**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

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

****

BÁO CÁO BTL THUỘC HỌC PHẦN:

ĐỒ ÁN CHUYÊN NGÀNH

**XÂY DỰNG WEBSITE QUẢNG CÁO VÀ BÁN HÀNG CHO TH TRUE MILK**

|  |  |
| --- | --- |
| GVHD: | Gv. Phạm Văn Hiệp |
| Nhóm - Lớp: | 20 – 20241IT6122001 |
| Thành viên: | 2021600262 – Bùi Quang Thắng |
|  | 2021604798 – Hoàng Kim Tấn  2021607906 – Nguyễn Thái Sơn |
|  | 2021607252 – Nguyễn Thị Sen |
|  |  |
|  |  |

Hà nội, Năm 2023

# LỜI NÓI ĐẦU

Hiện nay trên thế giới thương mại điện tử đang phát triển rất mạnh mẽ. Kỹ thuật số giúp chúng ta tiết kiệm đáng kể các chi phí nhờ chi phí vận chuyển trung gian và đặc biệt là giúp tiết kiệm thời gian để con người đầu tư vào các hoạt động khác. Khi mà Internet trở nên thân quen và dần trở thành một công cụ không thể thiếu trong cuộc sống thì lợi ích của website đối với việc quảng bá sản phẩm và thương hiệu của một công ty thật là to lớn. Trang web trở thành một cửa ngõ để doanh nghiệp tiếp thị sản phẩm của mình đến khách hàng trên khắp nơi trên toàn thế giới, nó đóng một vai trò rất quan trọng trong công cuộc phát triển và đấu tranh trong lĩnh vực kinh doanh.Chính vì vậy nhóm em chọn đề tài về: **“Xây dựng Website quảng cáo và bán hàng cho TH True Milk”.**

Trong lời đầu tiên của báo cáo đồ án tốt nghiệp “Xây dựng Website quảng cáo và bán hàng cho TH True Milk” này, nhóm em muốn gửi những lời cám ơn và biết ơn chân thành nhất của mình tới tất cả những người đã hỗ trợ, giúp đỡ em về kiến thức và tinh thần trong quá trình thực hiện.

Nhóm em xin chân thành cám ơn Thầy Phạm Văn Hiệp, Giảng viên Khoa Công Nghệ Thông Tin, Trường Đại học Công nghiệp Hà Nội, người đã trực tiếp hướng dẫn, nhận xét, giúp đỡ nhóm em trong suốt quá trình thực hiện.

Nhóm em xin chân thành cảm ơn!

Hà Nội, ngày 15 tháng 10 năm 2023

MỤC LỤC

[LỜI NÓI ĐẦU i](#_Toc152685348)

[DANH MỤC HÌNH VẼ iv](#_Toc152685349)

[MỞ ĐẦU 1](#_Toc152685350)

[Chương 1. Giới thiệu tổng quan 4](#_Toc152685351)

[1.1 Giới thiệu chung về đề tài 4](#_Toc152685352)

[1.2 Nghiên cứu hiện trạng và giải quyết vấn đề 5](#_Toc152685353)

[*1.2.1* *Hiện trạng cơ cấu tổ chức* 5](#_Toc152685354)

[*1.2.2* *Mô tả hoạt động của các bộ phận* 6](#_Toc152685355)

[*1.2.3* *Xác định yêu cầu* 7](#_Toc152685356)

[1.3 Các giai đoạn phát triển dự án 8](#_Toc152685357)

[1.4 Vấn đề - giải pháp 9](#_Toc152685358)

[1.5 Giới thiệu công nghệ, ngôn ngữ xây dựng website 9](#_Toc152685359)

[*1.5.1* *ASP.NET và MVC* 9](#_Toc152685360)

[*1.5.2* *Javascript* 16](#_Toc152685361)

[*1.5.3* *Ajax* 18](#_Toc152685362)

[*1.5.4* *Bootstrap* 18](#_Toc152685363)

[*1.5.5* *Hệ cơ sở dữ liệu SQL server* 19](#_Toc152685364)

[*1.5.6* *Truy vấn Linq* 20](#_Toc152685365)

[Chương 2. Phân tích thiết kế hệ thống 26](#_Toc152685366)

[2.1 Mô hình hóa chức năng 26](#_Toc152685367)

[*2.1.1* *Biểu đồ use case* 26](#_Toc152685368)

[*2.1.2* *Mô tả chi tiết các use case* 30](#_Toc152685369)

[2.2 Mô hình hóa dữ liệu 45](#_Toc152685370)

[*2.2.1* *Các yêu cầu về dữ liệu* 70](#_Toc152685371)

[*2.2.2* *Biểu đồ thực thể liên kết mức logic* 72](#_Toc152685372)

[*2.2.3* *Mô hình cơ sở dữ liệu mức vật lý* 73](#_Toc152685373)

[2.3 Phân tích use case 45](#_Toc152685374)

[*2.3.1* *Biểu đồ các lớp Entity của hệ thống* 45](#_Toc152685375)

[*2.3.2* *Phân tích các use case* 46](#_Toc152685376)

[*2.3.3* *Biểu đồ các lớp phân tích* 68](#_Toc152685377)

[Chương 3. Cài đặt và triển khai 78](#_Toc152685378)

[3.1 Công cụ và môi trường cài đặt 78](#_Toc152685379)

[*3.1.1* *Visual Studio Code* 78](#_Toc152685380)

[*3.1.2* *Visual Studio 2019* 78](#_Toc152685381)

[*3.1.3* *Microsoft SQL server* 80](#_Toc152685382)

# DANH MỤC HÌNH VẼ

[Hình 1.1: Các thành phần chính của mô hình MVC 12](#_Toc154185501)

