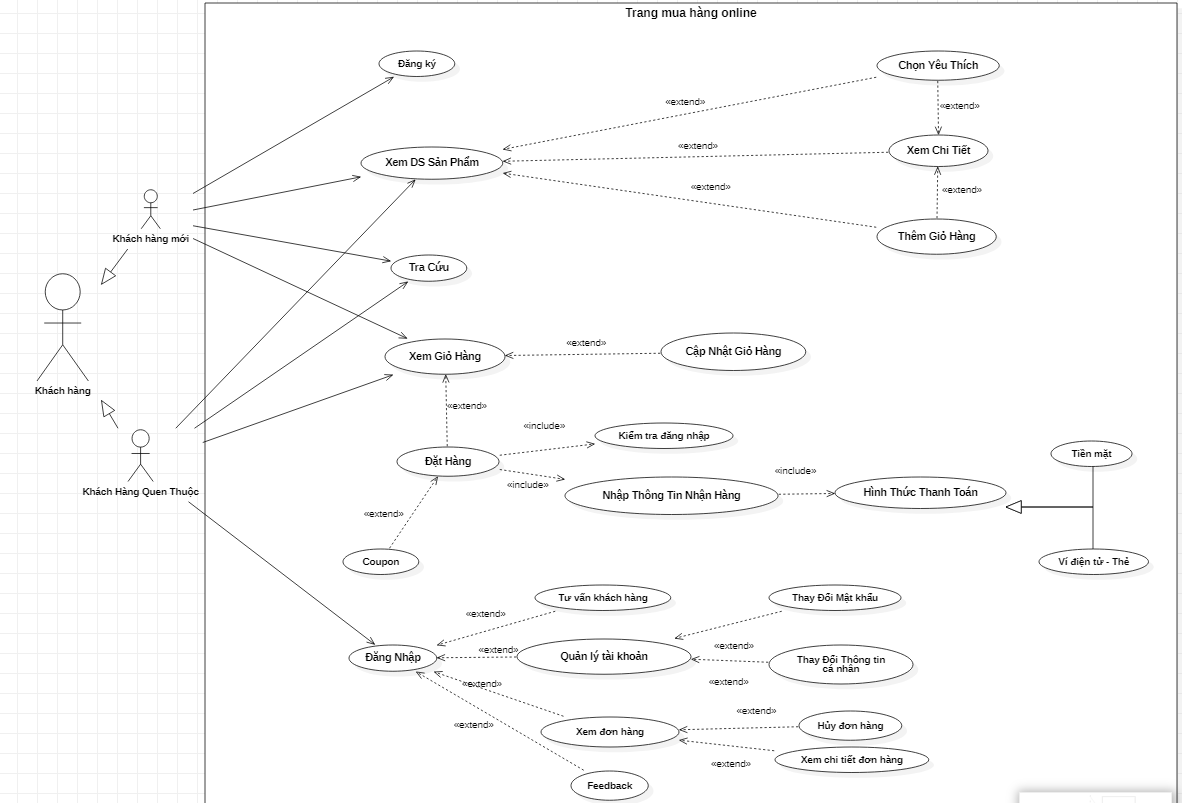
+ Được viết trên nền tảng web.

+ Hiệu năng: tốc độ, khả năng và độ tin cậy của hệ thống.

+ Website bảo mật phân quyền để thực hiện các chức năng của hệ thống.

## Biểu đồ usecase tổng quát

### *Biểu đồ usecase admin và khách hàng*



*Hình 3. 1: Biểu đồ usecase admin và khách hàng.*

## Đặc tả usecase

### *Use case đăng nhập*

1. Biểu đồ use case:

Diagram, venn diagram

Description automatically generated

*Hình 3. 2: Biểu đồ usecase đăng nhập.*

1. Đặc tả use case:

Use case này cho phép người dùng đăng nhập để xác định quyền truy cập vào hệ thống.

* Luồng sự kiện:
  + Luồng cơ bản:

1. Use case này bắt đầu khi người dùng bấm vào nút “Đăng nhập” trên góc phải màn hình.
2. Hệ thống yêu cầu người dùng nhập các thông tin bao gồm: Tên đăng nhập và mật khẩu.
3. Người dùng nhập các thông tin: Tên đăng nhập và mật khẩu và bấm nút “Đăng nhập”.
4. Hệ thống kiểm tra tên đăng nhập và mật khẩu đã nhập và hiển thị menu chính.
   * Luồng rẽ nhánh:
5. Sai tên đăng nhập hoặc mật khẩu:

Tại bước 3 trong luồng cơ bản nếu người dùng nhập tên đăng nhập hoặc mật khẩu sai, hệ thống hiển thị thông báo lỗi. Người dùng có thể nhập lại hoặc bỏ qua thao tác, khi đó use case kết thúc.

1. Đăng ký: Tại bước 3 trong luồng cơ bản, nếu người dùng bấm nút “Đăng ký” use case kết thúc.
2. Quên mật khẩu: Tại bước 3 trong luồng cơ bản, nếu người dùng bấm nút “Quên mật khẩu” use case kết thúc.
3. Tại bất kì thời điểm nào trong quá trình thực hiện use nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Nếu use case thành công, người dùng sẽ đăng nhập được vào hệ thống. Nếu không trạng thái của hệ thống không thay đổi.

* Điểm mở rộng:

Không có.

1. Biểu đồ trình tự:

Diagram

Description automatically generated

*Hình 3. 3: Biểu đồ trình tự đăng nhập.*

1. Biểu đồ hoạt động:

Diagram

Description automatically generated

*Hình 3. 4: Biểu đồ hoạt động đăng nhập.*

1. Biểu đồ các lớp tham gia ca sử dụng:

Diagram

Description automatically generated

*Hình 3. 5: Biểu đồ các lớp tham gia ca sử dụng đăng nhập.*

### *Use case đăng ký*

1. Biểu đồ use case:

Diagram

Description automatically generated

*Hình 3. 6: Biểu đồ use case đăng ký.*

1. Đặc tả use case:

Use case này cho phép khách truy vãng lai thực hiện đăng ký tài khoản để có quyền truy cập vào hệ thống.

* Luồng sự kiện:
  + Luồng cơ bản:

1. Use case này bắt đầu khi người dùng bấm vào nút “Đăng Ký”.
2. Hệ thống yêu cầu người dùng nhập các thông tin bao gồm: Họ và tên, email, số điện thoại, tên đăng nhập và mật khẩu.
3. Sau khi người người nhập các thông tin trên và bấm nút “Đăng ký”.
4. Hệ thống sẽ gửi thông tin người dùng vừa nhập lên hệ thống và lưu thông tin vào cơ sở dữ liệu. Use case kết thúc.
   * Luồng rẽ nhánh:
5. Đăng nhập: Tại bước 3 trong luồng cơ bản, nếu người dùng bấm nút “Đăng nhập” use case kết thúc.
6. Tại bất kì thời điểm nào nếu hệ thống không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Nếu use case thành công, người dùng có tài khoản để đăng nhập vào hệ thống.

* Điểm mở rộng:

Không có.

1. Biểu đồ trình tự:

Diagram

Description automatically generated

*Hình 3. 7: Biểu đồ trình tự đăng ký.*

1. Biểu đồ hoạt động:

Diagram

Description automatically generated

*Hình 3. 8: Biểu đồ hoạt động đăng ký**.*

### *Use case xem chi tiết sản phẩm:*

1. Biểu đồ use case:

Diagram

Description automatically generated

*Hình 3. 9: Biểu đồ use case xem chi tiết sản phẩm.*

1. Đặc tả use case:

Use case này cho phép khách hàng xem chi tiết về sản phẩm.

* Luồng sự kiện:
  + Luồng cơ bản:

1. Use case này bắt đầu khi người dùng kích vào tên sản phẩm trong danh sách sản phẩm. Hệ thống lấy các thông tin sản phẩm trong cơ sở dữ liệu và hiển thị lên màn hình.
2. Use case kết thúc.
   * Luồng rẽ nhánh:
3. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu hệ thống không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Không có.

* Điểm mở rộng:

Không có.

