

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

**KHOA CÔNG NGHỆ THÔNG TIN**

-----



**BÁO CÁO THỰC TẬP CƠ SỞ**

**Đề tài: XÂY DỰNG HỆ THỐNG WEBSITE BÁN  
MỸ PHẨM**

**Giảng viên hướng dẫn:**

ThS. Nguyễn Văn B

**Sinh viên thực hiện:**

Nguyễn Văn A – D19123435

Nguyễn Văn B – D19123435

Nguyễn Văn C – D19123435

**Hệ đào tạo:**

Đại học chính quy

**Hà Nội 202x**

## MỤC LỤC

<b>DANH MỤC BẢNG.....</b>	<b>iii</b>
<b>DANH MỤC HÌNH.....</b>	<b>iv</b>
<b>LỜI MỞ ĐẦU.....</b>	<b>1</b>
<b>CHƯƠNG I. GIỚI THIỆU CHUNG.....</b>	<b>1</b>
1.1 Giới thiệu chung.....	1
1.2 Mô tả đề tài.....	2
1.3 Định hướng giải quyết vấn đề.....	2
1.4 Tiến độ thực hiện.....	2
1.5 Kết luận chương.....	3
<b>CHƯƠNG II. CÔNG NGHỆ SỬ DỤNG.....</b>	<b>4</b>
2.1 Tóm tắt cơ sở lý thuyết.....	4
2.1.1 Ngôn ngữ lập trình JavaScripts.....	4
2.1.2 Frontend.....	4
2.1.3 Backend.....	6
2.1.4 Cơ sở dữ liệu.....	9
2.2 Kết luận chương.....	11
<b>CHƯƠNG III. PHÂN TÍCH VÀ THIẾT KẾ.....</b>	<b>12</b>
3.1 Phân tích hệ thống.....	12
3.1.1 Các yêu cầu đối với hệ thống.....	12
3.1.2 Các tác nhân trong hệ thống.....	12
3.1.3 Mô tả các ca sử dụng.....	13
3.1.4 Xây dựng biểu đồ ca sử dụng (usecase).....	14
3.1.5 Xây dựng sơ đồ thực thể liên kết.....	20
3.2 Thiết kế hệ thống.....	20
3.2.2 Xác định thuộc tính và gán phương thức cho các lớp.....	21
3.2.3 Thiết kế cơ sở dữ liệu.....	25
3.3 Kết luận chương.....	26
<b>CHƯƠNG IV. CÀI ĐẶT VÀ KIỂM THỬ.....</b>	<b>27</b>
4.1 Môi trường triển khai ứng dụng.....	27
4.2 Giao diện ứng dụng.....	27
4.2.1 Giao diện một số trang chính.....	27
4.2.2 Giao diện một số chức năng.....	29
<b>KẾT LUẬN.....</b>	<b>36</b>
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>38</b>

## DANH MỤC BẢNG

## DANH MỤC HÌNH

## LỜI MỞ ĐẦU

Ngày nay, ứng dụng công nghệ thông tin và tin học hóa đã có những phát triển mạnh mẽ cả về chiều rộng lẫn chiều sâu. Là những yếu tố mang tính quyết định trong hoạt động của các tổ chức, cũng như các công ty, nó đóng vai trò hết sức quan trọng và có thể tạo ra những bước đột phá mạnh mẽ. Cùng với sự phát triển không ngừng về kỹ thuật máy tính và mạng điện tử, việc ứng dụng công nghệ thông tin là phương tiện làm việc, giải trí không những ở công sở mà thậm chí còn ở ngay trong gia đình.

Mạng Internet là một trong những sản phẩm có giá trị hết sức lớn lao và ngày càng trở thành một công cụ không thể thiếu, là nền tảng chính cho sự truyền tải, trao đổi thông tin trên toàn cầu. Giờ đây, mọi việc liên quan đến thông tin trở nên thật dễ dàng cho người sử dụng, chỉ cần có một máy tính kết nối Internet và một dòng dữ liệu thì gần như lập tức cả thế giới thông tin về vấn đề mà bạn đang quan tâm sẽ hiện ra, có đầy đủ thông tin, hình ảnh, và thậm chí cả những âm thanh nếu bạn cần... Bằng Internet, chúng ta đã thực hiện được nhiều công việc với tốc độ nhanh hơn và chi phí thấp hơn so với cách truyền thống. Chính điều này, đã thúc đẩy sự khai sinh và phát triển của thương mại điện tử, làm biến đổi đáng kể bộ mặt văn hóa thế giới, nâng cao chất lượng đời sống con người.

Trong hoạt động sản xuất, kinh doanh, giờ đây, thương mại điện tử đã khẳng định được vai trò xúc tiến và thúc đẩy sự phát triển của doanh nghiệp. Đối với một cửa hàng việc bán và giới thiệu cũng như quảng bá các sản phẩm của mình đến với khách hàng là vô cùng cần thiết nhằm nâng cao năng lực kinh doanh và mở rộng, phát triển quy mô toàn diện của mình. Vì vậy, xây dựng một website cho cửa hàng là công cụ để chủ cửa hàng thực hiện quả các mục đích trên. Đồng thời giúp cho khách hàng có thể tiếp cận tìm hiểu đầy đủ, chi tiết các sản phẩm mà mình có nhu cầu và thực hiện việc mua sản phẩm một cách nhanh chóng và thuận tiện.

### Mục tiêu cần đạt được

*Yêu cầu chức năng:*

- Website hỗ trợ khách hàng trong việc tìm kiếm, tra cứu theo nhiều tiêu chí như: tìm kiếm theo tên, theo giá của sản phẩm... , lựa chọn sản phẩm, đăng ký thành viên và đăng nhập một cách nhanh chóng, dễ dàng;
- Website có khả năng lưu trữ, quản lý những thông tin về khách hàng, về đơn đặt hàng;
- Website có khả năng hỗ trợ nhà quản trị trong việc cập nhật thông tin sản phẩm, danh mục sản phẩm, cập nhật thông tin về đơn đặt hàng, xem sản phẩm;
- Website có khả năng tổng hợp, thống kê, báo cáo giúp nhà quản trị biết được doanh thu cửa hàng.

*Yêu cầu phi chức năng:*

- Giao diện thân thiện, dễ sử dụng, tính thẩm mỹ và tính ổn định cao;
- Website có thể hoạt động 24/7/365.

*Phạm vi đề tài*

**Đề tài thực hiện xây dựng hệ thống website bán mỹ phẩm với lượng dữ liệu nhỏ và trung bình. Website chạy thử trên máy cục bộ. Dựa trên kết quả của đề tài, trong thời gian tới, website có thể phát triển thêm để có thể chạy với dữ liệu lớn hơn và triển khai trên internet.**

*Giải pháp công nghệ được lựa chọn:*

**Để xây dựng hệ thống website bán hàng mỹ phẩm cho các cửa hàng vừa và nhỏ, đề tài lựa chọn công nghệ lập trình web động dùng ngôn ngữ JavaScripts và mô hình client-server 2 lớp với những lý do sau:**

- Thời gian triển khai không quá dài;
- Chi phí vừa phải, phù hợp trong khuôn khổ của đề tài.

*Phương pháp thực hiện:*

- Tiến hành thu thập và phân tích những thông tin, tài liệu liên quan đến đề tài để hình thành nên những ý tưởng tổng quan (mục đích cần đạt đến của

**Website, đối tượng cần hướng đến là ai?, thông tin gì đã có trong tay và sử dụng chúng như thế nào?);**

**- Xác định các yêu cầu nhằm phân tích thiết kế hệ thống chương trình cho phù hợp;**

**- Xây dựng chương trình theo những yêu cầu đã đặt ra;**

**- Triển khai chương trình và đánh giá kết quả đạt được.**

**Bố cục đề tài gồm 4 chương:**

**CHƯƠNG I. GIỚI THIỆU CHUNG:** Tìm hiểu về yêu cầu, thực trạng của thị trường bán mỹ phẩm cũng như xây dựng các kênh giao tiếp, quản lí... Từ đó đưa ra mục tiêu và chức năng chính của hệ thống, đưa ra hướng giải quyết vấn đề đặt ra.

**CHƯƠNG II. CÔNG NGHỆ SỬ DỤNG:** Từ các yêu cầu đặt ra cho hệ thống định hướng các giải pháp công nghệ và tóm tắt cơ sở lí thuyết.

**CHƯƠNG III. PHÂN TÍCH VÀ THIẾT KẾ:** Tiến hành phân tích hệ thống, xây dựng biểu đồ, kịch bản phù hợp với từng chức năng cụ thể. Đưa ra bản thiết kế kiến trúc tổng quan của hệ thống tiến hành thiết kế cơ sở dữ liệu phù hợp, các biểu đồ mức chi tiết, sơ đồ tuần tự các ca sử dụng trong hệ thống.

**CHƯƠNG IV. CÀI ĐẶT VÀ KIỂM THỬ:** Giai đoạn lựa chọn công nghệ, tiến hành cài đặt, lập trình hệ thống. Bên cạnh đó chương này sẽ đưa ra một số màn hình chạy thử của chương trình.

## CHƯƠNG I. GIỚI THIỆU CHUNG

### 1.1 Giới thiệu chung

Xã hội ngày nay đang không ngừng phát triển, song song với đó xu hướng mua hàng online ngày phổ biến. Mọi việc giờ đây thật đơn giản, chỉ cần có một chiếc máy tính hay thậm chí chỉ với một chiếc điện thoại thông minh có kết nối với internet, việc mua bán, trao đổi thương mại trở nên thật dễ dàng hơn bao giờ hết với tất cả mọi người.

Với việc thương mại điện tử hóa, mọi rào cản về không gian địa lý hay thời gian làm việc đều được xóa bỏ. Các sản phẩm được giới thiệu rõ ràng dành cho không chỉ những người mua hàng ở khu vực đó mà trên cả đất nước Việt Nam, thậm chí là người dân trên toàn thế giới. Người bán giờ đây không chỉ còn ngồi một chỗ chờ khách hàng tìm đến mà đã tích cực chủ động đứng lên và tìm đến khách hàng. Và khi số lượng khách hàng tăng lên thì nó cũng tỉ lệ thuận với việc doanh thu sẽ tăng, đó chính là điều mà mọi doanh nghiệp đều hướng tới.

Không chỉ dừng lại vậy, kinh doanh online còn tạo ra những cơ hội làm ăn cho những ai không đủ vốn bởi: bạn không phải mất tiền thuê mặt bằng ở những nơi đắt đỏ, thuê nhân viên, đầu tư nhiều cho việc chạy quảng cáo... mà chỉ cần đầu tư, chăm chút kỹ lưỡng cho một trang web thương mại điện tử với đầy đủ thông tin về doanh nghiệp của bạn cũng như các tính năng hỗ trợ tìm kiếm mua hàng, đưa hình ảnh, thông tin về sản phẩm. Từ đó, khách hàng sẽ có thể tiếp cận thông tin chủ động hơn, nhờ tư vấn và mua bán dễ dàng, chính xác và nhanh gọn hơn. Với tình hình cạnh tranh cực kỳ “khốc liệt” như hiện nay giữa các doanh nghiệp thì rất khó để có thể độc quyền một sản phẩm nào, bởi vậy nơi chinh phục được khách hàng chính là nơi làm họ cảm thấy thoải mái nhất, hài lòng nhất.

Bằng việc thương mại điện tử hóa, tất cả các doanh nghiệp từ lớn, vừa và nhỏ đều có thể thoải sức sáng tạo, cạnh tranh công bằng. Những ý tưởng kinh doanh mới táo bạo, những chiến lược tiếp thị, khuyến mại... đều có thể được áp dụng và hướng trực tiếp đến khách hàng nhanh nhất mà không tốn quá nhiều chi phí bởi tất cả vẫn được gói gọn trong một trang thương mại điện tử (website).