[Hình 1.2: Kiến trúc của linq trong .NET Framework 3.5 20](#_Toc154185502)

[Hình 1.3: Các thành phần chính của mô hình MVC 22](#_Toc154185503)

[Hình 1.4: Kiến trúc của Entity Framework 23](#_Toc154185504)

[Hình 2.1: Biểu đồ Usecase 27](#_Toc154185505)

[Hình 2.2: Các usecase chính 27](#_Toc154185506)

[Hình 2.3: Các usecase thứ cấp 29](#_Toc154185507)

[Hình 2.4: Quan hệ giữa các usecase 30](#_Toc154185508)

[Hình 2.5: Biểu đồ các lớp entity của hệ thống 45](#_Toc154185509)

[Hình 2.6: Biểu đồ trình tự usecase Đăng nhập 46](#_Toc154185510)

[Hình 2.7: Biểu đồ lớp phân tích usecase Đăng nhập 47](#_Toc154185511)

[Hình 2.8: Biểu đồ trình tự usecase Đăng ký 48](#_Toc154185512)

[Hình 2.9: Biểu đồ lớp phân tích usecase Đăng ký 49](#_Toc154185513)

[Hình 2.10: Biểu đồ trình tự usecase Bảo trì loại mặt hàng 52](#_Toc154185514)

[Hình 2.11: Biểu đồ lớp phân tích usecase Bảo trì loại mặt hàng 53](#_Toc154185515)

[Hình 2.12: Biểu đồ trình tự usecase Xem mặt hàng theo loại mặt hàng 54](#_Toc154185516)

[Hình 2.13: Biểu đồ lớp phân tích usecase Xem mặt hàng theo loại mặt hàng 55](#_Toc154185517)

[Hình 2.14: Biểu đồ trinh tự usecase Bảo trì khuyến mại 57](#_Toc154185518)

[Hình 2.15: Biểu đồ lớp phân tích usecase Bảo trì khuyến mại 58](#_Toc154185519)

[Hình 2.16: Biểu đồ trinh tự usecase Bảo trì mặt hàng 60](#_Toc154185520)

[Hình 2.17: Biểu đồ lớp phân tích usecase Bảo trì mặt hàng 61](#_Toc154185521)

[Hình 2.18: Biểu đồ trình tự usecase Xem chi tiết mặt hàng 62](#_Toc154185522)

[Hình 2.19: Biểu đồ lớp phân tích usecase Xem chi tiết mặt hàng 62](#_Toc154185523)

[Hình 2.20: Biểu đồ trình tự usecase Thêm vào giỏ hàng 63](#_Toc154185524)

[Hình 2.21: Biểu đồ lớp phân tích usecase Thêm vào giỏ hàng 64](#_Toc154185525)

[Hình 2.22: Biểu đồ trình tự usecase Quản lý tài khoản 66](#_Toc154185526)

[Hình 2.23: Biểu đồ lớp phân tích usecase Quản lý tài khoản 67](#_Toc154185527)

[Hình 2.24: Biểu đồ các lớp phân tích nhóm usecase chính 68](#_Toc154185528)

[Hình 2.25: Biểu đồ các lớp phân tích nhóm usecase thứ cấp 69](#_Toc154185529)

[Hình 2.26: Biểu đồ thực thể liên kết mức logic 72](#_Toc154185530)

[Hình 2.27: Biểu đồ thực thể liên kết mức vật lý 73](#_Toc154185531)

[Hình 3.1: Giao diện Trang chủ 81](#_Toc154185532)

[Hình 3.2: Giao diện Trang Câu chuyện thật TH 82](#_Toc154185533)

[Hình 3.3: Giao diện Trang Tư vấn 82](#_Toc154185534)

[Hình 3.4: Giao diện Trang Liên hệ 83](#_Toc154185535)

# DANH MỤC BẢNG

[Bảng 2.1: Loại mặt hàng 74](#_Toc154185559)

[Bảng 2.2: Nhóm mặt hàng 74](#_Toc154185560)

[Bảng 2.3: Mặt hàng 74](#_Toc154185561)

[Bảng 2.4: Hình ảnh 75](#_Toc154185562)

[Bảng 2.5: Tư vấn mặt hàng 75](#_Toc154185563)

[Bảng 2.6: Khách hàng 75](#_Toc154185564)

[Bảng 2.7: Giỏ hàng 76](#_Toc154185565)

[Bảng 2.8: Đơn hàng 76](#_Toc154185566)

[Bảng 2.9: Chi tiết đơn hàng 76](#_Toc154185567)

[Bảng 2.10: Nhân viên 77](#_Toc154185568)

[Bảng 2.11: Chức vụ 77](#_Toc154185569)

[Bảng 2.12: Khuyến mại 77](#_Toc154185570)

# MỞ ĐẦU

**Lý do chọn đề tài**

Hiện nay trên thế giới thương mại điện tử đang phát triển rất mạnh mẽ. Kỹ thuật số giúp chúng ta tiết kiệm đáng kể các chi phí nhờ chi phí vận chuyển trung gian và đặc biệt là giúp tiết kiệm thời gian để con người đầu tư vào các hoạt động khác. Khi mà Internet trở nên thân quen và dần trở thành một công cụ không thể thiếu trong cuộc sống thì lợi ích của website đối với việc quảng bá sản phẩm và thương hiệu của một công ty thật là to lớn. Trang web trở thành một cửa ngõ để doanh nghiệp tiếp thị sản phẩm của mình đến khách hàng trên khắp nơi trên toàn thế giới, nó đóng một vai trò rất quan trọng trong công cuộc phát triển và đấu tranh trong lĩnh vực kinh doanh. TH True là một trong những thương hiệu sữa nổi tiếng của Việt Nam, với chỉ hơn 10 năm thành lập nhưng đã trở thành cái tên nổi bật trong ngành sữa, hiện nay TH True Milk đang dần tiến đến trở thành thương hiệu cung cấp các sản phẩm sữa bò hàng đầu Việt Nam và vươn tầm quốc tế.Chính vì vậy nhóm em chọn đề tài về: “Xây dựng Website quảng cáo và bán hàng cho TH True Milk”.