1. Biểu đồ trình tự:

Diagram

Description automatically generated

*Hình 3. 10: Biểu đồ trình tự xem chi tiết sản phẩm.*

1. Biểu đồ hoạt động:

Diagram

Description automatically generated

*Hình 3. 11: Biểu đồ hoạt động xem chi tiết sản phẩm.*

1. A picture containing text

   Description automatically generatedBiểu đồ các lớp tham gia vào ca sử dụng:

*Hình 3. 12: Biểu đồ các lớp tham gia ca sử dụng xem chi tiết sản phẩm.*

### *Use case quản lý sản phẩm:*

1. Biểu đồ use case:

Diagram

Description automatically generated

*Hình 3. 13: Biểu đồ use case quản lý sản phẩm.*

1. Đặc tả use case:

* Luồng sự kiện:
  + Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị kích vào nút “Quản lý sản phẩm” ở menu người quản trị. Hệ thống sẽ truy vấn bảng pro\_product trong cơ sở dữ liệu và hiển thị lên màn hình danh sách các sản phẩm lên màn hình.
2. **Thêm sản phẩm:**

a. Khi người quản trị bấm vào nút “Thêm mới”.

b. Hệ thống sẽ hiển thị popup bao gồm các thông tin: Tên sản phẩm, loại sản phẩm, giá chuẩn, số lượng, ảnh sản phẩm, mô tả sản phẩm cho phép người quản trị nhập.

c. Người quản trị sẽ nhập các thông tin đã nêu trên và bấm vào nút “Lưu lại”. Hệ thống sẽ kiểm tra dữ liệu nhập vào, tạo một mã sản phẩm cho sản phẩm và một bản ghi về sản phẩm sẽ được thêm vào bảng pro\_product, đồng thời thông tin sản phẩm này sẽ được hiển thị trong danh sách sản phẩm.

1. **Sửa sản phẩm:**
2. Khi người quản trị bấm vào update trên từng dòng thông tin về sản phẩm.
3. Hệ thống sẽ hiển thị ra một popup bao gồm các thông tin về sản phẩm các trường cho phép người quản trị sửa thông tin.
4. Người quản trị sửa đổi các thông tin về sản phẩm gồm các thông tin được chỉ ra trong mục thêm sản phẩm và kích vào nút “Lưu lại”.
5. Hệ thống sẽ kiểm tra thông tin thay đổi và cập nhật vào bảng pro\_product và cập nhật hiển thị của danh sách sản phẩm.
6. **Xoá sản phẩm:**

a. Khi người quản trị bấm vào button delete ở trên một dòng thông tin về sản phẩm.

b. Hệ thống hiển thị một thông báo hỏi có xác nhận xoá, nếu người quản trị chọn “ Có” thì hệ thống sẽ xoá bản ghi sản phẩm đó ra khỏi bảng pro\_product và cập nhật lại danh sách sản phẩm.

1. **Tìm kiếm sản phẩm:**
   1. Khi người quản trị nhập thông tin vào Form tìm kiếm sản phẩm và bấm nút “Tìm kiếm”. Hệ thống sẽ lấy ra các sản phẩm có thông tin khớp với thông tin người quản trị vừa nhập vào từ bản pro\_product và hiển thị trên danh sách sản phẩm.
2. Use case kết thúc khi người quản trị kích chọn thao tác khác trên menu người quản trị.
   * Luồng rẽ nhánh:
3. **Thêm sản phẩm**: Tại bước 2c trong luồng cơ bản nếu người quản trị bấm vào nút “Huỷ”, hệ thống sẽ không thực hiện tác vụ nào và use case kết thúc.
4. **Sửa sản phẩm**: Tại bước 3c trong luồng cơ bản nếu người quản trị bấm vào nút “Huỷ” và use case kết thúc.
5. **Xoá sản phẩm**: Tại bước 4b người quản trị chọn “Không” thì hệ thống không thực hiện xoá và use case kết thúc.
6. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu hệ thống không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Use case này cho chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.

* Tiền điều kiện:

Người dùng cần đăng nhập vào hệ thống với vai trò là quản trị viên trước khi thực hiện use case.

* Hậu điều kiện:

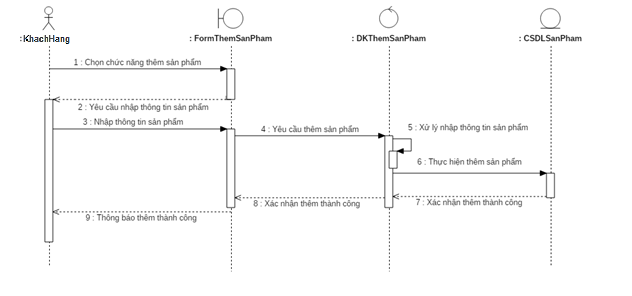
Nếu use case kết thúc thành công thì thông tin về sản phẩm sẽ được cập nhật trong cơ sở dữ liệu.

* Điểm mở rộng:

Không có.

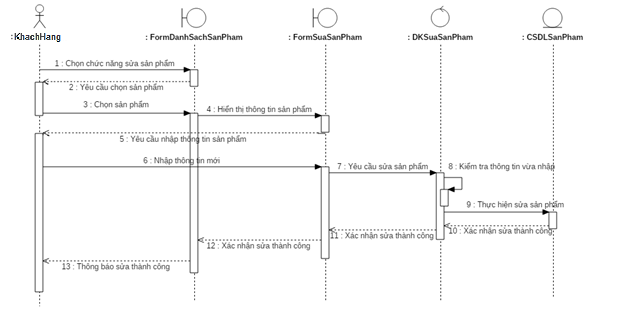
1. Biểu đồ trình tự:

* Thêm sản phẩm:



*Hình 3. 14: Biểu đồ tuần tự thêm sản phẩm.*

* Sửa sản phẩm:



*Hình 3. 15: Biểu đồ tuần tự sửa sản phẩm.*

* Xoá sản phẩm:

Diagram

Description automatically generated

*Hình 3.* *16: Biểu đồ tuần tự xóa sản phẩm.*

* Tìm kiếm sản phẩm:

Diagram

Description automatically generated

*Hình 3. 17: Biểu đồ tuần tự tìm kiếm sản phẩm.*

1. Biểu đồ hoạt động:

Diagram

Description automatically generated

*Hình 3. 18: Biểu đồ hoạt động quản lý sản phẩm.*

1. Biểu đồ các lớp tham gia ca sử dụng:

Diagram

Description automatically generated

*Hình 3. 19: Biểu đồ các lớp tham gia vào ca sử dụng quản lý sản phẩm.*

### *Use case quản lý tài khoản:*

* + 1. Biểu đồ use case:

Diagram

Description automatically generated

*Hình 3. 20: Biểu đồ use case quản lý tài khoản.*

* + 1. Đặc tả use case:
* Luồng sự kiện:
  + Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị kích vào nút “Quản lý sản phẩm” ở menu người quản trị. Hệ thống sẽ truy vấn bảng user trong cơ sở dữ liệu và hiển thị lên màn hình danh sách các sản phẩm lên màn hình.
2. **Thêm tài khoản quản trị viên:**

a. Khi người quản trị bấm vào nút “Thêm mới”.

b. Hệ thống sẽ hiển thị popup bao gồm các thông tin: Họ và tên, giới tính, ngày sinh, nơi sinh, email, số điện thoại, chứng minh thư, ngày cấp, nơi cấp, nguyên quán, địa chỉ thường trú, địa chỉ hiện tại, tài khoản, mật khẩu, ảnh đại cho phép người quản trị nhập.

c. Người quản trị sẽ nhập các thông tin đã nêu trên và bấm vào nút “Lưu lại”. Hệ thống sẽ kiểm tra dữ liệu nhập vào, tạo một mã người dùng cho tài khoản và một bản ghi về tài khoản sẽ được thêm vào bảng user, đồng thời thông tin tài khoản này sẽ được hiển thị trong danh sách tài khoản.

1. **Xoá tài khoản:**

a. Khi người quản trị bấm vào biểu tượng ở trên một dòng thông tin về tài khoản.

b. Hệ thống hiển thị một thông báo hỏi có xác nhận xoá, nếu người quản trị chọn “ Có” thì hệ thống sẽ xoá bản ghi tài khoản đó ra khỏi bảng user và cập nhật lại danh sách tài khoản.