## 1.2 Mô tả đề tài

- Giới thiệu hình ảnh về loại mỹ phẩm.
- Các thông tin có liên quan và giá đi kèm.
- Các thông tin về sản phẩm sẽ luôn được update thường xuyên.
- Một số chức năng tìm kiếm như: Tìm kiếm theo giá, theo tên sản phẩm.
- Các thông tin về khuyến mại sẽ được chú trọng.
- Tư vấn, trao đổi trực tuyến với khách hàng.
- Một số hình thức thanh toán.

### 1.3 Định hướng giải quyết vấn đề

## 1.4 Tiến độ thực hiện

[illegible]

	công nghệ		các công nghệ web											
2	Phân tích và thiết kế hệ thống	25	Phân tích hệ thống, xây dựng biểu đồ, kịch bản sử dụng với các chức năng cụ thể											
3	Cài đặt hệ thống	25	Cài đặt, lập trình hệ thống											
4	Tổng kết	14	Hoàn chỉnh đề tài và viết báo cáo											

1.5 Kết luận chương

Chương này đã giới thiệu tổng quan về đề tài. Giúp cho bạn đọc hiểu rõ hơn lý do chọn đề tài dựa trên nhu cầu thực tế của xã hội cũng như nhu cầu sử dụng hệ thống bán mỹ phẩm online. Qua đó cũng giới thiệu được sơ lược các chức năng cần có trong hệ thống. Các chương tiếp theo, đề tài sẽ đi sâu vào tìm hiểu công nghệ sử dụng và bắt đầu phân tích, thiết kế hệ thống một cách chi tiết.

## CHƯƠNG II. CÔNG NGHỆ SỬ DỤNG

### 2.1 Tóm tắt cơ sở lý thuyết

#### 2.1.1 Ngôn ngữ lập trình JavaScripts

JavaScript là ngôn ngữ lập trình hướng đối tượng được tạo ra với mục đích xây dựng website. Đây được phân loại là ngôn ngữ cao cấp mang tính kịch bản. JavaScript giúp các website tương thích với người dùng. Nếu như trước đây với HTML và CSS, các website chỉ có thể hiển thị nội dung tĩnh thì giờ đây, với JavaScript, chúng ta có thể tương tác và truy xuất thông tin từ website.

JavaScript được phát triển bởi Brendan Eich tại Hãng truyền thông Netscape với cái tên đầu tiên Mocha, rồi sau đó đổi tên thành LiveScript, và cuối cùng thành JavaScript. Giống Java, JavaScript có cú pháp tương tự C, nhưng nó gần với Self hơn Java. Hiện nay phiên bản mới nhất của JavaScript là ECMAScript 12 với nhiều cải tiến vượt bậc so với phiên bản tiền nhiệm.[1]

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 viện của bên thứ 3 nên có thể tăng thêm chức năng cho website mà không phải code từ đầu.

Nếu xem ngôn ngữ lập trình như là việc xây ngôi nhà, HTML sẽ định dạng kiến trúc của căn nhà, CSS sẽ là thảm và tường để trang trí ngôi nhà đẹp hơn. JavaScript thêm yếu tố tương tác trong ngôi nhà, như là việc mở cánh cửa và làm đèn sáng. Bạn vẫn có thể làm web mà không có JavaScript nhưng rủi ro là website của bạn trông như là những web thời những năm 1995.

JavaScript là ngôn ngữ kịch bản phía máy khách (client-side) duy nhất. Do đó, nó được sử dụng trên hầu hết các trang web mà bạn nhìn thấy trên internet. Tuy nhiên cú pháp của nó vô cùng lộn xộn và khó làm chủ, do đó JQuery – một thư viện của JavaScript ra đời nhằm giúp lập trình viên xây dựng những chức năng có sử dụng Javascript trở nên dễ dàng hơn như: đơn giản hóa việc duyệt tài liệu HTML, xử lý sự kiện, hoạt ảnh và tương tác Ajax để phát triển web nhanh chóng... JQuery là một bộ công cụ JavaScript gọn nhẹ nhưng đầy mạnh mẽ, được chứng minh qua hơn 90% các website trên thế giới sử dụng (trừ các website chạy bằng JavaScript Framework).[4]

#### 2.1.2 Frontend

*Phần front-end của một trang web là phần tương tác với người dùng. Tất cả mọi thứ bạn nhìn thấy khi điều hướng trên Internet, từ các font chữ, màu sắc cho tới các menu xổ xuống và các thanh trượt, là một sự kết hợp của HTML, CSS, và JavaScript được điều khiển bởi trình duyệt máy tính.*

### **2.1.2.1 HTML**

HTML là từ viết tắt của Hypertext Markup Language, là sự kết hợp của Hypertext và Markup, hay còn được gọi là ngôn ngữ siêu văn bản. HTML không phải là ngôn ngữ lập trình, mà chỉ là ngôn ngữ đánh dấu, nó đơn giản và dễ học ngay cả với những người mới học làm web. HTML có chức năng giúp người dùng xây dựng và cấu trúc các phần trong trang web hoặc ứng dụng, thường được sử dụng trong phân chia các đoạn văn, heading, link, blockquotes,...[1]

HTML được tạo ra bởi Tim Berners-Lee, một nhà vật lý tại viện nghiên cứu CERN của Thụy Sĩ. Ông đã đưa ra ý tưởng về hệ thống siêu văn bản trên Internet. Siêu văn bản nghĩa là văn bản có thể chứa liên kết đến văn bản khác mà người dùng có thể truy cập ngay lập tức.

### **2.1.2.2 CSS**

CSS là ngôn ngữ tạo phong cách cho trang web – Cascading Style Sheet language. Nó dùng để tạo phong cách và định kiểu cho những yếu tố được viết dưới dạng ngôn ngữ đánh dấu, như là HTML. Nó có thể điều khiển định dạng của nhiều trang web cùng lúc để tiết kiệm công sức cho người viết web. Nó phân biệt cách hiển thị của trang web với nội dung chính của trang bằng cách điều khiển bố cục, màu sắc, và font chữ.

CSS được phát triển bởi **W3C** (World Wide Web Consortium) vào năm 1996, vì một lý do đơn giản. HTML không được thiết kế để gắn tag để giúp định dạng trang web.[1]

Giống như HTML, CSS không thực sự là một ngôn ngữ lập trình mà là một ngôn ngữ định kiểu – style. Điều này có nghĩa là nó cho phép bạn áp dụng kiểu có chọn lọc cho các phần tử trong tài liệu HTML.

### **2.1.2.3 Framework ReactJS**

**ReactJS** là một thư viện **JavaScript** mã nguồn mở được phát triển bởi Facebook, ra mắt vào năm 2013 với mục đích để xây dựng giao diện người dùng. Nó được sử dụng rộng rãi để xây dựng các trang web **SPA (Single Page Application)** và các ứng dụng trên nền tảng di động. Nó rất dễ sử dụng và cho phép người dùng có thể tạo các component UI có thể tái sử dụng.

ReactJS có những tính năng hết sức nổi bật bao gồm [1,2]:

- *JSX*: viết tắt của JavaScript extension, nó là React extension , giúp cho việc thay đổi cây DOM dễ dàng hơn bằng HTML-style code đơn giản. Nó là một trong những tính năng tốt và dễ sử dụng.

- *Components*: Một trang web được xây dựng bằng ReactJS là một sự kết hợp nhiều component lại với nhau chứ không phải chung một Template như bình thường. Các component cũng như các hàm JavaScript bình thường, giúp tạo ra các code dễ dàng bằng cách tách các logic ra thành các đoạn code độc lập có thể tái sử dụng. Chúng ta có thể sử dụng component dưới dạng function hoặc class, ngoài ra các component còn có state và props.

- *Virtual DOM*: ReactJS tạo một thứ gọi là Virtual DOM (DOM ảo). Đúng như tên gọi, nó là một copy của DOM thật trên trang web đó. ReactJS dùng những DOM ảo đó để tìm đúng những DOM thật cần được cập nhật khi có bất kỳ sự kiện nào làm các thành phần bên trong nó thay đổi.

- *Javascript Expressions*: Biểu thức JS có thể sử dụng trong file .jsx hoặc .js bằng cách sử dụng cặp dấu ngoặc nhọn “{}”.

### 2.1.3 Backend

Nếu Front End Developer có quyền lực kiến tạo nên vẻ đẹp của các trang web, thì Back End developer là người xử lý mọi logic nghiệp vụ phức tạp ở ẩn ở phía sau, giúp cho hệ thống hoạt động trơn tru. Dữ liệu của người dùng, thuật toán phân tích ... đều nằm ở back-end.

#### 2.1.3.1 Platform NodeJS

NodeJS là một nền tảng (platform) không phải là một web framework như một số người thường nhầm lẫn. Bên cạnh đó NodeJS cũng không phải một ngôn ngữ lập trình mà là một môi trường runtime dùng để chạy JavaScripts bên ngoài trình duyệt. Vì

vậy để học và sử dụng NodeJS thì người dùng cần có những kiến thức nhất định về một số giao thức, JavaScript hay kỹ thuật lập trình,... NodeJS có thể được sử dụng như một frontend lẫn backend nhờ vào các framework như Express.js hay Meteor.js. Một số stack phổ biến như MERN sử dụng Express.js làm backend. Bên cạnh đó, nhiều thành phần cũng có thể được tái sử dụng giữa backend và frontend vô cùng tiện lợi.

NodeJS là một trong những nền tảng phổ biến nhất hiện nay cho mục đích phát triển ứng dụng mạng phía server. Những đặc điểm khiến NodeJS là lựa chọn hàng đầu cho các developer hiện nay:

- *Tốc độ cực nhanh*: Được xây dựng dựa trên engine JavaScript V8 của Google Chrome, do đó các thư viện của nó có khả năng thực thi code chỉ rất nhanh.
- *NPM*: Với hơn 50,000 package khác nhau, các developer có thể dễ dàng lựa chọn bất kỳ tính năng nào để xây dựng cho ứng dụng của mình.
- *Lập trình không đồng bộ*: Mọi API của NodeJS đều có tính không đồng bộ (non-blocking), do đó một server dựa trên NodeJS không cần phải đợi API trả về dữ liệu.
- *Không có buffering*: NodeJS giúp tiết kiệm thời gian xử lý file khi cần upload âm thanh hoặc video vì các ứng dụng này không bao giờ buffer dữ liệu mà chỉ xuất dữ liệu theo từng phần (chunk).
- *Đơn luồng*: NodeJS sử dụng mô hình đơn luồng với vòng lặp sự kiện. Do đó các ứng dụng có thể xử lý số lượng request lớn hơn rất nhiều so với các server truyền thống như Apache HTTP Server.

### 2.1.3.2 ExpressJS Framework

**Expressjs** là một framework được xây dựng trên nền tảng của **Nodejs**. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. **Expressjs** hỗ trợ các method HTTP và middleware tạo ra API vô cùng mạnh mẽ và dễ sử dụng.

Vì Express js chỉ yêu cầu ngôn ngữ lập trình Javascript nên việc xây dựng các ứng dụng web và API trở nên đơn giản hơn với các lập trình viên và nhà phát

triển. Expressjs cũng là một khuôn khổ của Node.js do đó hầu hết các mã code đã được viết sẵn cho các lập trình viên có thể làm việc.