**Mục đích**

Đề tài được phát triển nhằm mục đích thương mại điện tử, quản lý việc mua bán thông qua mạng internet, nâng cao trải nghiệm sử dụng website của người dùng qua đó đưa thương hiệu đến rông rãi với người dùng hơn. Thiết kế một website thân thiện, dễ dàng sử dụng đối với khách hàng và tối ưu trên các thiết bị. Thiết kế website thương hiệu TH True Milk, giúp khách hàng dễ nhận diện và gắn bó với thương hiệu.

**Đối tượng phạm vi nghiên cứu**

Đối tượng:

* Gia đình có trẻ nhỏ, người lớn tuổi và những người tiêu dùng chú trọng đến sức khỏe
* Những người quan tâm đến nguồn gốc tự nhiên của sản phẩm

Phạm vi nghiên cứu:

* Thói quen tiêu thụ: tần suất, lượng tiêu thụ và thời điểm mua sắm phổ biến
* Phản hồi và đánh giá của khách hàng: Tìm hiểu về ý kiến và trải nghiệm của người tiêu dùng sau khi sử dụng sản phẩm
* Thách thức rủi ro khi mua sắm trực tuyến: vận chuyển, bảo quản và thanh toán
* Yêu cầu về thông tin sản phẩm: thành phần dinh dưỡng, hướng dẫn sử dụng, ngày sản xuất, hạn sử dụng

**Ý nghĩa Khoa học**

Nghiên cứu thị trường: Việc phát triển trang web này có thể bắt đầu với một quá trình nghiên cứu thị trường sâu rộng để hiểu rõ hơn về ngành công nghiệp sữa và thị trường tiêu dùng. Điều này đòi hỏi việc thu thập và phân tích dữ liệu về cơ cấu thị trường, nguồn cung cấp, khách hàng tiềm năng và đối thủ cạnh tranh.

Phân tích dữ liệu khách hàng: Trang web có thể thu thập dữ liệu về hành vi của khách hàng, sở thích, và phản hồi của họ. Dữ liệu này có thể được sử dụng để phân đoạn thị trường và tạo ra chiến lược tiếp cận khách hàng hiệu quả.

Khoa học dữ liệu và phân tích: Dữ liệu thu thập từ trang web có thể được phân tích sâu hơn bằng các phương pháp khoa học dữ liệu để hiểu rõ hơn về mô hình tiêu dùng và phản hồi của khách hàng. Điều này giúp cải thiện chiến lược kinh doanh và tiếp thị.

**Ý nghĩa thực tiễn**

Quảng cáo và tiếp thị: Trang web sẽ là một công cụ quan trọng để tiếp cận và tương tác với khách hàng tiềm năng. Nó cho phép Th True Milk quảng cáo sản phẩm và dịch vụ của họ, tạo nền tảng để thực hiện chiến dịch quảng cáo trực tuyến và tiếp cận đối tượng mục tiêu.

Bán hàng trực tuyến: Trang web có thể tích hợp chức năng mua sắm trực tuyến, cho phép khách hàng mua sữa Th True Milk và các sản phẩm liên quan một cách thuận tiện. Điều này tạo ra cơ hội bổ sung để tăng doanh số bán hàng và thu hút khách hàng trực tuyến.

Tạo mối quan hệ khách hàng: Trang web cung cấp một kênh để tương tác với khách hàng, thu thập phản hồi và cung cấp dịch vụ khách hàng. Điều này giúp cải thiện mối quan hệ với khách hàng và tạo lòng tin.

Theo dõi và đánh giá hiệu suất: Trang web cho phép theo dõi hiệu suất các chiến dịch quảng cáo, đo lường chức năng bán hàng trực tuyến và đánh giá hiệu suất tổng thể của trang web. Điều này giúp điều chỉnh chiến lược tiếp thị và cải thiện trải nghiệm của khách hàng.

# Giới thiệu tổng quan

## Giới thiệu chung về đề tài

Đề tài "Xây dựng trang web quảng cáo và bán hàng cho hãng sữa Th True Milk" là một dự án quan trọng và hấp dẫn trong lĩnh vực tiếp thị trực tuyến và kinh doanh điện tử. Trang web này được thiết kế và phát triển để hỗ trợ hãng sữa Th True Milk trong việc quảng bá sản phẩm, tạo dấu ấn thương hiệu, tăng doanh số bán hàng và cung cấp dịch vụ khách hàng. Với một website thương mại điện tử sẽ giúp nhãn hàng cập nhật thông tin về giá, sản phẩm, dịch vụ đến với người tiêu dùng một cách nhanh chóng. Ngoài ra, thông qua website, có thể hỗ trợ tối đa người dùng một cách tốt nhất. Bất cứ khi nào khách hàng cần được tư vấn, đội ngũ chăm sóc khách hàng sẽ tư vấn kịp thời, đảm bảo không để khách hàng chờ đợi lâu.