* + Luồng rẽ nhánh:

1. **Thêm tài khoản người quản trị**: Tại bước 2c trong luồng cơ bản nếu người quản trị bấm vào nút “Huỷ”, hệ thống sẽ không thực hiện tác vụ nào và use case kết thúc.
2. **Sửa tài khoản**: Tại bước 3c trong luồng cơ bản nếu người quản trị bấm vào nút “Huỷ” và use case kết thúc.
3. **Xoá tài khoản**: Tại bước 4b người quản trị chọn “Không” thì hệ thống không thực hiện xoá và use case kết thúc.
4. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu hệ thống không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Use case này cho chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.

* Tiền điều kiện:

Người dùng cần đăng nhập vào hệ thống với vai trò là quản trị viên trước khi thực hiện use case.

* Hậu điều kiện:

Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.

* Điểm mở rộng:

Không có.

* + 1. Biểu đồ tuần tự quản lý tài khoản:
* Thêm tài khoản:

Diagram

Description automatically generated

*Hình 3. 22: Biểu đồ tuần tự thêm tài khoản.*

* Sửa thông tin tài khoản:

Diagram

Description automatically generated

*Hình 3. 23: Biểu đồ tuần tự sửa tài khoản.*

* Xóa tài khoản:

Diagram

Description automatically generated

*Hình 3. 24: Biểu đồ tuần tự xóa tài khoản.*

* + 1. Diagram

       Description automatically generatedBiểu đồ hoạt động quản lý tài khoản:

*Hình 3. 25: Biểu đồ hoạt động quản lý tài khoản.*

* + 1. Biểu đồ các lớp tham gia vào ca sử dụng:

Diagram

Description automatically generated

*Hình 3. 26: Biểu đồ lớp tham gia vào ca sử dụng quản lý tài khoản.*

.

### *Use case đặt hàng.*

1. Biểu đồ use case:

Diagram, venn diagram

Description automatically generated

*Hình 3. 27: Biểu đồ use case mua hàng.*

1. Đặc tả use case:

Use case này cho phép khách hàng đặt mua hàng.

* Luồng sự kiện:
  + Luồng cơ bản:

1. Use case này bắt đầu khi người dùng bấm vào nút “Đặt hàng”.
2. Hệ thống cho phép người dùng chỉnh sửa số lượng các sản phẩm và kích vào nút “Đặt hàng”.
3. Hệ thống hiển thị ra màn hình cho người dùng điền thông tin khách hàng “Hoàn tất”.
4. Use case kết thúc.
   * Luồng rẽ nhánh:
5. Tại bước 2 trong luồng cơ bản, nếu người dùng bấm nút “Tiếp tục mua hàng” use case kết thúc.
6. Tại bước 3 trong luồng cơ bản, nếu người dùng kích chọn “Quay lại giỏ hàng” use case kết thúc.
7. Tại bất kì thời điểm nào trong quá trình thực hiện use nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Nếu use case thành công đơn hàng sẽ được cập nhật vào hệ thống.

* Điểm mở rộng:

Không có.