Nhờ có Expressjs mà các nhà lập trình có thể dễ dàng tạo các ứng dụng 1 web, nhiều web hoặc kết hợp. Do có dung lượng khá nhẹ, Expressjs giúp cho việc tổ chức các ứng dụng web thành một kiến trúc MVC có tổ chức hơn. Để có thể sử dụng được mã nguồn này, chúng ta cần phải biết về Javascript và HTML.

Expressjs cũng là một phần của công nghệ giúp quản lý các ứng dụng web một cách dễ dàng hơn hay còn được gọi là ngăn xếp phần mềm MEAN. Nhờ có thư viện Javascript của Express js đã giúp cho các nhà lập trình xây dựng nên các ứng dụng web hiệu quả và nhanh chóng hơn. Expressjs cũng được sử dụng để nâng cao các chức năng của Node.js.[8]

### 2.1.3.3 Loopback 4 Framework

**Loopback** chạy tại chỗ hoặc trong đám mây. Nó cũng cung cấp một số [thành phần bổ](#) trợ để quản lý tệp, đăng nhập của bên thứ 3, OAuth2, nhà cung cấp lưu trữ, v.v. Nói một cách đơn giản, Loopback cho phép bạn tạo API REST trong vài phút. Nó làm điều đó bằng cách:

- Cung cấp một công cụ CLI để sử dụng.
- Cung cấp trình khám phá API tích hợp.
- Cho phép nhà phát triển tạo các mô hình dựa trên một lược đồ cụ thể.
- Cho phép nhà phát triển tạo các mô hình động trong trường hợp không có lược đồ.
- Cung cấp SDK cho iOS, Android và AngularJS để dễ dàng tạo ứng dụng khách.

Loopback 4 là một framework mã nguồn mở dựa trên NodeJs và ExpressJs. Nó cho phép nhanh chóng tạo ra APIs và microservices từ phía Backend như là SOAP hoặc REST services.

Ưu điểm mà ta có thể nhận thấy ngay là khả năng tạo ra các controller, model, repository, entity một cách nhanh chóng và Loopback 4 hỗ trợ kết nối với nhiều cơ sở dữ liệu trong một project.

#### **2.1.4. Cơ sở dữ liệu**

Cơ sở dữ liệu là hệ thống bao gồm rất nhiều thông tin, dữ liệu được xây dựng theo một cấu trúc nhất định nhằm đáp ứng nhu cầu khai thác, sử dụng của nhiều người hay chạy nhiều chương trình ứng dụng cùng một lúc. Khi áp dụng hình thức lưu trữ này, nó sẽ giúp khắc phục được những điểm yếu của việc lưu file thông thường trên máy tính. Các thông tin lưu trữ sẽ đảm bảo được nhất quán, hạn chế tình trạng trùng lặp thông tin. Đề tài sẽ sử dụng cơ sở dữ liệu phi quan hệ.

##### **2.1.4.1 NoSQL**

NoSQL là 1 dạng cơ sở dữ liệu (CSDL) mã nguồn mở và được viết tắt bởi: None-Relational SQL hay có nơi thường gọi là Not-Only SQL.

NoSQL được phát triển trên Javascript Framework với kiểu dữ liệu là JSON và dạng dữ liệu theo kiểu key và value.

NoSQL ra đời như là 1 mảnh vá cho những khuyết điểm và thiếu sót cũng như hạn chế của mô hình dữ liệu quan hệ RDBMS (Relational Database Management System - Hệ quản trị cơ sở dữ liệu quan hệ) về tốc độ, tính năng, khả năng mở rộng,...

Với NoSQL chúng ta có thể mở rộng dữ liệu mà không lo tới những việc như tạo khóa ngoại, khóa chính, kiểm tra ràng buộc .v.v ...

NoSQL bỏ qua tính toàn vẹn của dữ liệu và transaction để đổi lấy hiệu suất nhanh và khả năng mở rộng.

NoSQL được sử dụng ở rất nhiều công ty, tập đoàn lớn, ví dụ như FaceBook sử dụng Cassandra do FaceBook phát triển, Google phát triển và sử dụng BigTable,...[1,4]

##### **2.1.4.2 MongoDB**

MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở, là CSDL thuộc NoSql và được hàng triệu người sử dụng.

MongoDB là một database hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu JSON thay vì dạng bảng như CSDL quan hệ nên truy vấn sẽ rất nhanh.



Với CSDL quan hệ chúng ta có khái niệm bảng, các cơ sở dữ liệu quan hệ (như MySQL hay SQL Server...) sử dụng các bảng để lưu dữ liệu thì với MongoDB chúng ta sẽ dùng khái niệm là collection thay vì bảng.

So với RDBMS thì trong MongoDB collection ứng với table, còn document sẽ ứng với row, MongoDB sẽ dùng các document thay cho row trong RDBMS.

Các collection trong MongoDB được cấu trúc rất linh hoạt, cho phép các dữ liệu lưu trữ không cần tuân theo một cấu trúc nhất định.

Thông tin liên quan được lưu trữ cùng nhau để truy cập truy vấn nhanh thông qua ngôn ngữ truy vấn MongoDB.[4]

#### ❖ Ưu điểm của MongoDB

Do MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ có các kích cỡ và các document khác nhau, linh hoạt trong việc lưu trữ dữ liệu, nên bạn muốn gì thì cứ insert vào thoải mái.

Dữ liệu trong MongoDB không có sự ràng buộc lẫn nhau, không có join như trong RDBMS nên khi insert, xóa hay update nó không cần phải mất thời gian kiểm tra xem có thỏa mãn các ràng buộc dữ liệu như trong RDBMS.

MongoDB rất dễ mở rộng (Horizontal Scalability). Trong MongoDB có một khái niệm cluster là cụm các node chứa dữ liệu giao tiếp với nhau, khi muốn mở rộng hệ thống ta chỉ cần thêm một node vào cluster.

Trường dữ liệu “\_id” luôn được tự động đánh index (chỉ mục) để tốc độ truy vấn thông tin đạt hiệu suất cao nhất.

Khi có một truy vấn dữ liệu, bản ghi được cached lên bộ nhớ Ram, để phục vụ lượt truy vấn sau diễn ra nhanh hơn mà không cần phải đọc từ ổ cứng.

Hiệu năng cao: Tốc độ truy vấn (find, update, insert, delete) của MongoDB nhanh hơn hẳn so với các hệ quản trị cơ sở dữ liệu quan hệ (RDBMS). Với một lượng dữ liệu đủ lớn thì thử nghiệm cho thấy tốc độ insert của MongoDB có thể nhanh tới gấp 100 lần so với MySQL.

#### ❖ Nhược điểm của MongoDB

Một ưu điểm của MongoDB cũng chính là nhược điểm của nó. MongoDB không có các tính chất ràng buộc như trong RDBMS nên khi thao tác với mongoDB thì phải hết sức cẩn thận.

Tồn bộ nhớ do dữ liệu lưu dưới dạng key-value, các collection chỉ khác về value do đó key sẽ bị lặp lại. Không hỗ trợ join nên dễ bị dữ thừa dữ liệu.

Khi insert/update/remove bản ghi, MongoDB sẽ chưa cập nhật ngay xuống ổ cứng, mà sau 60 giây MongoDB mới thực hiện ghi toàn bộ dữ liệu thay đổi từ RAM xuống ổ cứng điều này sẽ là nhược điểm vì sẽ có nguy cơ bị mất dữ liệu khi xảy ra các tình huống như mất điện...[4]

## **2.2 Kết luận chương**

Qua nội dung của chương 2, đề tài cũng đã lựa chọn được giải pháp công nghệ để phát triển website trên nền tảng NodeJS, hệ quản trị cơ sở phi dữ liệu MongoDB cùng với ngôn ngữ frontend HTML, CSS, ReactJS. Từ cơ sở này, ở chương 3, đề tài sẽ tiếp tục đi phân tích hệ thống và thiết kế chi tiết các chức năng sẽ có trong hệ thống bán hàng mỹ phẩm.

## CHƯƠNG III. PHÂN TÍCH VÀ THIẾT KẾ

### 3.1 Phân tích hệ thống

#### 3.1.1 Các yêu cầu đối với hệ thống

Hệ thống là một website bán mỹ phẩm nên mọi doanh thu và lợi nhuận đều đến từ phía khách hàng. Do đó các yêu cầu đặt ra cho hệ thống cũng phải phù hợp với nhu cầu của khách hàng.

Yêu cầu về giao diện website: phải có giao diện thân thiện, bắt mắt, dễ sử dụng, phù hợp với thị yếu và nhu cầu của người dùng. Không nên có quá nhiều hiệu ứng, hoa mỹ, lòe loẹt, nhiều chi tiết không cần thiết gây mỏi mắt. Trang chủ phải hiển thị bao quát toàn bộ hệ thống và phải làm nổi bật những sản phẩm mới, sản phẩm ưa chuộng.

Về hiệu năng của hệ thống: hệ thống phải có hiệu năng xử lý yêu cầu của khách hàng nhanh nhất có thể, ngay lập tức phải hồi lại khi người dùng click vào một chức năng cụ thể. Bên cạnh hiệu năng nhanh còn phải xử lý chính xác yêu cầu của khách hàng và đảm bảo an toàn bảo mật cho khách hàng.

Về các chức năng của hệ thống: đảm bảo hệ thống có đầy đủ chức năng cần thiết của một trang bán hàng online. Khách hàng có thể tạo tài khoản và đăng nhập, tìm kiếm, thêm sản phẩm vào giỏ hàng, mua hàng... Người quản trị có thể quản lý tất cả các người dùng trong hệ thống, quản lý sản phẩm, quản lý đơn hàng, thống kê doanh số...

#### 3.1.2 Các tác nhân trong hệ thống

Hệ thống có ba tác nhân chính là: Người dùng chưa đăng nhập (Guest), Người dùng đã đăng nhập (User) và Người quản trị (Admin).

*Guest (khách vãng lai)*: là những người truy cập vào website nhưng chưa đăng nhập. Mục đích của hệ thống hướng tới những tác nhân này chỉ nhằm quảng bá cho website, do đó những tác nhân này chỉ được sử dụng các chức năng cơ bản như: tìm kiếm, xem sản phẩm, bài viết, đánh giá – bình luận của những người dùng đăng nhập, khuyến mại của cửa hàng.

*User (khách hàng có tài khoản)*: là những người dùng đã đăng nhập vào hệ thống. Những tác nhân này là mục đích chính mà hệ thống hướng tới do đó ngoài các

chức năng cơ bản mà Guest được sử dụng thì còn được sử dụng thêm nhiều chức năng nâng cao khác, như là: quản lý tài khoản, mua sản phẩm, đánh giá – bình luận về 1 bài viết hay sản phẩm. Hơn nữa những tác nhân này còn có thể nhận được thông báo từ hệ thống về tình trạng đơn hàng...

*Admin (người quản trị hệ thống):* là tác nhân giữ vai trò chính trong hệ thống website. Là người điều hành, quản lý và theo dõi mọi hoạt động của hệ thống. Tác nhân có thể thực hiện được tất cả chức năng quản trị trong hệ thống như là: quản lý kho hàng, quản lý các tài khoản thành viên khác, xử lý đơn hàng, thống kê doanh thu, ...

### **3.1.3 Mô tả các ca sử dụng**

**Đăng nhập:** Người dùng đăng nhập bằng tài khoản đã đăng kí trên hệ thống để có thể sử dụng các chức năng đầy đủ nhất của hệ thống.