**Mục tiêu chính**

* Quảng cáo: Trang web sẽ là một nền tảng quảng cáo trực tuyến, giúp Th True Milk tiếp cận và thu hút sự chú ý của khách hàng tiềm năng. Nó sẽ chứa các nội dung quảng cáo, video, hình ảnh và thông tin sản phẩm để giới thiệu sữa Th True Milk.
* Bán hàng trực tuyến: Trang web cung cấp tính năng mua sắm trực tuyến, cho phép khách hàng dễ dàng chọn mua sản phẩm sữa Th True Milk và các sản phẩm liên quan một cách thuận tiện từ nhà.
* Tương tác khách hàng: Ngoài việc quảng cáo và bán hàng, trang web cũng là một kênh tương tác với khách hàng. Khách hàng có thể đặt câu hỏi, đưa ra phản hồi và tìm kiếm thông tin về sản phẩm và thương hiệu.

**Thiết kế và trải nghiệm người dùng:**

* Trang web cần được thiết kế với giao diện thân thiện, dễ sử dụng và hấp dẫn. Màu sắc, hình ảnh và nội dung cần phản ánh đúng tinh thần của thương hiệu Th True Milk.
* Phải đảm bảo trải nghiệm mua sắm trực tuyến thuận tiện và an toàn để khách hàng có thể thực hiện giao dịch một cách dễ dàng và tin tưởng.

**Quản lý dữ liệu và phân tích:**

* Dữ liệu từ trang web, bao gồm thông tin về người dùng, hành vi mua sắm và tương tác, sẽ được thu thập và quản lý một cách an toàn và bảo mật.
* Phân tích dữ liệu sẽ giúp đánh giá hiệu suất của trang web và các chiến dịch tiếp thị trực tuyến. Dữ liệu này sẽ được sử dụng để cải thiện sản phẩm, dịch vụ và chiến lược kinh doanh.

**Quyền lợi và lợi ích:**

* Trang web sẽ mang lại lợi ích cho Th True Milk bằng cách tạo một kênh tiếp thị hiệu quả, tăng doanh số bán hàng và cải thiện quan hệ khách hàng.
* Khách hàng cũng sẽ được hưởng lợi với sự thuận tiện khi mua sắm trực tuyến và cơ hội tương tác với thương hiệu.

Tóm lại, việc xây dựng trang web quảng cáo và bán hàng cho hãng sữa Th True Milk là một dự án quan trọng trong việc phát triển thương hiệu và kinh doanh của họ trên môi trường trực tuyến ngày càng cạnh tranh.

## Nghiên cứu hiện trạng và giải quyết vấn đề

### *Hiện trạng cơ cấu tổ chức*

Xây dựng một trang web quảng cáo và bán hàng cho hãng sữa TH true milk đòi hỏi một cơ cấu tổ chức cụ thể để đảm bảo sự hiệu quả trong quá trình phát triển và quản lý dự án. Dưới đây là một mô hình cơ cấu tổ chức đề xuất cho dự án này:

1. Ban Quản lý Dự án:

* Project Manager (Quản lý Dự án): Người chịu trách nhiệm chung cho việc quản lý dự án, xác định mục tiêu và tiến độ.
* Product Owner (Chủ sở hữu sản phẩm): Người đại diện cho hãng sữa TH true milk, chịu trách nhiệm xác định yêu cầu của sản phẩm và xác định ưu tiên.

1. Nhóm Phát triển Website

* Front-end Developers (Lập trình viên Front-end): Xây dựng giao diện người dùng và trải nghiệm người dùng trên trang web.
* Back-end Developers (Lập trình viên Back-end): Phát triển hệ thống máy chủ và cơ sở dữ liệu, quản lý dữ liệu và giao tiếp với front-end.
* UI/UX Designers (Thiết kế giao diện người dùng/Trải nghiệm người dùng): Thiết kế giao diện trực quan và hấp dẫn.

1. Nhóm Quảng cáo và Tiếp thị

* Digital Marketers (Nhà tiếp thị số): Quảng cáo trực tuyến, tối ưu hóa tìm kiếm, quảng cáo trên mạng xã hội và quảng cáo đích thực cho trang web.
* Content Creators (Người tạo nội dung): Tạo nội dung chất lượng, bài viết blog, hình ảnh, video và bài viết trên trang web.
* SEO Specialists (Chuyên gia SEO): Tối ưu hóa trang web cho công cụ tìm kiếm (SEO) để tăng hiển thị trong kết quả tìm kiếm.

1. Nhóm Quản lý Dữ liệu và Thương mại điện tử:

* Database Administrators (Quản trị cơ sở dữ liệu): Quản lý và bảo trì cơ sở dữ liệu sản phẩm và thông tin khách hàng.
* E-commerce Managers (Quản lý thương mại điện tử): Quản lý mua sắm trực tuyến, quản lý sản phẩm và dịch vụ trên trang web.

1. Nhóm Hỗ trợ Khách hàng:

* Customer Support Representatives (Đại diện hỗ trợ khách hàng): Trả lời câu hỏi của khách hàng, giải quyết vấn đề và cung cấp hỗ trợ trực tuyến.

1. Nhóm Quản lý Dự án và Tài chính:

* Project Coordinators (Các nhân viên phối hợp dự án): Hỗ trợ quản lý dự án và theo dõi tiến độ.
* Financial Analysts (Chuyên viên phân tích tài chính): Quản lý nguồn lực tài chính và thực hiện dự án theo ngân sách.
* Cơ cấu tổ chức này có thể điều chỉnh tùy theo quy mô của dự án và các yêu cầu cụ thể của hãng sữa TH true milk. Điều quan trọng là đảm bảo tất cả các phần tử này làm việc cùng nhau để xây dựng một trang web chất lượng cao để quảng cáo và bán sản phẩm của hãng.