1. Biểu đồ trình tự:

Diagram

Description automatically generated

*Hình 3. 28: Biểu đồ trình tự đặt hàng.*

1. Biểu đồ hoạt động:

Diagram

Description automatically generated

*Hình 3. 29: Biểu đồ hoạt động đặt hàng.*

1. Biểu đồ các lớp tham gia vào ca sử dụng:

Diagram

Description automatically generated

*Hình 3. 30: Biểu đồ các lớp tham gia ca sử dụng đặt hàng.**.*

CHƯƠNG 4: THIẾT KẾ HỆ THỐNG



## Thiết kế cơ sở dữ liệu

### *Biểu đồ mô tả cơ sở dữ liệu*



Diagram, schematic

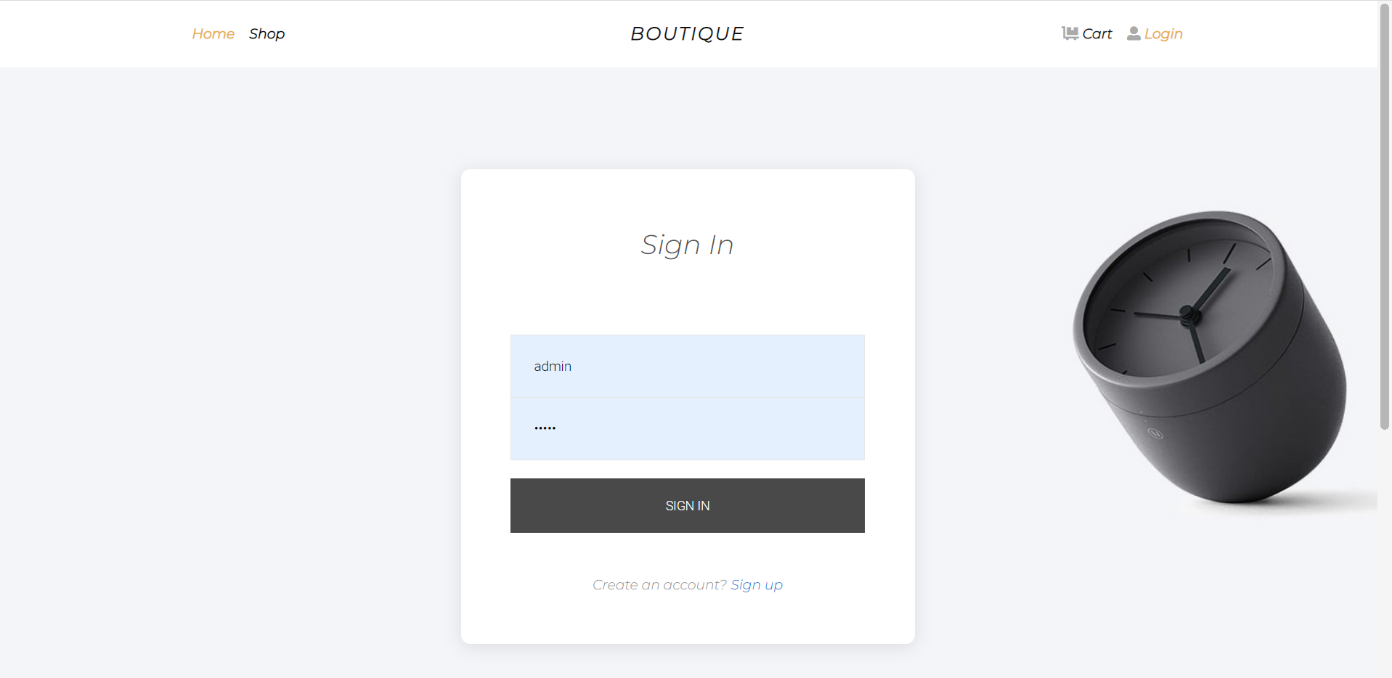
Description automatically generated

*Hình 4. 1: Biểu đồ mô tả cấu trúc database.*

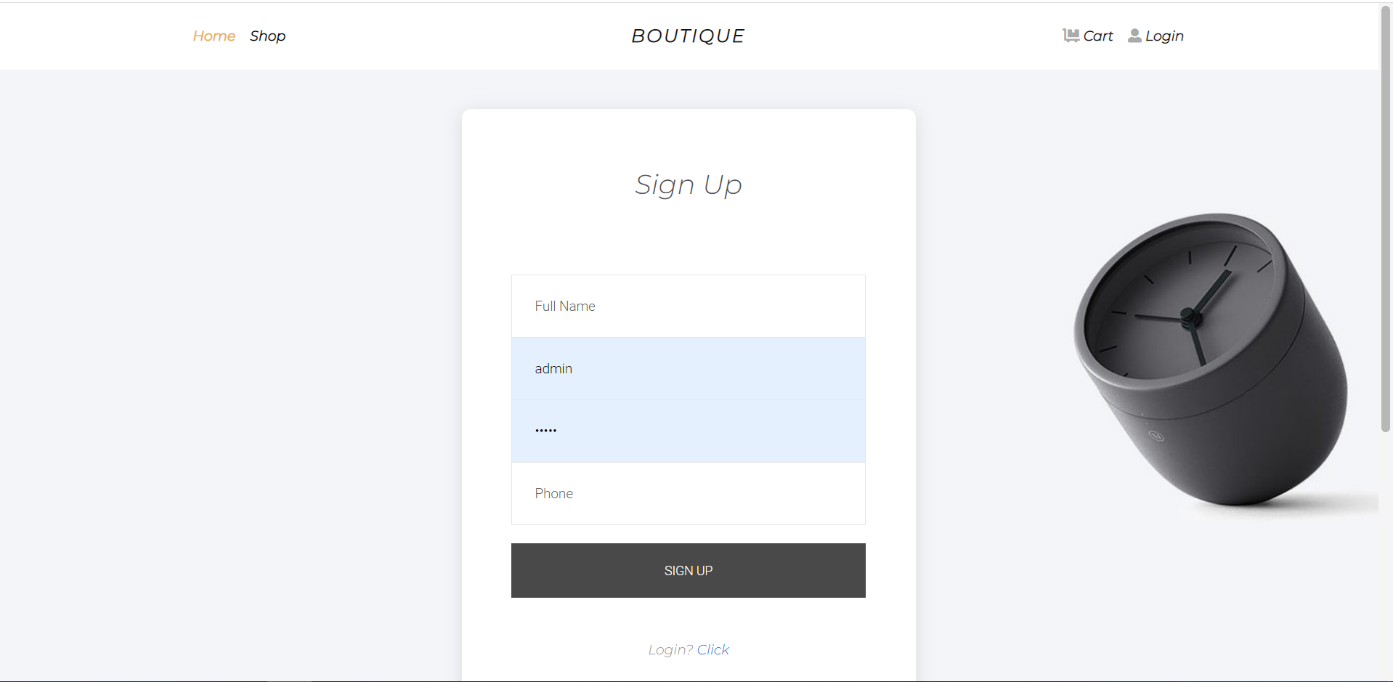


## Thiết kế giao diện

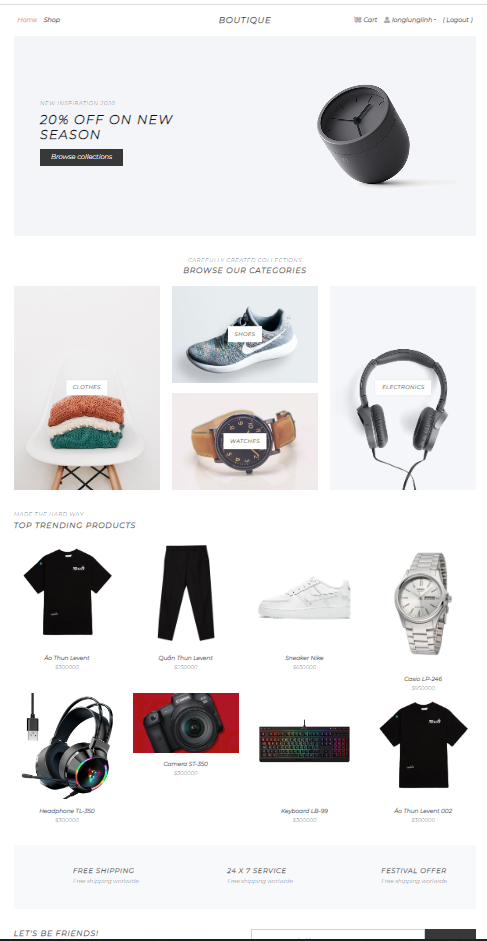
- Giao diện trang đăng nhập được minh hoạ bằng hình ảnh dưới đây.



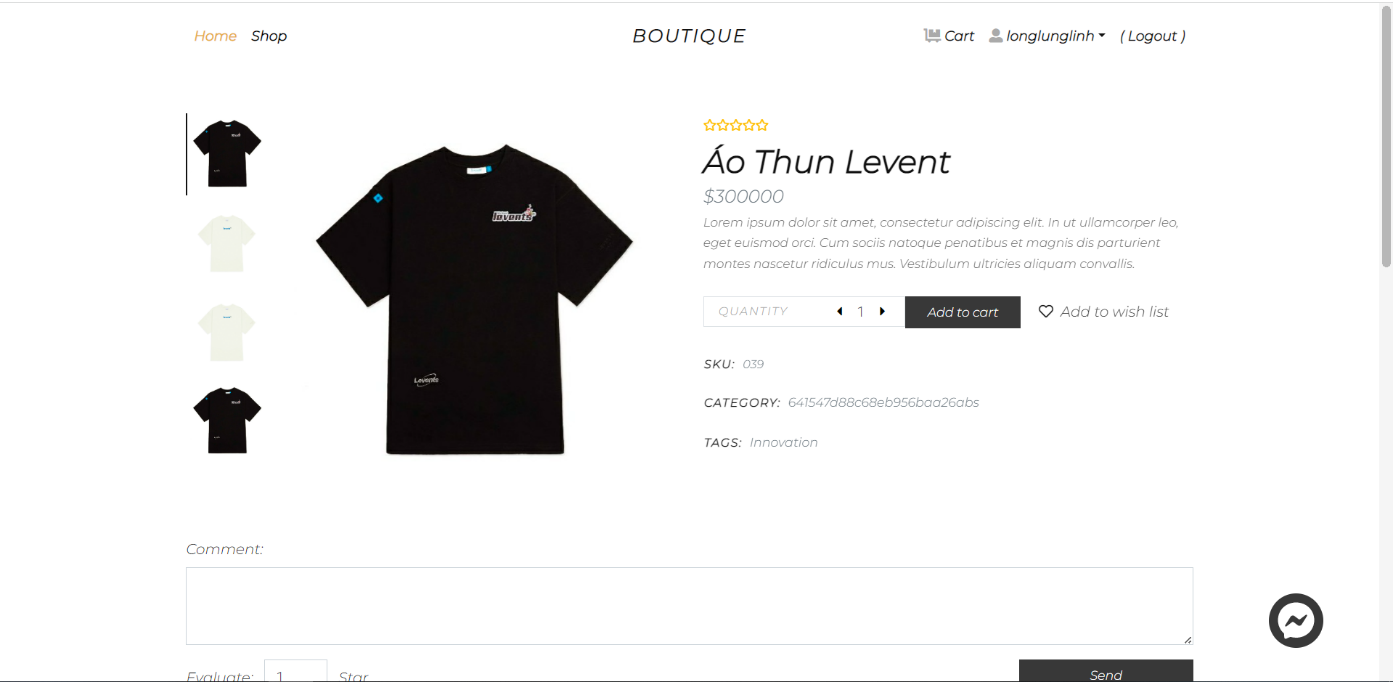
*Hình 4. 2: Giao diện trang đăng nhập.*

- Khách hàng có thể đăng ký tài khoản theo hình minh hoạ dưới đây.