**Đăng kí:** Người dùng nhập thông tin đăng kí hợp lệ, hệ thống kiểm tra và xác thực sau đó lưu vào cơ sở dữ liệu.

**Quản lý thông tin tài khoản:** người dùng thay đổi thông tin cá nhân của mình.

**Tìm kiếm sản phẩm:** Người dùng tìm kiếm sản phẩm theo loại.

**Tạo giỏ hàng:** Người dùng thêm sản phẩm vào giỏ hàng để tiếp tục lựa chọn các sản phẩm khác, người dùng có thể quản lý các sản phẩm trong giỏ hàng của mình, khi tất trình duyệt thì sản phẩm sẽ bị xóa, chức năng này áp dụng cho Guest(khách vãng lai) và user(người có tài khoản).

**Bình luận và đánh giá sản phẩm:** Người dùng có thể gửi bình luận và đánh giá của mình về một sản phẩm. Mọi bình luận và đánh giá đều được hiển thị công khai và mỗi tài khoản chỉ được đánh giá và bình luận một lần, những lần tiếp theo sẽ ghi đè lên lần trước để đảm bảo kết quả là chính xác và công bằng nhất. Chức năng này hiện tại chỉ áp dụng cho người dùng đã đăng nhập (User).

**Quản lý nhập liệu:** Sẽ bao gồm việc nhân viên quản lý sản phẩm, thể loại, quản lý bình luận và quản lý ảnh trình chiếu trên Slideshow.

Quản lý bình luận: Nhân viên quản lý sẽ xem các bình luận tương ứng với sản phẩm và có quyền xóa bình luận.

Quản lý đơn hàng: người dùng có thể kiểm tra lịch sử mua hàng, thông tin, trạng thái của đơn hàng, người quản trị có thể quản lý tất cả các đơn hàng của hệ thống như: xem các đơn hàng, tạo và in hóa đơn.

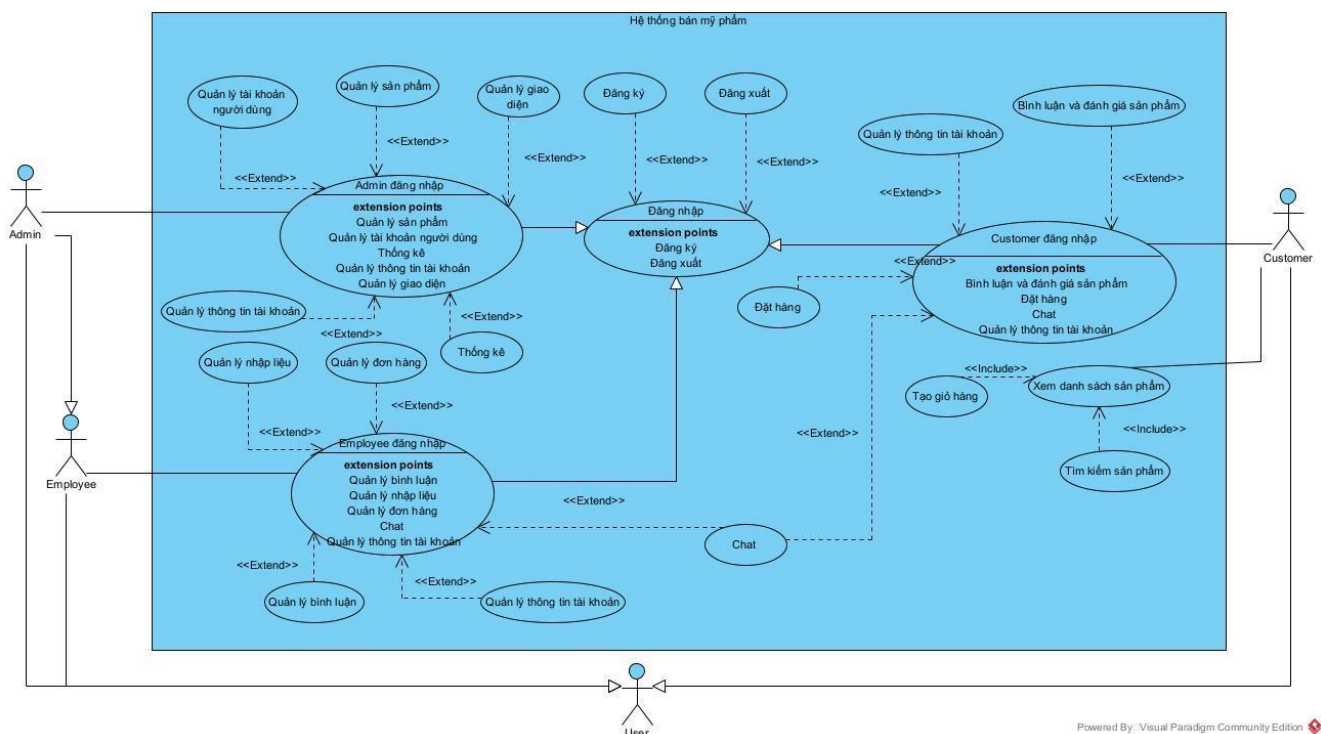
Quản lý sản phẩm: Người quản trị có thể quản lý các sản phẩm trong kho hàng như: thêm sản phẩm mới, chỉnh sửa sản phẩm, xóa các sản phẩm đã ngừng kinh doanh. Chức năng này chỉ áp dụng cho Admin(người quản trị hệ thống).

Quản lý người dùng: Người quản trị hệ thống có thể quản lý các tài khoản khách hàng như: thêm tài khoản, xóa tài khoản vi phạm điều khoản hệ thống, phân quyền tài khoản, ... Chức năng này chỉ áp dụng cho Admin(người quản trị hệ thống).

### 3.1.4 Xây dựng biểu đồ ca sử dụng (usecase)

#### 3.1.4.1 Xây dựng biểu đồ Usecase

Usecase mức tổng quát của hệ thống



Hình 3.1 Biểu đồ Usecase tổng quát

#### 3.1.4.2 Các kịch bản ứng với ứng dụng chung cho User

a. Kịch bản của usecase “Đăng ký”

**Bảng 3.2: Kịch bản của usecase “Đăng ký”**

Tên Usecase	Đăng ký
Tác nhân chính	User
Tiền điều kiện	User truy cập vào hệ thống trên website
Đảm bảo tối thiểu	Hệ thống cho phép người dùng đăng ký lại
Đảm bảo thành công	User đăng ký thành công nếu chưa có tài khoản tồn tại trong hệ thống
Kích hoạt	Người dùng chọn chức năng “Đăng ký”
<b>Chuỗi sự kiện chính:</b> 1. Hệ thống hiển thị giao diện đăng ký tài khoản mới gồm các ô email, mật khẩu, nhập lại mật khẩu và nút Đăng ký. 2. User điền đầy đủ thông tin vào giao diện đăng ký sau đó nhấn “Đăng ký” 3. Hệ thống kiểm tra, lưu thông tin vào cơ sở dữ liệu, thông báo thành công và chuyển đến trang đăng nhập của ứng dụng.	
<b>Ngoại lệ:</b> 2.1 User không nhập đủ thông tin yêu cầu và nhấn nút Đăng ký 2.1.a Hệ thống thông báo yêu cầu User nhập đủ thông tin. 2.1.b User nhập lại thông tin đầy đủ và tiến hành các bước tiếp theo 3.1 Hệ thống thông báo “Tên đăng nhập đã tồn tại trong hệ thống” và yêu cầu người dùng nhập lại thông tin	

b. Kịch bản của usecase “Đăng nhập”

**Bảng 3.3: Kịch bản của usecase “Đăng nhập”**

Tên Usecase	Đăng nhập
Tác nhân chính	User
Tiền điều kiện	User đã đăng nhập vào hệ thống trên website

Đảm bảo tối thiểu	Hệ thống cho phép User quay lui lại
Đảm bảo thành công	Hệ thống cho phép User đăng nhập thành công
Kích hoạt	User chọn chức năng “Đăng nhập”
<b>Chuỗi sự kiện chính:</b>  1. Hệ thống hiển thị giao diện đăng nhập vào hệ thống bao gồm: một ô nhập email, một ô nhập password và một nút đăng nhập. 2. User tiến hành đăng nhập vào hệ thống. User nhập đầy đủ thông tin email và password sau đó nhấn nút “Đăng Nhập”. 3. Hệ thống kiểm tra username và password, thông báo đăng nhập thành công và chuyển đến trang chủ của ứng dụng	
<b>Ngoại lệ</b>  2.1 User nhập thiếu thông tin đăng nhập và nhấn nút Đăng nhập. 2.1.a Hệ thống thông báo “E-mail/mật khẩu là bắt buộc!” và không cho đăng nhập. 2.1.b Người dùng nhập đầy đủ thông tin và tiến hành các bước tiếp theo. 2.2 User nhập sai trường email/password và nhấn nút đăng nhập. 2.2.a Hệ thống thông báo “Thất bại” và yêu cầu nhập lại.	

## c. Kịch bản của usecase “Quản lý thông tin tài khoản”

**Bảng 3.4: Kịch bản của usecase “Quản lý thông tin tài khoản”**

Tên Usecase	Quản lý thông tin tài khoản
Tác nhân chính	User
Tiền điều kiện	User đã đăng nhập vào hệ thống trên website
Đảm bảo tối thiểu	Hệ thống cho phép User quay lui lại
Đảm bảo thành công	Hệ thống cho phép User xem và cập nhật thông tin tài khoản
Kích hoạt	User chọn chức năng “Tài khoản”
<b>Chuỗi sự kiện chính:</b> 1. Hệ thống hiển thị form gồm có ô nhập họ và tên, số điện, thoại, số CMT, ngày sinh, địa chỉ, mật khẩu mới và nút Cập nhật 2. User nhập đầy đủ các thông tin bắt buộc và nhấn “Cập nhật” 3. Hệ thống kiểm tra và lưu thông tin vào cơ sở dữ liệu, thông báo cập nhật thành công.	
<b>Ngoại lệ</b> 2.1 User chưa nhập đủ trường thông tin bắt buộc 2.1.a Hệ thống thông báo cần nhập đủ trường thông tin bắt buộc và yêu cầu nhập lại. 2.1.b User nhập lại và tiếp tục các bước tiếp theo. 3.1 Hệ thống báo lỗi khi cập nhật thông tin.	

## d. Kịch bản cho usecase “Xem chi tiết sản phẩm”

## e. Kịch bản của usecase “Xem danh sách sản phẩm”

## f. Kịch bản cho usecase “Tìm kiếm sản phẩm”

## g. Kịch bản cho usecase “ Đăng xuất”

**3.1.4.3 Các kịch bản ứng với ứng dụng cho quản trị viên**

## a. Kịch bản của usecase “Quản lý tài khoản người dùng”



**Bảng 3.9: Kịch bản của usecase “Quản lý tài khoản người dùng”**

Tên Usecase	Quản lý tài khoản người dùng
Tác nhân chính	Admin
Tiền điều kiện	Admin đã đăng nhập vào hệ thống trên website
Đảm bảo tối thiểu	Hệ thống cho phép Admin quay lui lại
Đảm bảo thành công	Cho phép Admin xem và thực hiện được các chức năng thêm, sửa, xóa tài khoản người dùng
Kích hoạt	Admin chọn chức năng “Quản lý tài khoản”
<b>Chuỗi sự kiện chính:</b>  1. Hệ thống hiển thị bảng danh sách tất cả các tài khoản có trong hệ thống, có nút thêm mới, sửa xóa tài khoản.	
<b>Ngoại lệ</b> 4.1 Hệ thống lỗi, vui lòng thử lại sau.	