### *Mô tả hoạt động của các bộ phận*

1. Ban Quản lý Dự án:
   * Project Manager: Lập kế hoạch, theo dõi tiến độ, quản lý tài nguyên và đảm bảo dự án được thực hiện theo kế hoạch.
   * Product Owner: Xác định yêu cầu sản phẩm, ưu tiên công việc phát triển dự án, và đảm bảo sản phẩm đáp ứng mong đợi của hãng TH true milk.
2. Nhóm Phát triển Website:
   * Front-end Developers: Tạo giao diện người dùng hấp dẫn và dễ sử dụng.
   * Back-end Developers: Xây dựng hệ thống máy chủ và cơ sở dữ liệu, đảm bảo tính ổn định và bảo mật.
   * UI/UX Designers: Thiết kế giao diện và trải nghiệm người dùng để tạo ra trang web hấp dẫn và hiệu quả.
3. Nhóm Quảng cáo và Tiếp thị:
   * Digital Marketers: Tối ưu hóa chiến lược quảng cáo trực tuyến và tối ưu hóa tìm kiếm để tạo lưu lượng truy cập trang web.
   * Content Creators: Tạo nội dung chất lượng để thu hút và giữ chân khách hàng.
   * SEO Specialists: Tối ưu hóa trang web để cải thiện vị trí trong kết quả tìm kiếm.
4. Nhóm Quản lý Dữ liệu và Thương mại điện tử:
   * Database Administrators: Quản lý cơ sở dữ liệu sản phẩm và thông tin khách hàng để đảm bảo tính toàn vẹn và bảo mật.
   * E-commerce Managers: Quản lý trang web thương mại điện tử, bao gồm quản lý sản phẩm, giỏ hàng và thanh toán trực tuyến.
5. Nhóm Hỗ trợ Khách hàng:
   * Customer Support Representatives: Cung cấp hỗ trợ khách hàng qua trực tuyến, trả lời câu hỏi, giải quyết vấn đề và tạo trải nghiệm khách hàng tích cực.
6. Nhóm Quản lý Dự án và Tài chính:
   * Project Coordinators: Hỗ trợ quản lý dự án, theo dõi tiến độ và báo cáo cho Quản lý Dự án.
   * Financial Analysts: Quản lý nguồn lực tài chính, lập ngân sách và đảm bảo dự án được thực hiện trong phạm vi ngân sách.

### *Xác định yêu cầu*

Khi chưa có website hỗ trợ độ phủ sóng của sản phẩm chưa được cao cộng với việc công nghệ phát triển mọi người có xu hướng đặt qua các app website thay vì phải đến trực tiếp cửa hàng. Vì vậy hiện tại cần có một website cho TH True Milk để hỗ trợ trong việc quảng cáo rộng rãi trong nước và ra cả nước ngoài. Với một hệ thống như sau

Hệ thống có 2 phần:

a.Phần thứ nhất: Khách hàng

Khách hàng có nhu cầu mua hàng, họ sẽ tìm kiếm mặt hàng trên website, khi quyết định mua sẽ đặt hàng thông qua giao diện website. Đơn hàng sẽ được chuyển về cho cửa hàng, công ty. Vì vậy giao diện khách hàng cần có các chức năng sau:

Giao diện người dùng thân thiện quảng cáo sản phẩm gây ấn tượng với khách hàng nhằm đánh mạnh vào nhu cầu, tâm lý người dùng. Việc người dùng sử dụng website của bạn thêm một giây là gia tăng thêm tỷ lệ mua hàng của khách hàng.

Hiển thị các danh sách các mặt hàng để khách hàng xem, lựa chọn và mua

Sau khi khách hàng đã lựa chọn xong và muốn đặt hàng, cần có giao diện đặt hàng cho người dùng bao gồm các thông tin về hàng hóa đã chọn, số lượng, số tiền, tổng tiền đơn hàng, các thông tin cá nhân của khách hàng để phục vụ việc liên lạc lại với khách hàng.

Ngoài ra có thể phát triển một trang phản hồi khách hàng nhằm thu thập ý kiến, câu hỏi của khách hàng, phục vụ việc phát triển, giải đáp các thắc mắc của khách hàng.

b.Phần thứ hai: Giao diện quản trị.

Người làm chủ ứng dụng được cấp tài khoản riêng để có thể kiểm soát toàn các hoạt động của ứng dụng. Về phần này cần có các chức năng chính:

¬ Chức năng thêm, sửa, xóa các loại mặt hàng, nhà sản xuất, …

¬ Tiếp nhận đơn hàng của khách hàng và sửa lý đơn hàng

¬ Thống kê doanh thu

¬ Một số chức năng nâng cao theo yêu cầu như tạo giao diện menu động, tùy chỉnh theo ý người quản trị website, …

Giao diện quản trị nên làm đơn giản, tập chung vào các chức năng, đơn giản, dễ sử dụng

## Các giai đoạn phát triển dự án

* Giai đoạn 1: Phân tích các yêu cầu webstie, front-end website
* Giai đoạn 2: Xây dựng đầy đủ các chức năng cơ bản, sử dụng cơ dở dữ liệu SQL và xây dựng website theo mô hình 3 lớp kết hợp MVC 5

## Vấn đề - giải pháp

* Khó khăn

- Kiến thức thực tế còn hạn chế

- Công nghệ biến đổi theo từng đơn vị thời gian nên cần có chiến lược phát triền, thời gian và công sức tìm hiểu.