*Hình 4.* *3: Giao diện trang đăng ký.*

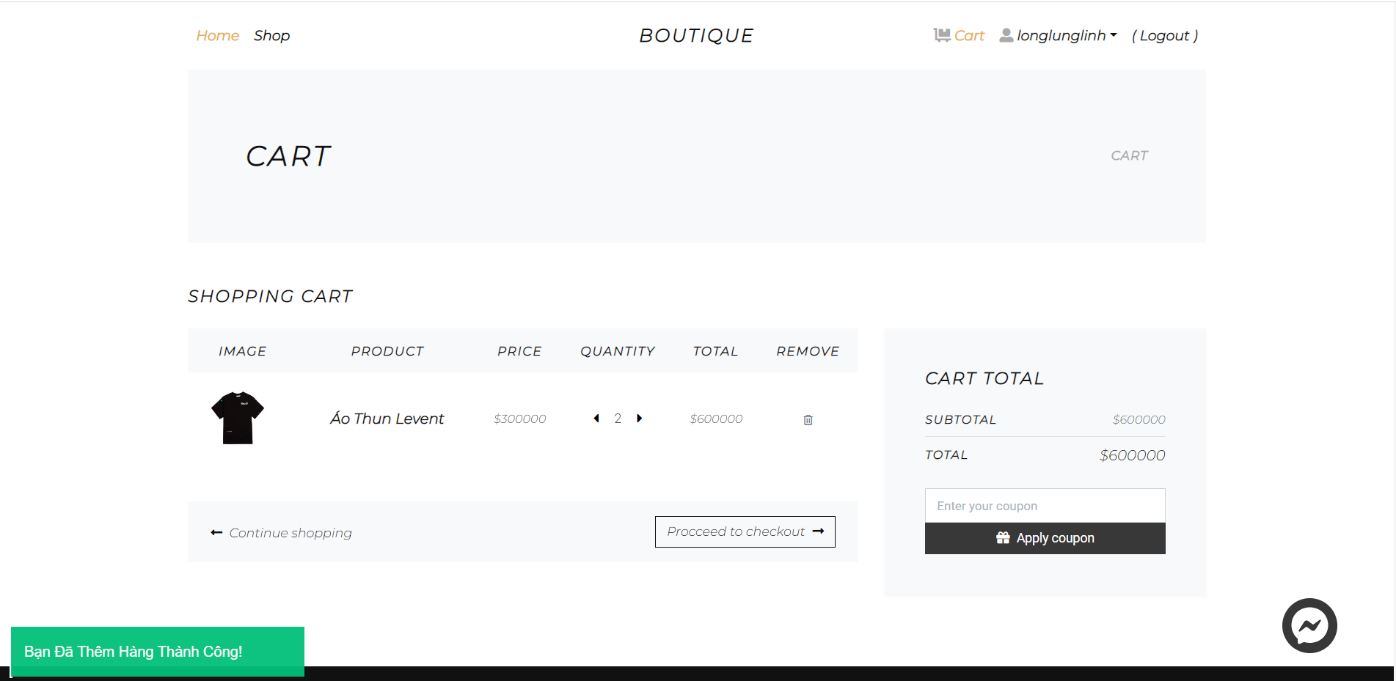
- Tại màn hình trang chủ, người dùng có thể thực hiện tìm kiếm sản phẩm theo: tên sản phẩm, danh mục, giá thành trong khoảng nào.

*Hình 4. 4: Giao diện cửa hàng.*

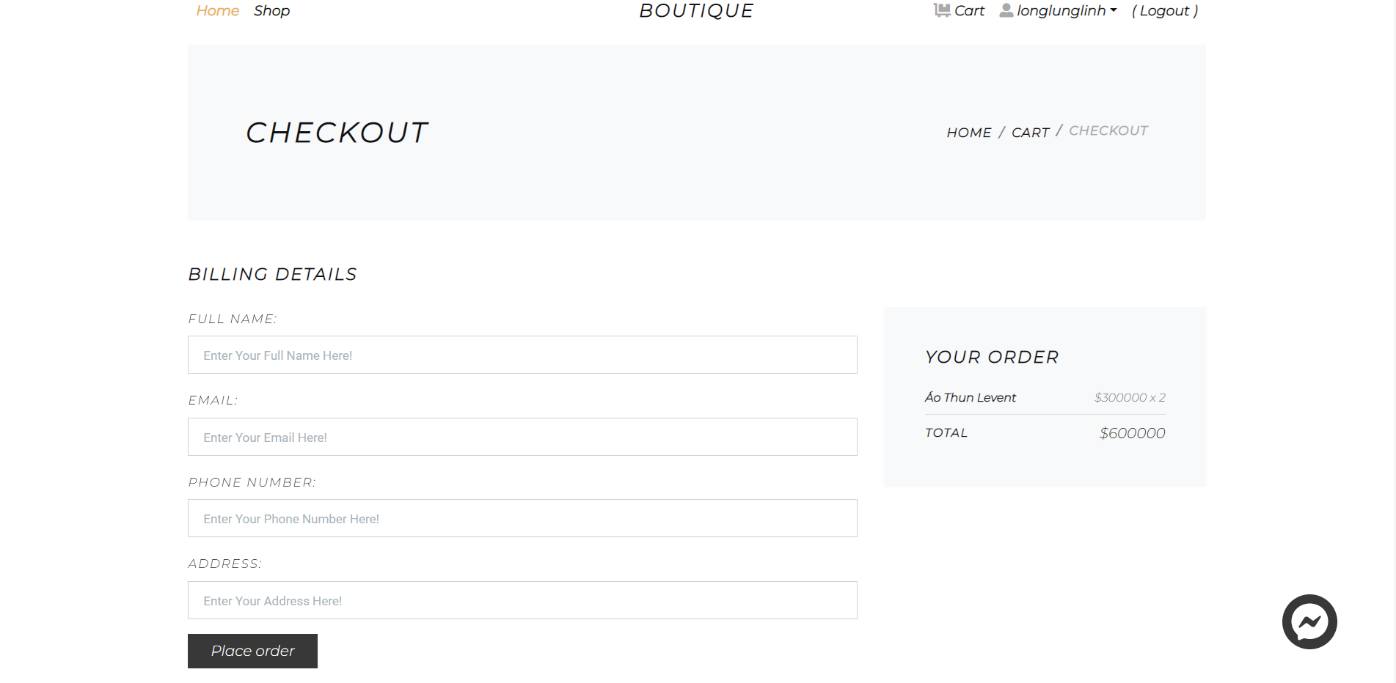
- Chi tiết sản phẩm

*Hình 4. 5: Giao diện chi tiết sản phẩm.*

- Sau khi người dùng thêm các sản phẩm vào giỏ hàng, người dùng bấm vào biểu tượng giỏ hàng để thực hiện mua hàng, chi tiết được minh hoạ như hình dưới đây.