- b. Kịch bản cho usecase “Thêm tài khoản người dùng”
- c. Kịch bản cho usecase “Sửa tài khoản người dùng”
- d. Kịch bản cho usecase “Xóa tài khoản người dùng”
- e. Kịch bản của usecase “Tổng kê”
- f. Kịch bản của usecase “Quản lý sản phẩm”

**3.1.4.4 Các kịch bản ứng với ứng dụng cho Employee**

- a. Kịch bản của usecase “Quản lý đơn hàng”

**Bảng 3.18: Kịch bản của usecase “Quản lý đơn hàng”**

Tên Usecase	Quản lý đơn hàng
Tác nhân chính	Employee
Tiền điều kiện	Employee đã đăng nhập vào hệ thống trên website
Đảm bảo tối thiểu	Không thay đổi dữ liệu khi xảy ra lỗi

Đảm bảo thành công	Employee xử lý đơn hàng thay đổi dữ liệu thành công.
Kích hoạt	Employee chọn chức năng “Danh sách đơn hàng” và chọn xem chi tiết các đơn để xử lý đơn hàng.
<b>Chuỗi sự kiện chính:</b> <ol style="list-style-type: none"> <li>1. Nhân viên click chọn “Đơn hàng”.</li> <li>2. Hệ thống hiển thị danh sách các đơn hàng và nhân viên chọn xem chi tiết.</li> <li>3. Nhân viên xem thông tin khách hàng đã đặt và chi tiết các sản phẩm kèm theo tổng tiền thanh toán. Phía dưới có các ô lựa chọn trạng thái của đơn hàng: Chưa xử lý, đang vận chuyển, giao thành công, đã hoàn trả.</li> <li>4. Nhân viên chọn trạng thái giao hàng và ấn nút xác nhận.</li> <li>5. Hệ thống báo xử lý đơn hàng thành công.</li> </ol>	
<b>Ngoại lệ</b> <ol style="list-style-type: none"> <li>5.1 Nếu nhân viên chọn trạng thái “Đã hoàn trả”. <ol style="list-style-type: none"> <li>5.1.a Hệ thống sẽ cập nhật lại số lượng sản phẩm của cửa hàng.</li> </ol> </li> </ol>	

b. Kịch bản của usecase “Quản lý bình luận”

### 3.1.4.5 Các kịch bản ứng với ứng dụng cho Customer

a. Kịch bản của usecase “Tạo giỏ hàng”

**Bảng 3.20: Kịch bản của usecase “Tạo giỏ hàng”**

Tên Usecase	Tạo giỏ hàng
Tác nhân chính	Khách hàng
Tiền điều kiện	Khách hàng đã truy cập vào hệ thống trên website
Đảm bảo tối thiểu	Khách hàng xem được thông tin sản phẩm
Đảm bảo thành công	Hệ thống hiển thị thông tin chi tiết các sản phẩm khách hàng chọn.

Kích hoạt	Khách hàng click vào hình ảnh sản phẩm.
<b>Chuỗi sự kiện chính:</b> <ol style="list-style-type: none"> <li>1. Khách hàng click vào ảnh của sản phẩm muốn mua</li> <li>2. Hệ thống hiển thị thông tin chi tiết của sản phẩm.</li> <li>3. Khách hàng chọn thêm vào giỏ hàng</li> <li>4. Hệ thống thêm sản phẩm khách hàng đã chọn vào giỏ hàng và hiển thị thông tin giỏ hàng.</li> </ol>	
<b>Ngoại lệ</b> <ol style="list-style-type: none"> <li>4.1 Khách hàng muốn xem các mặt hàng đã để trong giỏ hàng có thể click vào giỏ hàng ở góc phải màn hình.</li> <li>4.2 Khách hàng click chọn thêm vào giỏ hàng nhiều lần để tăng số lượng sản phẩm muốn mua             <ol style="list-style-type: none"> <li>4.2.a Khách hàng click vào dấu “+” để tăng số lượng sản phẩm. Nếu số lượng sản phẩm cần mua lớn hơn số lượng sản phẩm có sẵn thì hệ thống mặc định lấy số lượng sản phẩm có sẵn.</li> <li>4.2.b Khách hàng click dấu “x” để xóa sản phẩm khỏi giỏ hàng</li> <li>4.2.c Khách hàng click dấu “-“ để giảm số lượng sản phẩm. Nếu số lượng sản phẩm nhỏ hơn 1 hệ thống sẽ xóa sản phẩm khỏi giỏ hàng.</li> </ol> </li> </ol>	

b. Kịch bản của usecase “Đặt hàng”

c. Kịch bản của usecase “Bình luận và đánh giá sản phẩm”

### 3.1.5 Xây dựng sơ đồ thực thể liên kết

Nội dung phần này trình bày sơ đồ thực thể liên kết và mô tả các thuộc tính của các thực thể.

## 3.2 Thiết kế hệ thống

Phần này tập trung vào thiết kế chi tiết hệ thống. Bao gồm: thiết kế mô hình tuần tự cho hệ thống, xác định thuộc tính và gán phương thức cho các lớp, xây dựng biểu đồ lớp thiết kế. Chi tiết hóa hệ thống, người lập trình có thể hình dung chi tiết

từng chức năng của hệ thống, giúp cho việc cài đặt hệ thống thuận tiện và chính xác hơn.

### **3.2.2 Xác định thuộc tính và gán phương thức cho các lớp**

#### **3.2.2.1 Lớp User**

- Định nghĩa: Chứa các thuộc tính về thông tin người dùng trong hệ thống
- Thuộc tính:
  - + email: email đăng nhập của người dùng
  - + password: mật khẩu đăng nhập
  - + role: chức năng của người dùng trong hệ thống
  - + tel: số điện thoại của người dùng
  - + address: địa chỉ của người dùng
  - + bankAccount: tài khoản ngân hàng của người dùng (nếu có)
  - + name: tên người dùng
- Phương thức:
  - + login(): kiểm tra trước khi đăng nhập vào hệ thống
  - + signup(): tạo 1 tài khoản mới
  - + create(): tạo 1 tài khoản mới
  - + getUsers(): lấy ra tất cả các người dùng hiện có
  - + updateUser(): sửa thông tin tài khoản
  - + findById(): tìm người dùng theo id
  - + delById(): xóa người dùng theo id
  - + createPayment(): tạo thông tin thanh toán cho người dùng

#### **3.2.2.2 Lớp Location**

- Định nghĩa: Chứa các thuộc tính là thông tin về địa chỉ của người dùng
- Thuộc tính:
  - + name: tên địa chỉ
  - + type: vị trí nào
  - + description: mô tả vị trí cụ thể
- Phương thức:

- + getLocation(): lấy ra danh sách các địa chỉ

- + update(): sửa thông tin địa chỉ

- + replace(): xóa thông tin địa chỉ

### **3.2.2.3 Lớp BankAccount**

- Định nghĩa: Chứa các thuộc tính là thông tin về tài khoản ngân hàng của người dùng

- Thuộc tính:

- + createAt: ngày tạo tài khoản

- + bankAccount: số tài khoản của người dùng

- + bankHolder: tên tài khoản ngân hàng của người dùng

- + bankBranch: địa chỉ ngân hàng

- Phương thức:

- + create(): tạo tài khoản

- + get(): lấy ra danh sách các tài khoản ngân hàng

- + findById(): tìm kiếm theo id

- + updateById(): sửa thông tin tài khoản ngân hàng

- + delete(): xóa thông tin tài khoản

### **3.2.2.4 Organization**

- Định nghĩa: chứa thông tin về các ngân hàng liên kết với hệ thống

- Thuộc tính:

- + createAt(): thời gian khởi tạo

- + email: email của ngân hàng

- + name: tên ngân hàng

- + code: mã ngân hàng

- Phương thức:

+ getOrganization(): lấy ra danh sách các ngân hàng

+ findById(): tìm kiếm ngân hàng theo id

+ deleteById(): xóa ngân hàng theo id

### **3.2.2.5 Lớp Banner**

- Định nghĩa: Chứa các thuộc tính là thông tin về slideshow sẽ xuất hiện trên giao diện của hệ thống.

- Thuộc tính:

+ image: đường dẫn của ảnh là slide

+ description: mô tả nội dung slide

+ status: trạng thái hoạt động của slide “active” hoặc “inactive”

- Phương thức:

+ getBanner(): lấy tất cả các banner

+ createBanner(): tạo mới một banner

+ deleteBanner(): xóa một banner

### **3.2.2.6 Lớp Product**

- Định nghĩa: chứa các thuộc tính là thông tin về sản phẩm trong hệ thống

- Thuộc tính:

+ name: tên sản phẩm

+ price: giá của sản phẩm

+ inStock: số lượng còn lại trong kho

+ image: hình ảnh minh họa

+ description: mô tả về sản phẩm

- Phương thức:

- + getProduct(): lấy tất cả các sản phẩm
- + createProduct(): tạo mới một sản phẩm
- + update(): sửa thông tin của một sản phẩm
- + delete(): xóa sản phẩm khỏi danh sách

#### **3.2.2.7 Lớp Category**

- Định nghĩa: chứa các thuộc tính là thông tin về danh mục sản phẩm
- Thuộc tính:
  - + name: tên danh mục sản phẩm
  - +barcode: mã danh mục sản phẩm
- Phương thức:
  - + getCategory(): lấy ra danh sách các danh mục sản phẩm
  - + create(): tạo mới một danh mục sản phẩm
  - + update(): sửa thông tin của một danh mục sản phẩm
  - + delete(): xóa danh mục sản phẩm ra khỏi danh sách

#### **3.2.2.8 Lớp Order**

- Định nghĩa: chứa thuộc tính là thông tin về các đơn đặt hàng trong hệ thống
- Thuộc tính:
  - + products: đơn hàng có những sản phẩm này
  - + customer: đơn hàng được đặt bởi khách hàng này
  - + tel: số điện thoại khách hàng
  - + totalPrice: tổng tiền
  - + paymentMethod: phương thức thanh toán cho đơn hàng
  - + address: địa chỉ nhận hàng

- + paid: xác định đơn hàng đã được thanh toán hay chưa
- + status: tình trạng của đơn hàng
- + description: mô tả lý do hủy đơn hàng (nếu khách hàng hủy đơn)
- Phương thức:
  - + getOrder(): lấy ra danh sách các đơn hàng
  - + create(): tạo mới một đơn hàng
  - + update(): sửa thông tin của một đơn hàng
  - + delete(): xóa đơn hàng ra khỏi danh sách