* Giải pháp:

- Tự tìm dành thời gian tìm hiểu những công nghệ mới, giao lưu, học hỏi từ những người có kinh nghiệm trong ngành công nghệ thông tin và giáo viên hướng dẫn

## Giới thiệu công nghệ, ngôn ngữ xây dựng website

Việc xây dựng và phát triển website hiện nay có rất nhiều ngôn ngữ như sử dụng javascript có MeotorJS, Angularjs, C# , Java, PHP ... Trong quá trình tìm hiểu và xin ý kiến của những người đi trước và kiến thức bản thân trong quá trình rèn luyện học tập, em phát triển website theo ASP.NET sử dụng mô hình MVC 5. Về thiết kế giao diện em sử dụng framework Bootstrap 3 giúp giảm thời gian phát triển giao diện và kế thừa những ưu điểm của bootstrap. Bên cạnh đó sử dụng javascript, jquery, ajax … nhằm tối ưu hoá sử lý website, sử dụng SQL server làm cơ sở dữ liệu, enity framework để kết nối với cơ sở dữ liệu, sử dụng linq thay câu truy vấn SQL.

### *ASP.NET và MVC*

#### *ASP.NET*

ASP.NET là Active Server Pages .NET (.NET là .NET framework). Nói đơn giản và ngắn gọn thì ASP.NET là một công nghệ có tính cách mạng dùng để phát triển các ứng dụng về mạng hiện nay cũng như trong tương lai (ASP.NET is a revolutionary technology for developing web applications).

ASP.NET là một phương pháp tổ chức hay khung tổ chức (framework) để thiết lập các ứng dụng hết sức hùng mạnh cho mạng dựa trên CLR (Common Language Runtime) chứ không phải là một ngôn ngữ lập trình.

ASP.NET là một cơ cấu trong các cơ cấu của hệ điều hành Windows dưới dạng nền hay khung .NET. Như vậy ASP.NET không những có thể dùng các object của các ứng dụng cũ mà còn có thể sử dụng tất cả mọi tài nguyên Windows có một cách dễ dàng.

Tập tin của ASP.NET (ASP.NET file) có extension là .ASPX, còn tập tin của ASP là .ASP

Tập tin của ASP.NET (ASP.NET file) được phân tích ngữ pháp bởi XSPISAPI.DLL. Còn tập tin của ASP được phân tích bởi i ASP.DLL.

ASP.NET là kiểu mẫu lập trình phát động bằng sự kiện (event driven), còn các trang ASP được thi hành theo tuần tự từ trên xuống dưới.

ASP.NET sử dụng trình biên dịch (compiled code) nên rất nhanh, còn ASP dùng trình thông dịch (interpreted code) do đó hiệu suất và tốc độ cũng thua kém hẳn.

ASP.NET hỗ trợ gần 25 ngôn ngữ lập trình mới với .NET và chạy trong môi trường biên dịch (compiled environment), còn ASP chỉ chấp nhận VBScript và JavaScript nên ASP chỉ là một scripted language trong môi trường thông dịch (in the interpreter environment). Không những vậy, ASP.NET còn kết hợp nhuần 12 nhuyễn với XML (Extensible Markup Language) để vận chuyển các thông tin qua mạng.

ASP.NET hỗ trợ tất cả các browser và quan trọng hơn nữa là hỗ trợ các thiết bị lưu động (mobile devices). Chính các thiết bị lưu động, mà mỗi ngày càng phổ biến, đã khiến việc dùng ASP trong phát triển mạng nhằm vươn tới thị trường mới đó trở nên vô cùng khó khăn.

#### *MVC*

MVC là viết tắt của Model – View – Controller. Là một trong những design pattern. Được vận hành để tách mã lệnh thành 3 phần riêng biệt. Ở mỗi phần MVC sẽ có những chức năng đặc thù. Để xử lý các tác vụ mà request gởi tới. MVC làm cho mã lệnh trở nên trong sáng, dễ phát triển và dễ nâng cấp theo thời gian.

Để làm việc tốt đối với MVC, chúng ta cần nắm thật vững kiến thức OOP. Bản chất của các framework khác cũng được hình thành trên lý thuyết MVC. Do vậy, nếu chúng ta nắm tốt MVC, thì ở những framework khác chắc chắn sẽ không cảm thấy khó hiểu.

* Model: Là thành phần chịu trách nhiệm xử lý các thao tác trên database. Và gửi trả kết quả thông qua view.
* View: Là phần hiển thị thông tin trên website, sau khi đi qua controller và nhận kết quả từ phía model thì view là bước cuối cùng để chuyển thông tin tới người dùng.
* Controller: Là phần điều hướng các request tới những tác vụ tương ứng. Controller là một phần không thể thiếu ở bất cứ framework nào. Vì nó có trách nhiệm gởi và nhận request từ hệ thống tới người sử dụng.

**Xuất xứ**

Tất cả bắt đầu vào những năm 70 của thế kỷ 20, tại phòng thí nghiệm Xerox PARC ở Palo Alto. Sự ra đời của giao diện đồ họa (Graphical User Interface) và lập trình hướng đối tượng (Object Oriented Programming) cho phép lập trình viên làm việc với những thành phần đồ họa như những đối tượng đồ họa có thuộc tính và phương thức riêng của nó. Không dừng lại ở đó, những nhà nghiên cứu ở Xerox PARC còn đi xa hơn khi cho ra đời cái gọi là kiến trúc MVC (viết tắt của Model – View – Controller).

MVC được phát minh tại Xerox Parc vào những năm 70, bởi TrygveReenskaug. MVC lần đầu tiên xuất hiện công khai là trong Smalltalk-80. Sau đó trong một thời gian dài hầu như không có thông tin nào về MVC, ngay cả trong tài liệu 80 Smalltalk. Các giấy tờ quan trọng đầu tiên đƣợc công bố trên MVC là “A Cookbook for Using the Model-View-Controller User Interface Paradigm in Smalltalk – 80”, bởi Glenn Krasner và Stephen Pope, xuất bản trong tháng 8 / tháng 9 năm 1988.