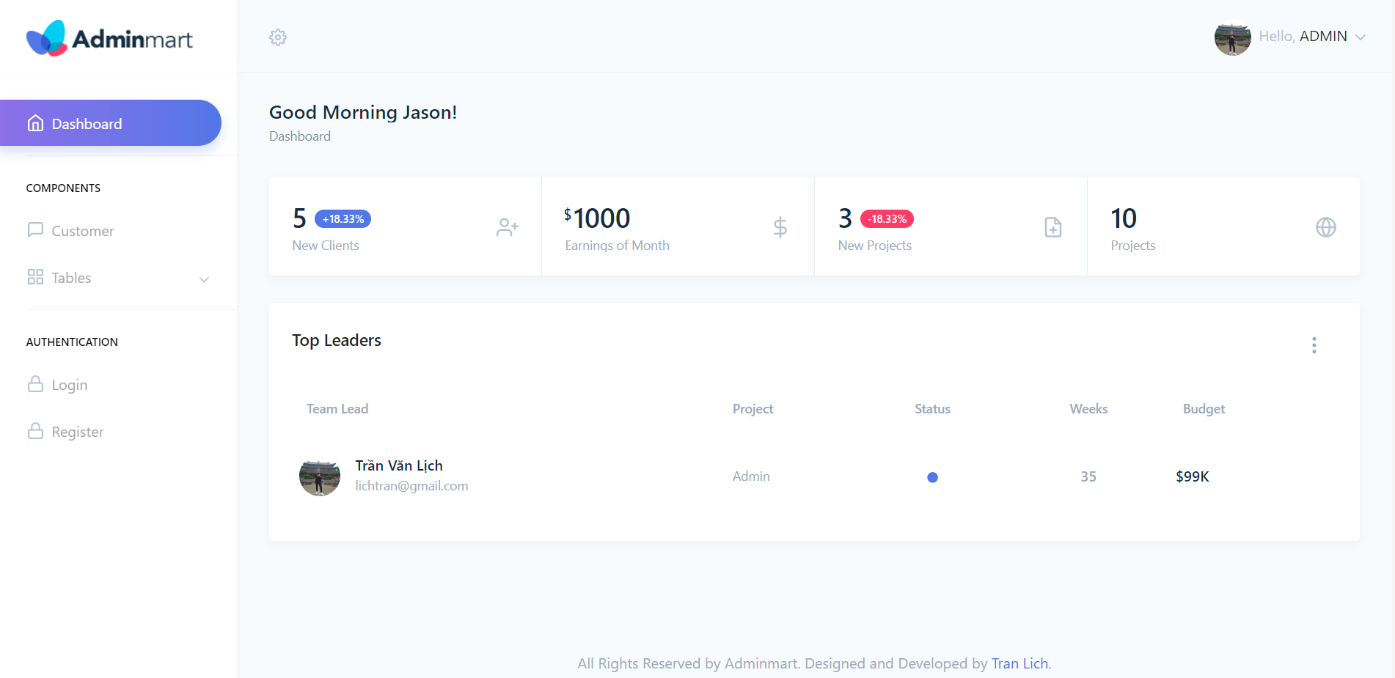


*Hình 4 .6: Giao diện đặt hàng*

* Khi người dùng muốn đặt mua hàng sẽ điền thông tin nhận hàng

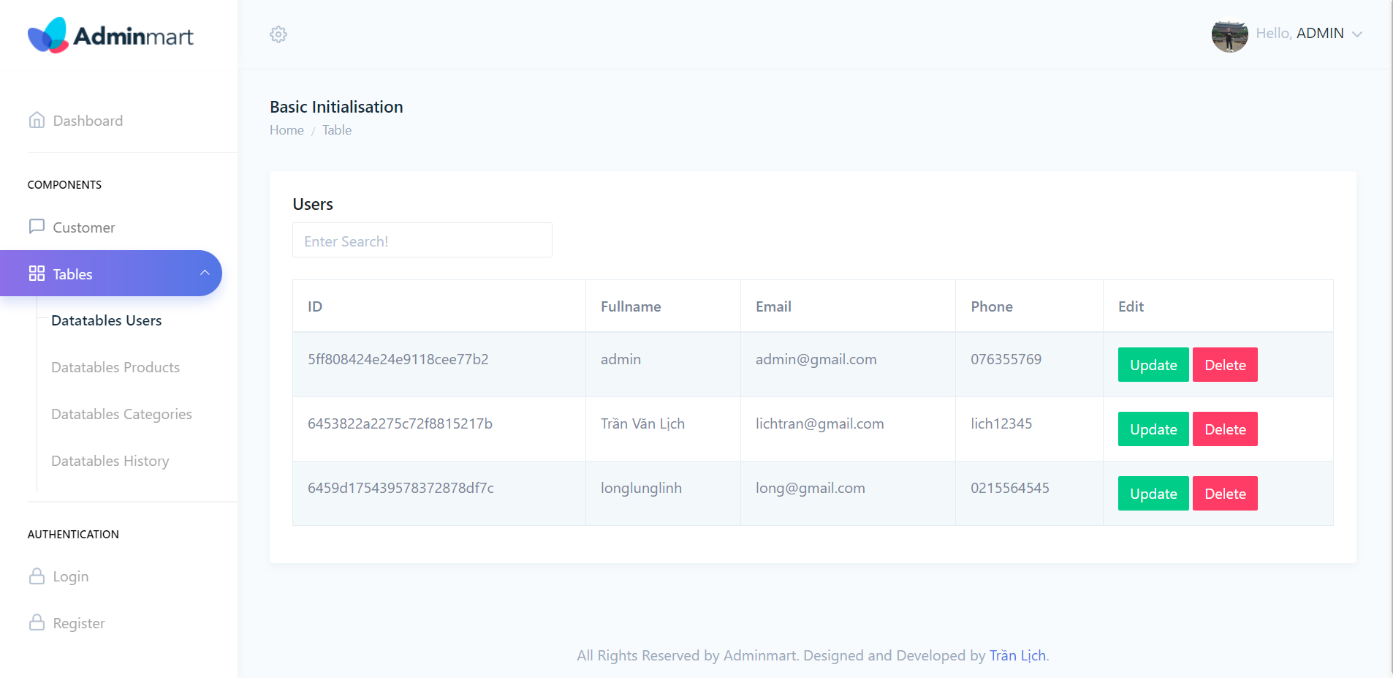
*Hình 4 .7: Giao diện sửa đổi thông tin nhận hàng.*

- Giao diện trang admin:

+ Dưới đây là hình ảnh minh hoạ cho màn hình trang chủ của người quản trị.

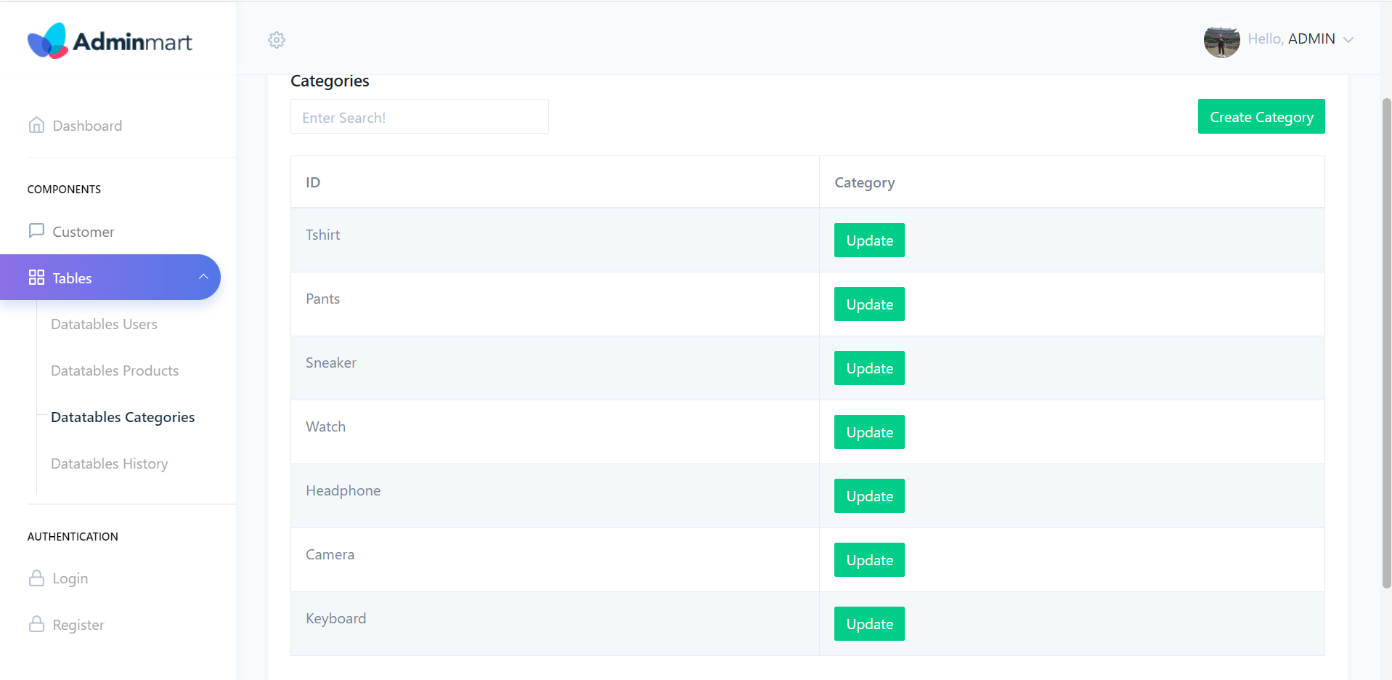
*Hình 4. 8: Giao diện trang chủ admin.*

+ Dưới đây là hình ảnh của màn hình quản lý tài khoản, tại đây người quản trị có thể thực hiện các thao tác: xem danh sách; xem thông tin khách hàng, xoá tài khoản khách hàng.