#### **3.2.2.9 Lớp Comment**

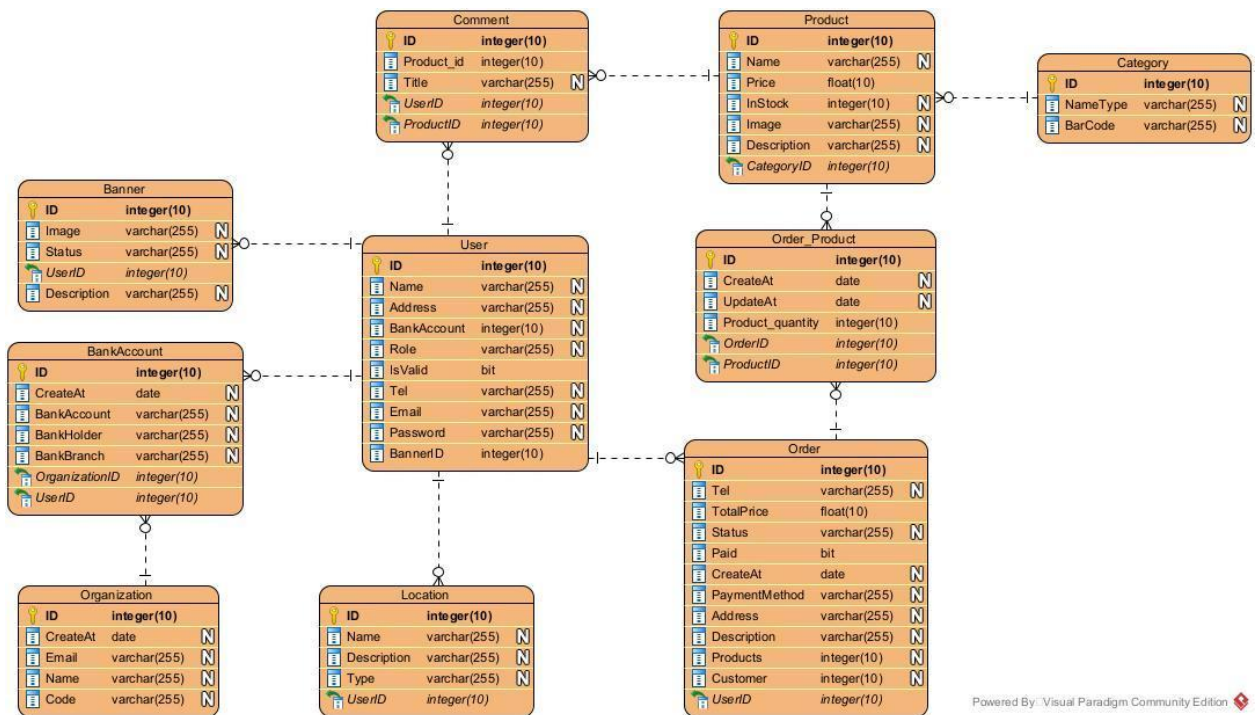
- Định nghĩa: chứa thuộc tính của bình luận về sản phẩm của khách hàng
- Thuộc tính:
  - + product\_id: mã sản phẩm mà khách hàng bình luận
  - + tittle: nội dung bình luận
  - + createAt: ngày tạo bình luận
- Phương thức:
  - + create(): tạo một bình luận
  - + delete(): xóa một bình luận

#### **3.2.3 Thiết kế cơ sở dữ liệu**

Dựa vào biểu đồ phân tích đã có ở phần trước có thể xác định được các đối tượng, thuộc tính và kiểu của thuộc tính được lưu trữ trong cơ sở dữ liệu. Từ đó xây dựng lên lược đồ cơ sở dữ liệu như hình vẽ bên dưới:

Lược đồ cơ sở dữ liệu của hệ thống





Hình 3.36 Lược đồ cơ sở dữ liệu của hệ thống

### 3.3 Kết luận chương

Trong chương này, đã thực hiện xem xét các bước phân tích và thiết kế hệ thống website bán mỹ phẩm. Có các UML để hiểu rõ các mô hình thông tin tuần tự của hệ thống. Mô tả được kiến trúc tổng quan của hệ thống, xây dựng lược đồ cơ sở dữ liệu quan hệ dựa trên mô hình lớp.

## CHƯƠNG IV. CÀI ĐẶT VÀ KIỂM THỬ

### 4.1 Môi trường triển khai ứng dụng

Ứng dụng được triển khai kiểm thử trên localhost với:

- Ngôn ngữ sử dụng là Javascript
- Sử dụng hệ quản trị cơ sở phi dữ liệu MongoDB
- Chạy trên hệ điều hành Window
- Công cụ hỗ trợ lập trình: Sublime Text
- Sử dụng Visual Paradigm là công cụ hỗ trợ phân tích và thiết kế hệ thống.

### 4.2 Giao diện ứng dụng

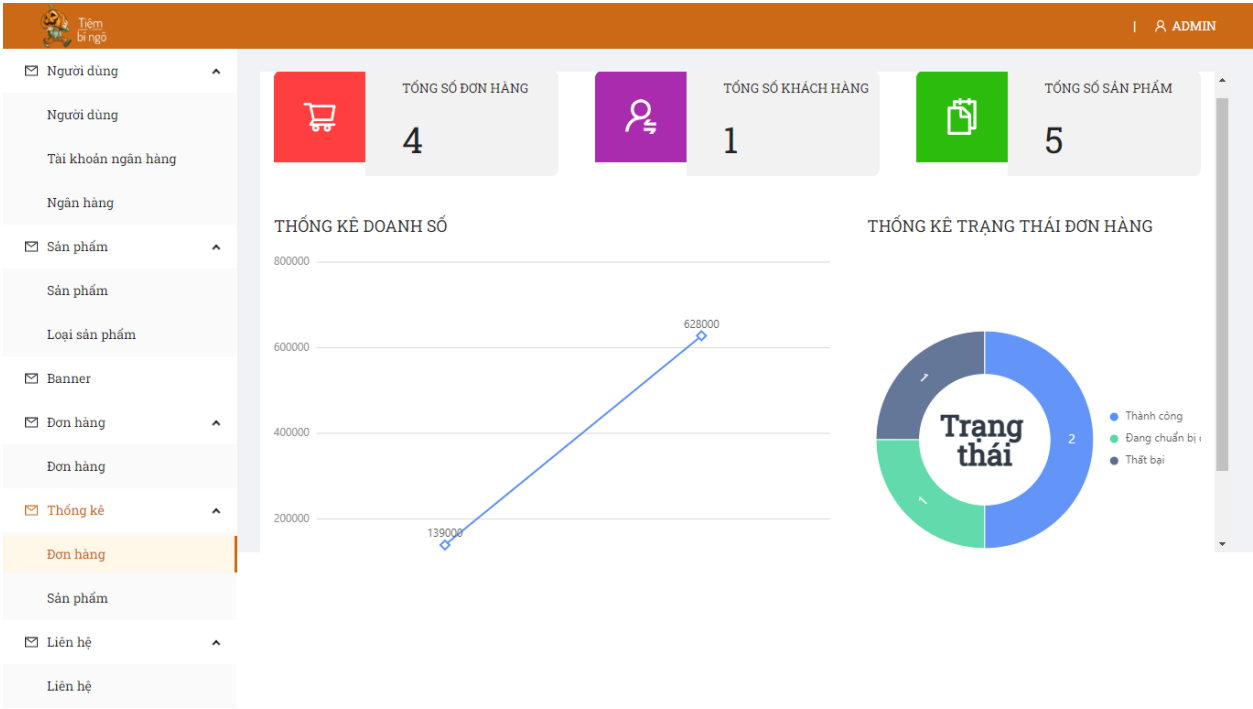
#### 4.2.1 Giao diện một số trang chính

- Giao diện trang chủ



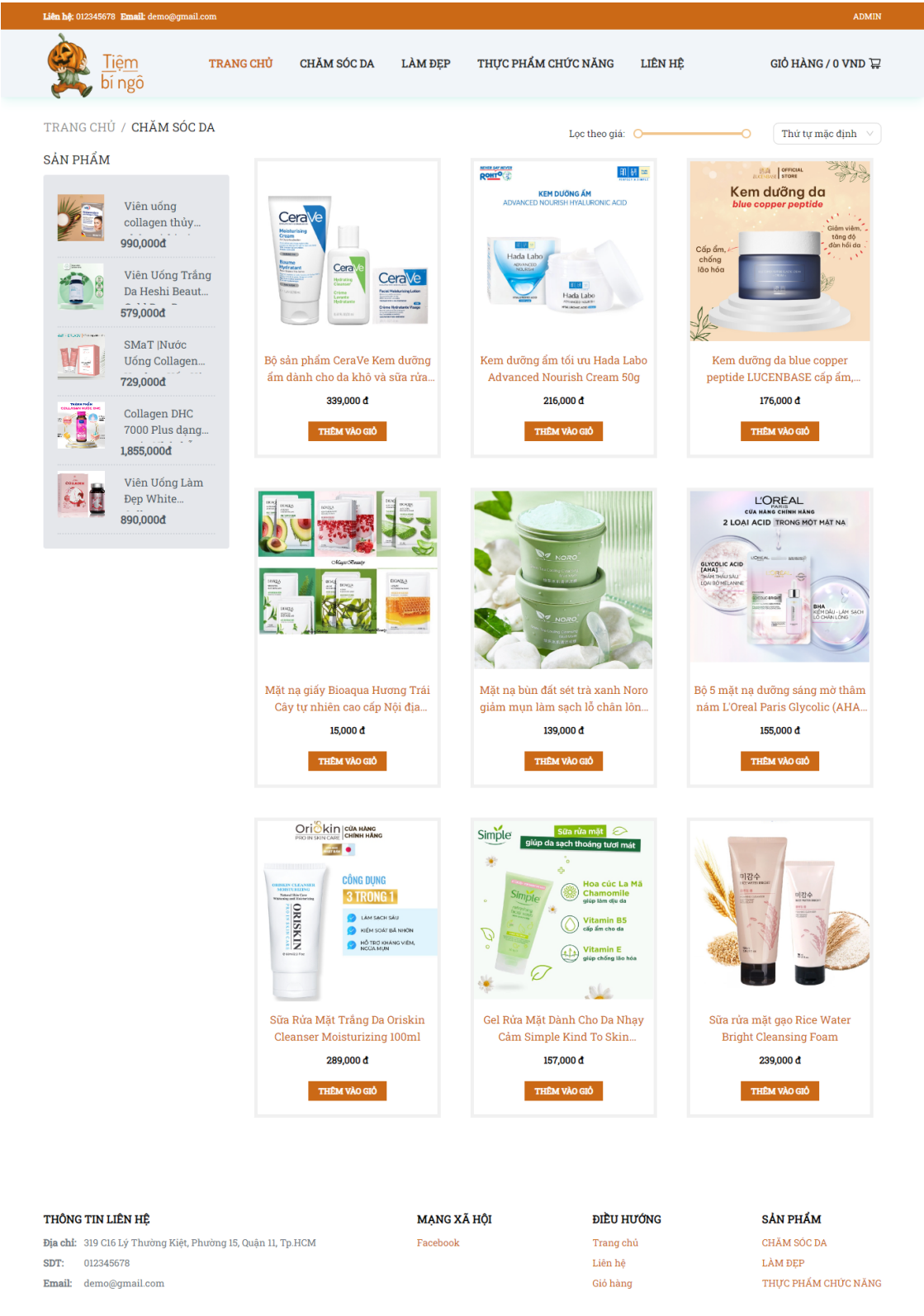
Hình 4.2 Giao diện trang chủ

- Giao diện trang quản trị



Hình 4.3 Giao diện trang quản trị

- Giao diện trang sản phẩm



Hình 4.4 Giao diện trang sản phẩm

4.2.2 Giao diện một số chức năng

## - Giao diện chức năng Đăng ký, Đăng nhập

Đăng ký

The screenshot shows a login and registration form on a website with an autumn-themed background featuring orange and brown tones with falling leaves and tree branches. The form is titled "Đăng nhập" (Login) and includes the following elements:

- \* E-mail:** A text input field containing "admin@gmail.com".
- \* Mật khẩu:** A password input field with masked characters "\*\*\*\*\*" and an eye icon to toggle visibility.
- Ghi nhớ:** A checked checkbox labeled "Ghi nhớ" (Remember me).
- Đăng nhập:** An orange button to submit the login form.
- Đăng ký:** A link labeled "Đăng ký" (Register) for users who do not have an account.