**Kiến trúc mô hình**

MVC Trong kiến trúc MVC, một đối tượng đồ họa người dùng (GUI Component) bao gồm 3 thành phần cơ bản: Model, View, và Controller. Model có trách nhiệm đối với toàn bộ dữ liệu cũng như trạng thái của đối tượng đồ họa. View chính là thể hiện trực quan của Model, hay nói cách khác chính là giao diện của đối tượng đồ họa. Và Controller điều khiển việc tương tác giữa đối tượng đồ họa với người sử dụng cũng như những đối tượng khác.

###### Hình 1.1: Các thành phần chính của mô hình MVC

Khi người sử dụng hoặc những đối tượng khác cần thay đổi trạng thái của đối tượng đồ họa, nó sẽ tương tác thông qua Controller của đối tượng đồ họa. Controller sẽ thực hiện việc thay đổi trên Model. Khi có bất kỳ sự thay đổi nào xảy ra ở Model, nó sẽ phát thông điệp (broadcast message) thông báo cho View và Controller biết. Nhận được thông điệp từ Model, View sẽ cập nhật lại thể hiện của mình, đảm bảo rằng nó luôn là thể hiện trực quan chính xác của Model. Còn Controller, khi nhận được thông điệp từ Model, sẽ có những tương tác cần thiết phản hồi lại người sử dụng hoặc các đối tượng khác

**Đặc điểm của mô hình MVC**

Cái lợi ích quan trọng nhất của mô hình MVC là nó giúp cho ứng dụng dễ bảo trì, module hóa các chức năng, và được xây dựng nhanh chóng. MVC tách các tác vụ của ứng dụng thành các phần riêng lẻ model, view, controller giúp cho việc xây dựng ứng dụng nhẹ nhàng hơn. Dễ dàng thêm các tính năng mới, và các tính năng cũ có thể dễ dàng thay đổi. MVC cho phép các nhà phát triển và các nhà thiết 14 kế có thể làm việc đồng thời với nhau. MVC cho phép thay đổi trong 1 phần của ứng dụng mà không ảnh hưởng đến các phần khác

Sở dĩ như vậy vì kiến trúc MVC đã tách biệt (decoupling) sự phụ thuộc giữa các thành phần trong một đối tượng đồ họa, làm tăng tính linh động (flexibility) và tính tái sử dụng (reusebility) của đối tượng đồ họa đó. Một đối tượng đồ họa bấy giờ có thể dễ dàng thay đổi giao diện bằng cách thay đổi thành phần View của nó trong khi cách thức lưu trữ (Model) cũng như xử lý (Controller) không hề thay đổi. Tương tự, ta có thể thay đổi cách thức lưu trữ (Model) hoặc xử lý (Controller) của đối tượng đồ họa mà những thành phần còn lại vẫn giữ nguyên.

Chính vì vậy mà kiến trúc MVC đã được ứng dụng để xây dựng rất nhiều framework và thư viện đồ họa khác nhau. Tiêu biểu là bộ thư viện đồ họa của ngôn ngữ lập trình hướng đối tượng SmallTalk (cũng do Xerox PARC nghiên cứu và phát triển vào thập niên 70 của thế kỷ 20). Các Swing Components của Java cũng được xây dựng dựa trên kiến trúc MVC. Đặc biệt là nền tảng ASP.NET MVC Frame work.

**Tiềm hiểu về Controler**

Controller có trách nhiệm chính là điều hướng các yêu cầu của người sử dụng. Như vậy trên toàn ứng dụng của ta, tất cả các request đều sẽ phải đi tới controller. Và tại đây, ứng với các tham số người sử dụng truyền mà ta đưa họ đến một tác vụ nào đó trên ứng dụng.

Tại các tác vụ này, chúng sẽ thông qua lớp model để làm việc và trả kết quả trở về controller. Cuối cùng controller sẽ đẩy dữ liệu thao tác tới view. View là thành phần cuối cùng mà người sử dụng nhận được khi họ giở request tới ứng dụng. Có thể hiểu controller, giống với kỹ thuật đa cấp với các tác vụ chạy ứng dụng phân cấp theo từng nhánh riêng biệt như: Module, action,….

**Tiềm hiểu Model**

Model là thành phần chủ yếu được sử dụng để thao tác xử lý dữ liệu. Trong các framework, Model vẫn thường sử dụng theo phương thức Active Record. Một trong những design pattern. Chúng có tác dụng rút ngắn thời gian viết câu truy vấn cho người sử dụng. Biến những câu truy vấn phức tạp trở nên gần gũi và thân thiện với người sử dụng thông qua các thư viện được định nghĩa sẵn.

Model thường sẽ là các phương thức có trách nhiệm xử lý các tác vụ như: select, insert, update, delete các record trong database. Ứng với các lấy dữ liệu, model thường sử dụng mảng để gởi trả kết quả về. Vì mảng có thể cho phép model lưu trữ nhiều thông tin hơn, nên thường các record khi bóc tách chúng sẽ mang các dữ liệu của database một cách chi tiết.

Khi sử dụng models, ta cũng cần tuân theo nguyên tắc chính của chúng là không 15 xuất giá trị trực tiếp trong model. Mà tất cả những dữ liệu ấy, phải đưa vào mảng và trả về theo phương thức. Và tiếp tục ở view ta sẽ sử dụng nó để lấy dữ liệu ra.