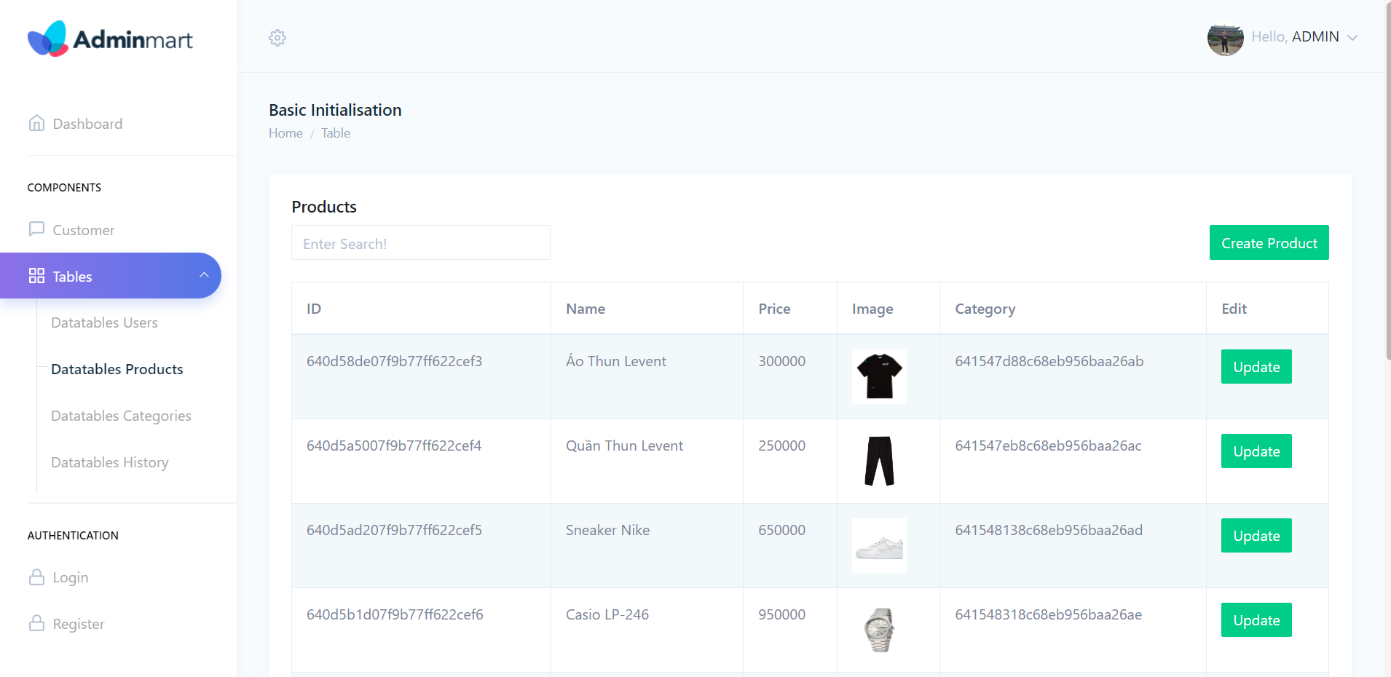


*Hình 4. 9: Giao diện quản lý tài khoản/ quản trị viên*

+ Dưới đây là hình ảnh minh hoạ màn hình thêm mới nhóm danh mục sản phẩm.

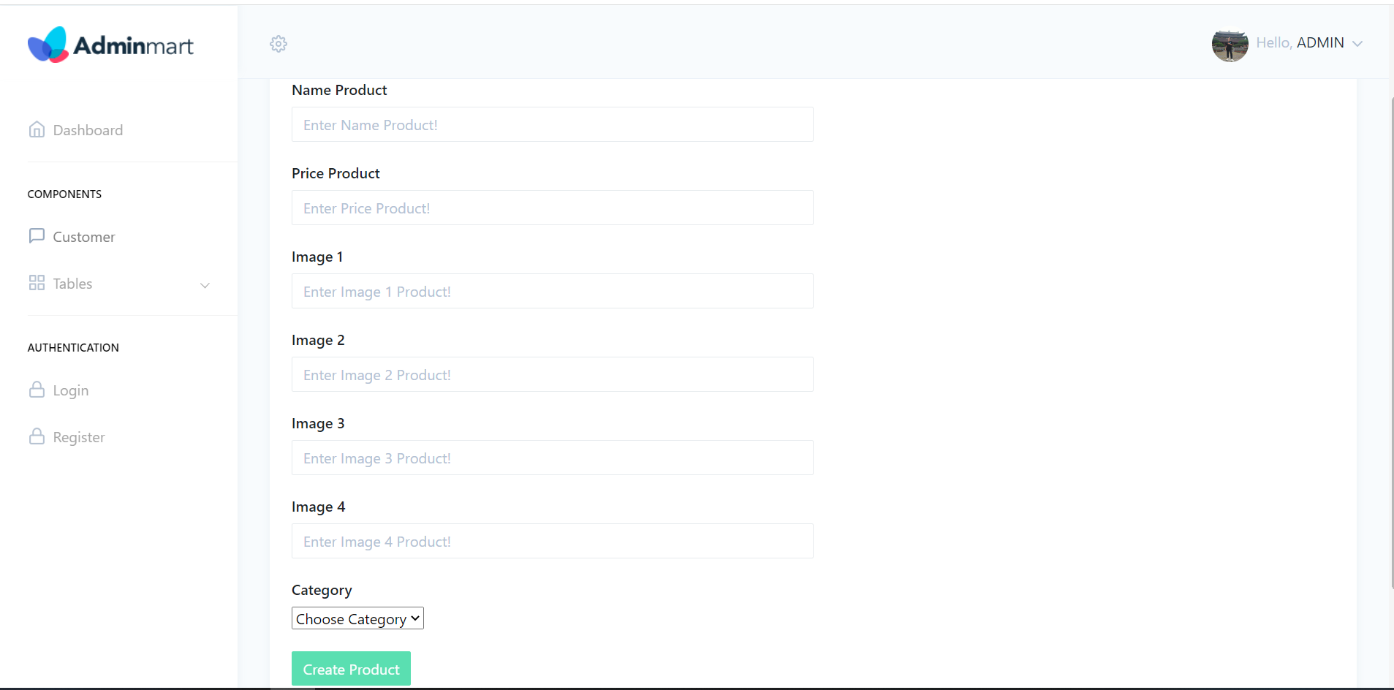


*Hình 4. 10: Giao diện trang quản lý danh mục sản phẩm.*

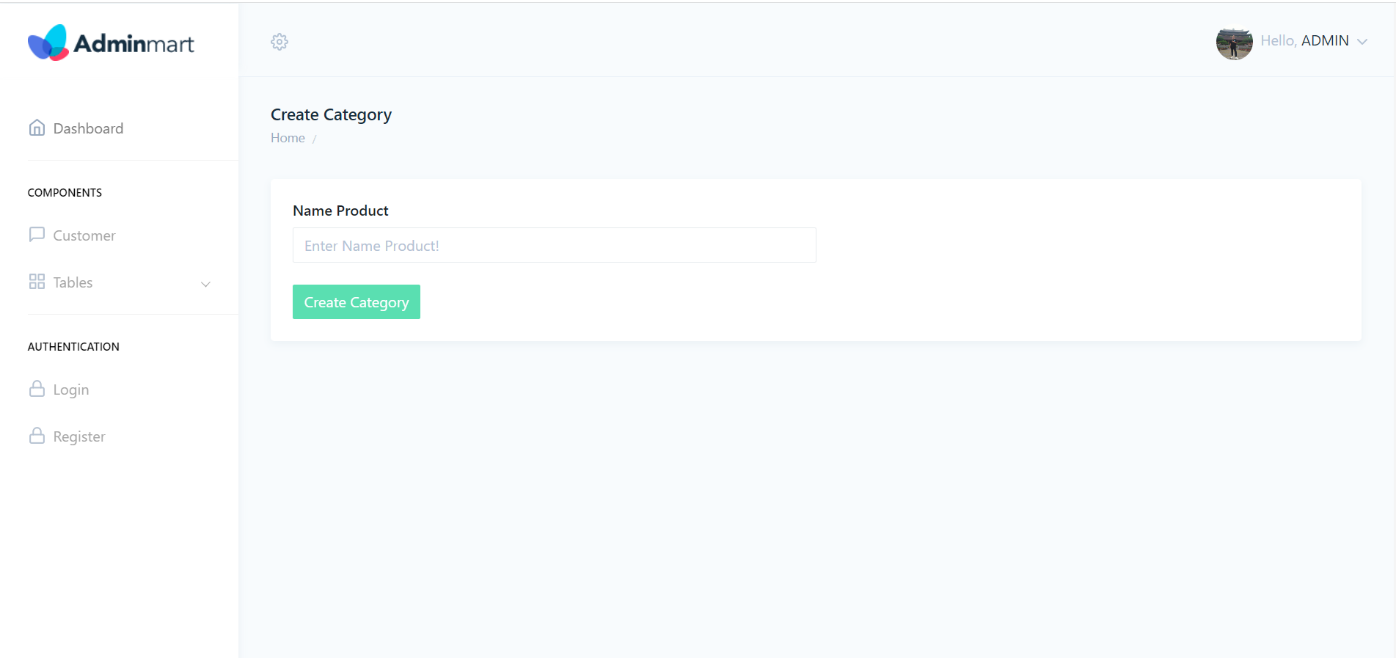
+Dưới đây là màn hình quản lý sản phẩm, tại đây người quản trị có thể thực hiện các thao tác: tìm kiếm, thêm, sửa.