### THÔNG TIN LIÊN HỆ

**Địa chỉ:** 319 C16 Lý Thường Kiệt, Phường 15, Quận 11, Tp.HCM  
**SDT:** 012345678  
**Email:** demo@gmail.com

### MẠNG XÃ HỘI

[Facebook](#)

### ĐIỀU HƯỚNG

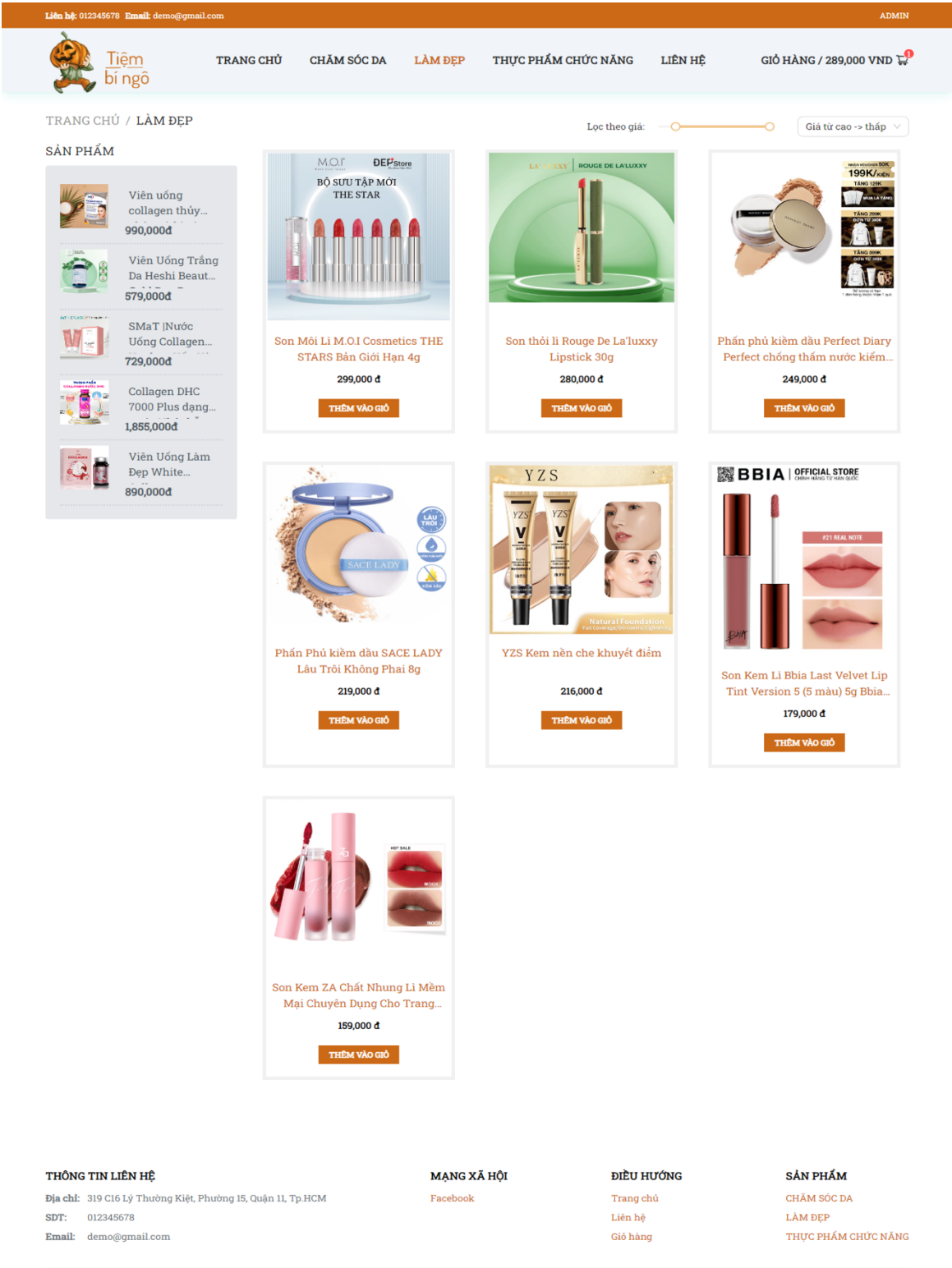
[Trang chủ](#)  
[Liên hệ](#)  
[Giỏ hàng](#)

### SẢN PHẨM

[CHĂM SÓC DA](#)  
[LÀM ĐẸP](#)  
[THỰC PHẨM CHỨC NĂNG](#)

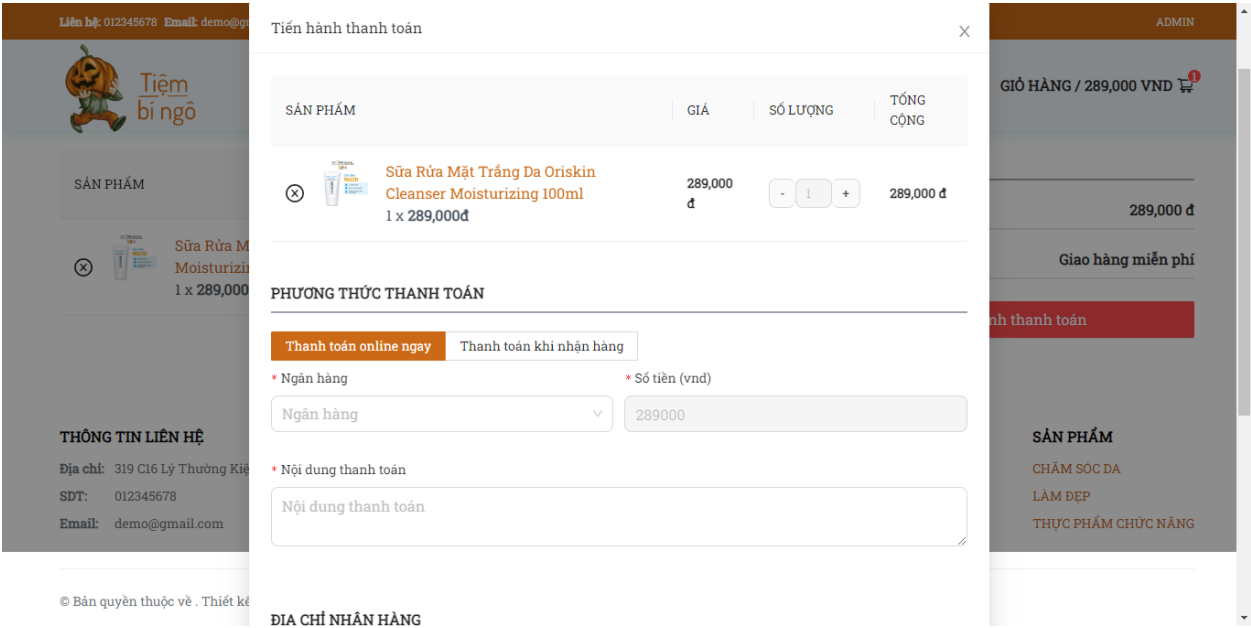
**Hình 4.5** Giao diện chức năng đăng nhập, đăng ký

Giao diện chức năng tìm kiếm, lọc, sắp xếp theo giá sản phẩm



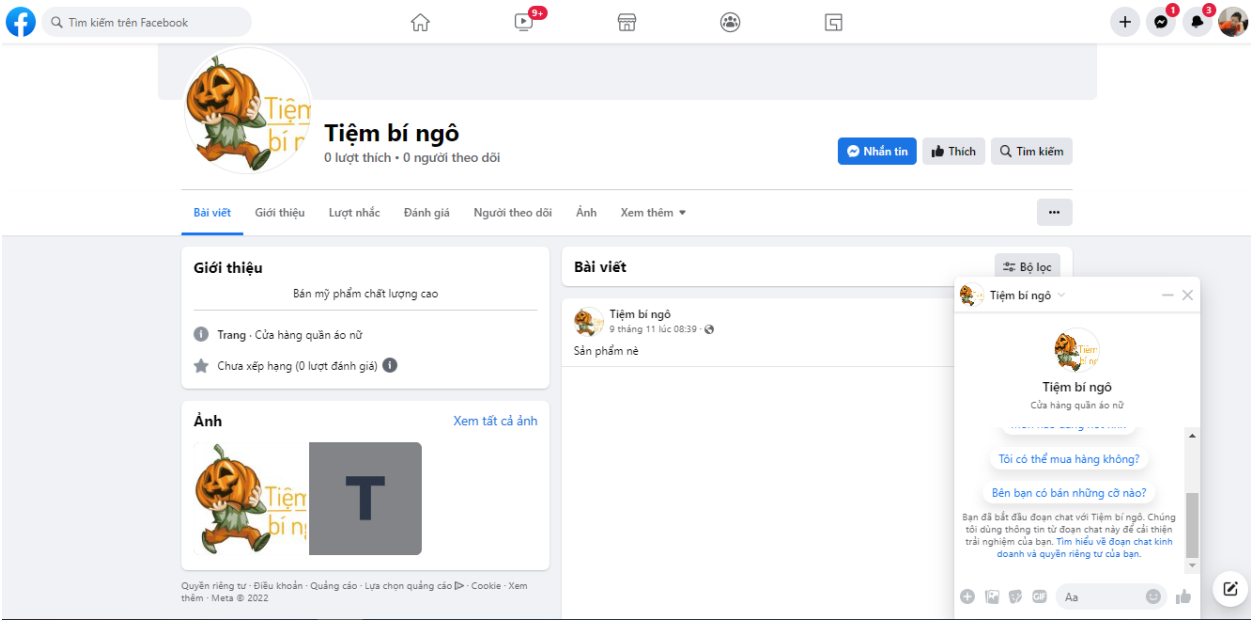
Hình 4.6 Giao diện chức năng tìm kiếm, lọc, sắp xếp theo giá sản phẩm

Giao diện chức năng mua hàng và thanh toán



Hình 4.7 Giao diện chức năng mua hàng và thanh toán

- Giao diện chat với hệ thống qua facebook



Hình 4.8 Giao diện chat với hệ thống qua facebook

- Giao diện admin quản lý người dùng

The screenshot shows an admin dashboard with a sidebar menu on the left. The main content area displays a modal window titled "Thông tin chi tiết người dùng" (User Details). The form contains the following fields:

- \* E-mail:** Text input with value "admin@gmail.com".
- \* Loại người dùng:** Dropdown menu with value "Admin".
- \* Quyền:** Text input with value "ADMIN".
- Họ và tên:** Text input with value "ADMIN".
- Tỉnh/ Thành phố:** Dropdown menu.
- Quận/huyện:** Dropdown menu.
- Xã/phường:** Dropdown menu.
- Submit:** Orange button at the bottom.

**Hình 4.9 Giao diện admin quản lý người dùng**

- Giao diện admin quản lý banner

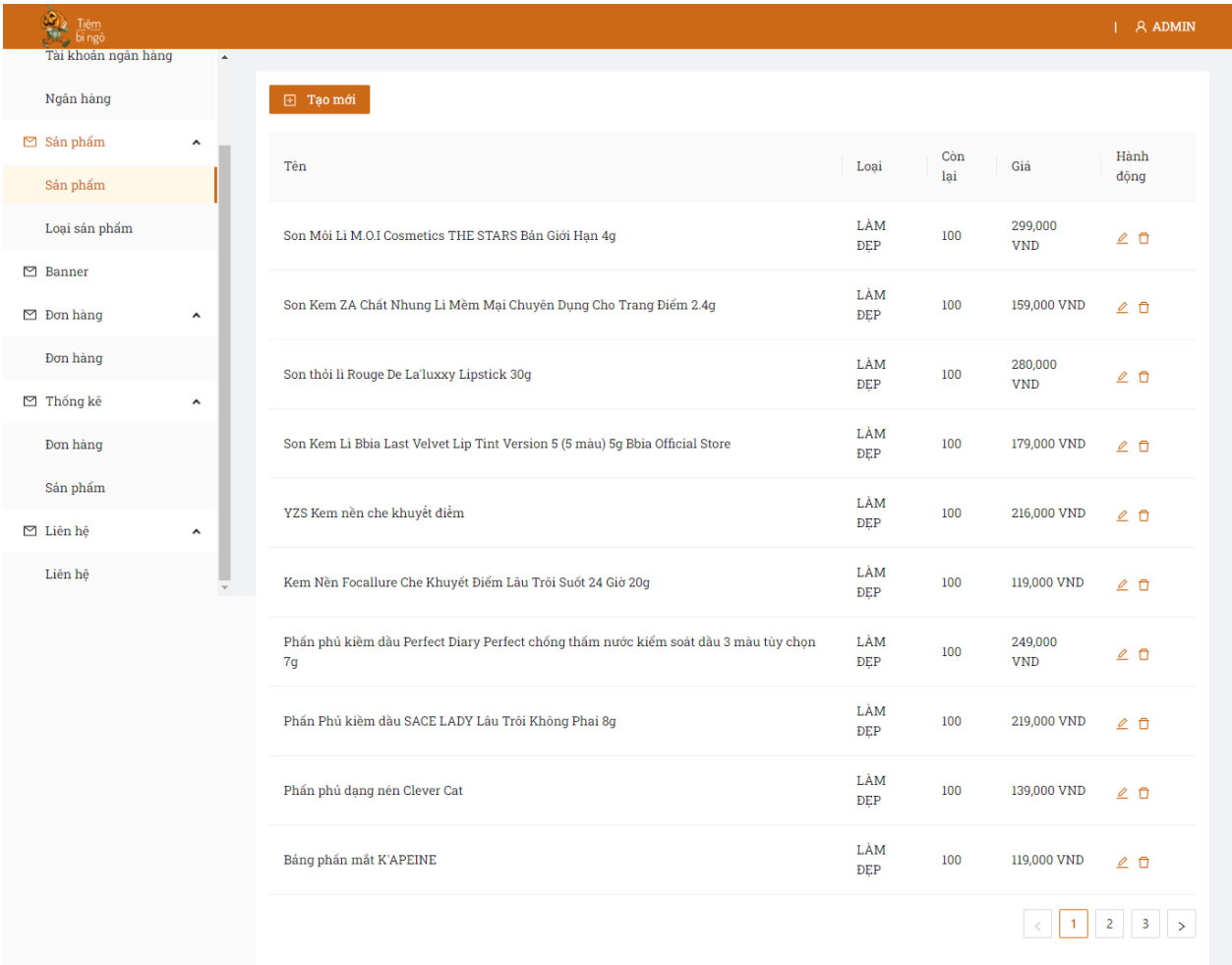
The screenshot shows the same admin dashboard, but the modal window is titled "Thông tin chi tiết banner" (Banner Details). The form contains the following fields:

- \* Hình ảnh:** Image upload field showing a placeholder image.
- \* Mô tả:** Text input with value "CHẤT LƯỢNG HÀNG ĐẦU".
- Trạng thái:** Radio buttons for "Hoạt động" (Active) and "Không hoạt động" (Inactive).
- Submit:** Orange button at the bottom.

**Hình 4.10 Giao diện admin quản lý banner**

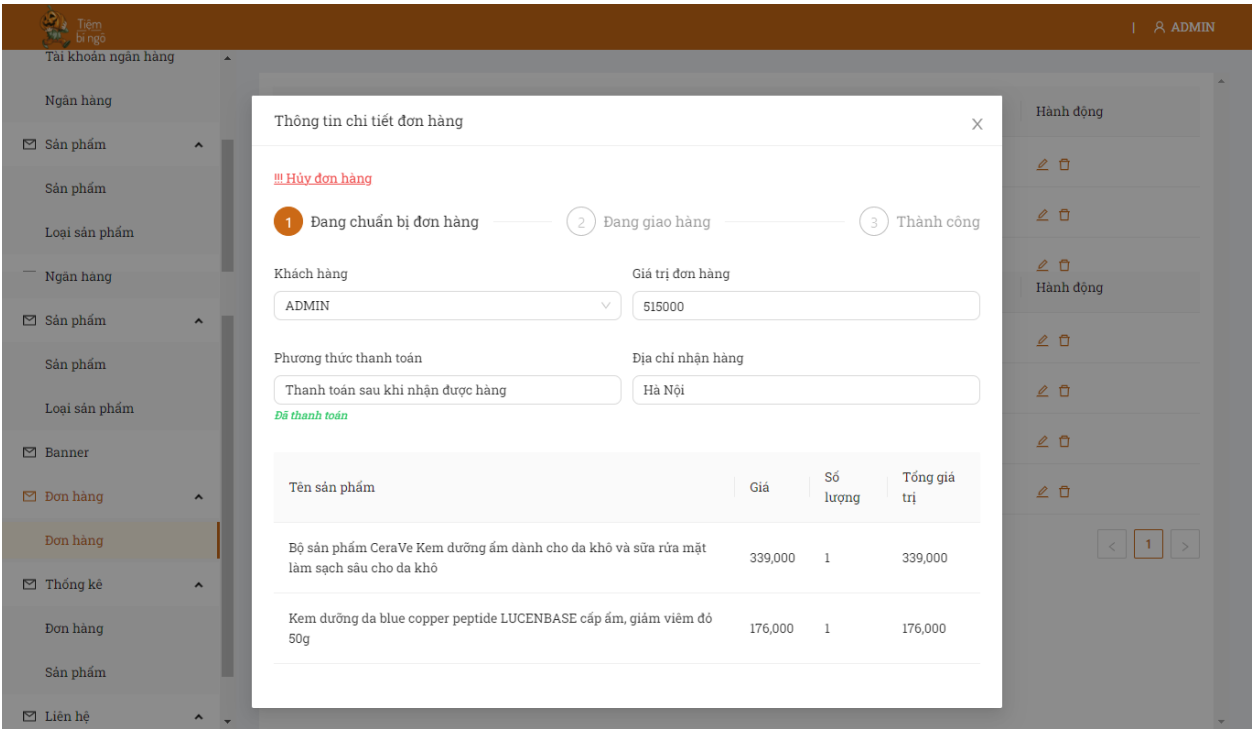


- Giao diện quản lý sản phẩm



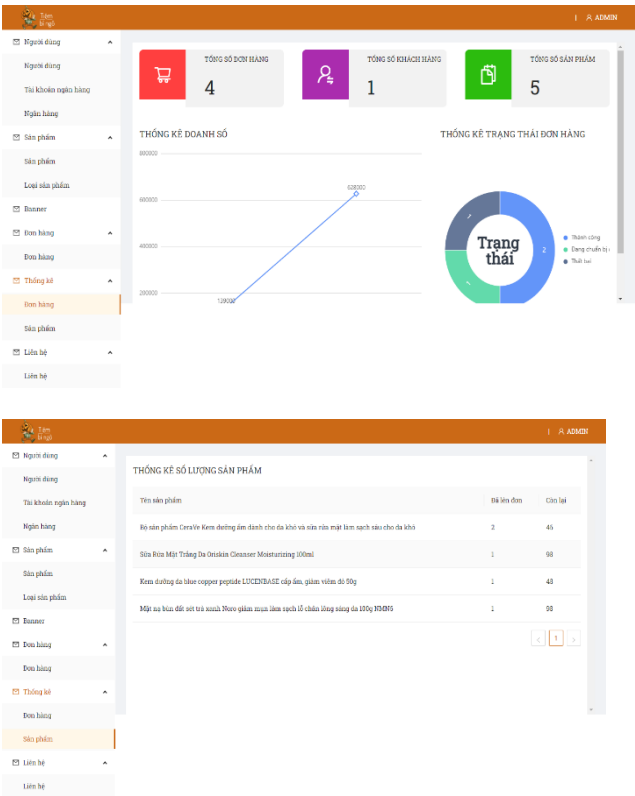
Hình 4.11 Giao diện quản lý sản phẩm

- Giao diện quản lý đơn hàng



Hình 4.12 Giao diện quản lý đơn hàng

- Giao diện thống kê đơn hàng, sản phẩm



Hình 4.13 Giao diện thống kê đơn hàng, sản phẩm

## KẾT LUẬN

### ● Kết quả thu được

Sau một thời gian tìm hiểu và khảo sát qua các website bán mỹ phẩm và một số tài liệu trực tuyến xây dựng website bán hàng bằng NodeJS, từ đó đề tài đã hoàn thiện được “xây dựng hệ thống nbans mỹ phẩm” với những kết quả đạt được như sau:

- Về chức năng: hệ thống đã hoàn thiện được hầu hết các chức năng cơ bản của website bán mỹ phẩm như là: người dùng có thể tạo tài khoản, đăng nhập và sử dụng các chức năng của hệ thống, xem sản phẩm, đọc tin tức, thêm sản phẩm vào giỏ hàng, giao nhận sản phẩm... Người quản trị có thể nắm bắt được tình trạng kho hàng, tình trạng đơn hàng, doanh thu của cửa hàng.

- Về giao diện: hệ thống phát triển giao diện tương đối đẹp mắt và có điểm nhận diện riêng bằng mắt với bố cục gọn gàng. Bên cạnh đó giao diện còn cung cấp cho người dùng đầy đủ thông tin, chức năng tiện dụng và hệ thống xử lý hiệu ứng đẹp mắt.

- Về hiệu năng: hệ thống có hiệu năng tương đối ổn và đạt được yêu cầu đặt ra từ đầu.

- Về kiến thức và kỹ năng: thông qua đề tài này chúng em đã mở mang được rất nhiều kiến thức về ngôn ngữ lập trình Javascript, đặc biệt là Framework NodeJs, kiến thức về lập trình giao diện: HTML, CSS, ReactJS để xử lý giao diện và nội dung. Bên cạnh đó kỹ năng tìm kiếm thông tin, tìm kiếm và xử lý vấn đề cũng được cải thiện rất nhiều.

### ● Kết quả chưa đạt được

Song song với các kết quả đạt được bên trên thì hệ thống cũng còn nhiều thiếu sót về nhiều mặt như là:

- Về chức năng: hiện tại chỉ gồm các chức năng cơ bản của trang thương mại điện tử, chưa có các chức năng nâng cao như là: thông báo, theo dõi sản phẩm,...

- Về giao diện: tuy thân thiện với người sử dụng nhưng kéo theo đó là hệ thống phải dành một phần hiệu năng tương đối mỗi lần tải giao diện.

- Về hiệu năng: hiệu năng tuy phù hợp ở thời điểm hiện tại nhưng trong tương lai, quy mô dự án thay đổi chắc chắn hiệu năng sẽ không đủ đáp ứng được số lượng người truy cập lớn.

- **Định hướng phát triển**

Trong tương lai, định hướng phát triển thêm của đề tài sẽ là tiếp tục phát triển các chức năng đang ở mức sơ khai. Bên cạnh đó đề tài sẽ tiếp tục phát triển thêm các chức năng mới như là: hỗ trợ đa ngôn ngữ, theo dõi sản phẩm, thông báo thời gian thực, hệ thống nhắn tin và trả lời tự động, hệ thống phân phối và giao hàng theo khu vực để đảm bảo tiết kiệm thời gian và tri phí giao hàng (Logistics)...

Việc cải thiện hiệu năng cũng vô cùng quan trọng trong tương lai khi mà lượng người dùng và dữ liệu ngày một nhiều lên. Giải pháp trong tương lai là tối ưu lại toàn bộ cấu trúc, thiết kế lại toàn bộ cơ sở dữ liệu, giao diện loại bỏ bớt hiệu ứng không cần thiết...

## TÀI LIỆU THAM KHẢO

### Danh mục tài liệu tham khảo:

[I] Trần Đình Quế, Phân tích và thiết kế Hệ Thống Thông Tin, Học Viện Công Nghệ Bưu Chính Viễn Thông, 2015.

.....