*Hình 4. 11: Giao diện trang quản lý sản phẩm.*

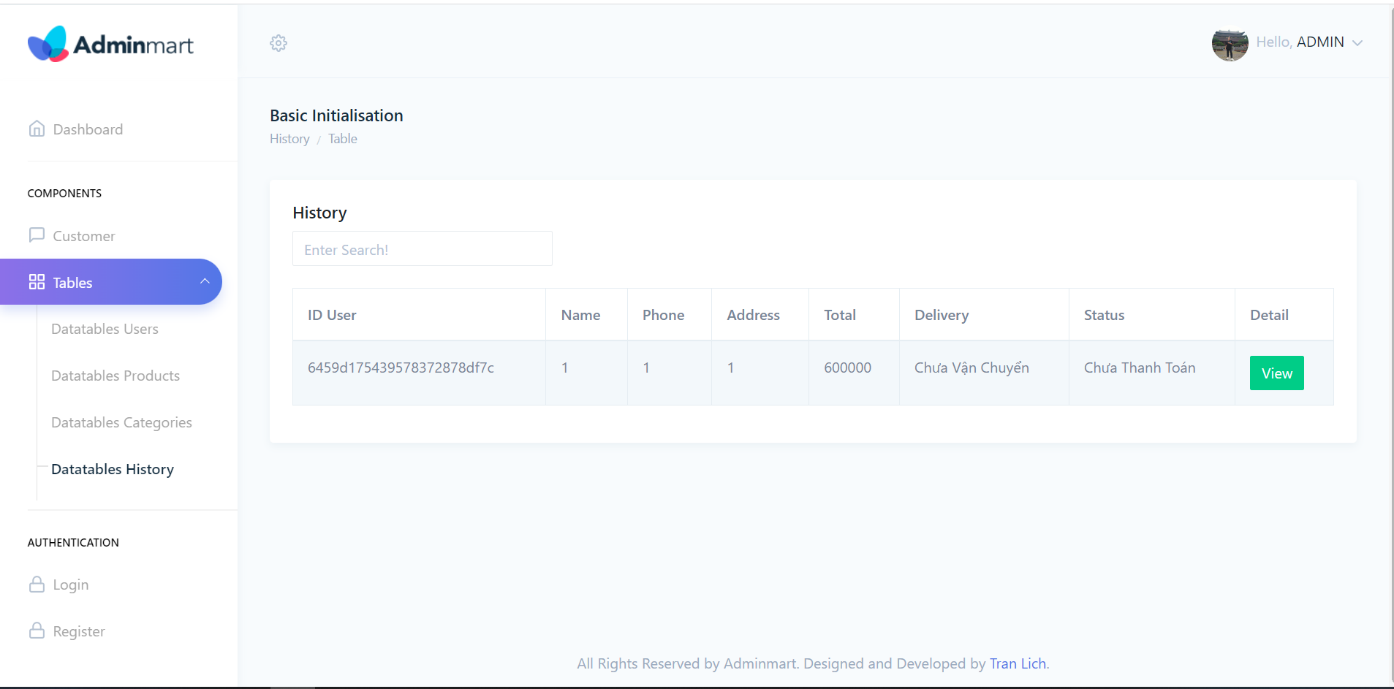
+Hình ảnh dưới đây là minh hoạ cho Form thêm mới sản phẩm.



*Hình 4. 12: Giao diện thêm mới sản phẩm.*

+ Hình ảnh dưới đây minh hoạ thao tác thêm mới danh mục:

*Hình 4. 13: Giao diện thêm mới danh mục.*

+ Hình ảnh dưới đây là minh hoạ cho màn hình quản lý hoá đơn, tại đây người quản trị có thể thực hiện các thao tác : xem đơn hàng

*Hình 4. 14: Giao diện quản lý hoá đơn.*

KẾT LUẬN

**\*Kết quả đạt được:**

Đề tài “Xây dựng website bán hàng thời trang” xuất phát từ những nhu cầu thực tế mà ngày nay trong ngành kinh doanh cần có. Những phương pháp bán hàng truyền thống dần lỗi thời và thay vào đó việc bán hàng qua website nâng cao hiệu quả hơn, khắc phục được những tồn đọng khó khăn mà phương pháp thủ công đang gặp phải.

Qua quá trình thực hiện đề tài, em đã tìm, tích lũy và học hỏi thêm được nhiều kinh nghiệm cũng như kiến thức công nghệ mới.

Những kết quả đạt được:

**Về công nghệ:**

Hiểu được quá trình thiết kế một website thực tế đi từ bước cơ bản: khảo sát dự án, phân tích hệ thống, thiết kế, thực hiện…

Xây dựng thành công website đáp ứng nhu cầu bán hàng và quảng bá các mặt hàng thời trang.

Nắm được các kiến thức xây dựng giao diện website: html, css, javascript, nodejs , reactjs

Sử dụng thành thạo các công cụ hỗ trợ lập trình: Visual Studio Code,MongoDB Compass...

**Về cài đặt chương trình:**

Cho phép khách hảng có thể xem thông tin sản phẩm và có thể mua hảng trực tuyến.

Tiếp nhận đơn đặt hàng của khách hàng.

Các mặt hàng được cập nhật liên tục theo nhu cầu của khách hàng.

Trang quản trị đầy đủ chức năng để quản lý một trang web: quản lý tài khoản, khách hảng. sản phẩm, hóa đơn…

**Tính năng khác:**

Giao diện đẹp thân thiện với khách hàng.

Tính bảo mật

**\*Hạn chế**

Code chương trình chưa được tối ưu.

Do chưa tiếp xúc với dự án thực tế nên sản phẩm chưa đáp ứng hoàn chỉnh của một trang web bán hàng thật.

Trang web bán hàng quy mô nhỏ.

Chưa hoàn thiện được chức năng trò chuyện trực tuyến.

**\*Hướng phát triển**

Tiếp tục hoàn thiện các tính năng mở rộng phần mềm nhằm cải tiến và nâng cấp chương trình.

Hoàn thiện chương trình, Sửa chữa các lỗi còn diễn ra, cải thiện tính bảo mật.

Xây dựng Website quy mô lớn hơn với nhiều ứng dụng

TÀI LIỆU THAM KHẢO

[1] **Ngô Thị Bích Thuý**. Slide môn phân tích thiết kế hệ thống.

|  |  |
| --- | --- |
| [2] | G. Sparks. (2000) An Introduction to modelling software systems using the Unified. [Online]. HYPERLINK " http://www.sparxsystems.com.au/," http://www.sparxsystems.com.au/, |
| [3] | D. Lebrknight R. LeMaster, *Object-Oriented Programming & Design”, CSCI 4448*.: University of Colorado, 2002. |
| [4] | K. Scott M. Fowler, *UML Distilled Second Edition – A Brief Guide to the Standard Object Modelling Langguage*.: Addison Wesley Book, 1999. |
| [5] | A. Munk-Madsen, P.A. Nielsen, J. Stage L. Mathiassen, *ObjectOriented Analysis&Design (OOA&D) – Concept, Principles & Methodology*. 2004. |
| Website | |
| [] | UML Notion Guide. [Online]. HYPERLINK "http://etna.int-evry.fr/COURS/UML/notation/index.html" http://etna.int-evry.fr/COURS/UML/notation/index.html |