**Tiềm hiểu về View**

View là phần hiển thị thông tin tương phản khi gởi và nhận request. Trước đây, khi người lập trình chưa nghĩ tới view. Họ thường thao tác xử lý dữ liệu ngay trực tiếp trên ứng dụng và đổ cả dữ liệu ngay trên file PHP đó. Điều này làm cho ứng dụng trở nên cồng kềnh, và đặc biệt rất khó cho việc bảo trì nâng cấp sau này. Nhất là đối với designer, việc thay đổi giao diện của một website luôn làm cho họ cảm thấy đau đầu vì phải chỉnh sửa thẳng vào core.

Trước đây, để giải quyết tình huống này. Người ta thường sử dụng template để phân tách website thành 2 mảng riêng biệt. Một là giao diện và một là core. Việc chỉnh sửa giao diện trở nên đơn giản hơn đối với họ so với cách viết thập cẩm kia. Tuy nhiên, các thư viện này thực chất sẽ làm cho ứng dụng của chúng ta trở nên chậm chạp hơn bao giờ hết, bởi chúng phải phiên dịch nhiều lần các kịch bản.

* Ưu Điểm:

- MVC làm cho ứng dụng trở nên trong sáng, giúp lập trình viên phân tách ứng dụng thành ba lớp một cách rõ ràng. Điều này sẽ rất giúp ích cho việc phát triển những ứng dụng xét về mặt lâu dài cho việc bảo trì và nâng cấp hệ thống.

- MVC hiện đang là mô hình lập trình tiên tiến bậc nhất hiện nay, điều mà các framework vẫn đang nổ lực để hướng tới sự đơn giản và yếu tố lâu dài cho người sử dụng.

* Khuyết điểm:

- Mặc dù, MVC tỏ ra lợi thế hơn nhiều so với cách lập trình thông thường. Nhưng MVC luôn phải nạp, load những thư viện đồ sộ để xử lý dữ liệu. Chính điều này làm cho mô hình trở nên chậm chạp hơn nhiều so với việc code tay thuần túy.

- MVC đòi hỏi người tiếp cận phải biết qua OOP, có kinh nghiệm tương đối cho việc thiết lập và xây dựng một ứng dụng hoàn chỉnh. Sẽ rất khó khăn nếu OOP của người sử dụng còn yếu.

- MVC tận dụng mảng là thành phần chính cho việc truy xuất dữ liệu. Nhất là với việc sử dụng active record để viết ứng dụng. Chúng luôn cần người viết phải nắm vứng mô hình mảng đa chiều.

### *Javascript*

JavaScript được phát minh bởi Brendan Eich của Netscape (với trình duyệt Navigator 2.0) vào năm 1995, và xuất hiện ở hầu hết các công cụ trình duyệt của Netscape và Microsoft kể từ năm 1996.

Ngày nay, gần như các trình duyệt đình đám đều có hỗ trợ Javascript, và chính Javascript là nhân tố gần đây đã châm ngòi cho cuộc chiến các trình duyệt

Javascript đã ra đời từ khá lâu, nhưng cho đến ngày nay, nó vẫn mang 1 tầm quan trọng rất lớn, vì vậy học Javascript là điều cần thiết.

**Javascript là gì?**

Về phương diện thực thi ngôn ngữ

Hiện nay khi phát triển các trang web động, có hai phương thức rất phổ biến đó là Server-Side và Client-Side.

**Client-Side**: là phương thức sẽ xử lý các mã lệnh bởi mỗi browser riêng biệt tại máy người dùng, sau đó gửi kết quả lên cho Server.

**Server-Side**: thì lại gửi các mã lệnh cho server xử lý trước, sau đó server gửi kết quả lại cho browser.

JavaScript là một ngôn ngữ lập trình kịch bản Client-Side (Client-Side Script Language), và dĩ nhiên nó được thực thi tại máy người dùng. Và vì javascript là một Client-Side Script Language nên nó không có khả năng kết nối và thao tác với CSDL trên Server.

Về phương diện phong cách ngôn ngữ lập trình

Các tài liệu cũ trước đây nói Javascript là ngôn ngữ lập trình dựa vào đối tượng (Object-based language). Nhưng từ lúc Ajax ra đời, Javascript trở nên hướng đối tượng hơn rất nhiều. Vì vậy, chúng ta có thể coi Javascript là một ngôn ngữ lập trình hướng đối tượng (Object-Oriented programming). Về cú pháp, Javascript cũng tương tự như C, Perl và Java, …ví dụ mệnh đề lặp if, while, for, tương tự nhau.

Nói tóm lại:

**Javascript là một ngôn ngữ lập trình:**

* Kịch bản Client-Side (Client-Side Script Language).
* Hướng đối tượng (Object-Oriented programming).

**Javascript được sử dụng để làm gì?**

* JavaScript được sử dụng nhằm bổ sung sự tương tác cho các trang HTML.
* JavaScript có thể đáp ứng các sự kiện như tải hay loại bỏ các form. Khả năng này cho phép JavaScript trở thành một ngôn ngữ script động.
* JavaScript có thể được sử dụng để xác nhận dữ liệu người dùng nhập vào trước khi nó được chuyển đến server.
* Sử dụng Javascript có thể giúp website của bạn tương tác với người dùng một cách uyển chuyển hơn.
* Tùy biến trình duyệt.

**\* Ưu điểm của Javascript:**

* Hoàn toàn miễn phí và dễ học.
* JavaScript được thiết kế độc lập với hệ điều hành. Nó có thể chạy trên bất kỳ hệ điều hành nào có trình duyệt hỗ trợ JavaScript.
* Dễ dàng tương tác, điều khiển và tránh bớt việc xử lý từ phía server
* Nắm vững kiến thức JavaScript bây giờ rất hữu dụng cho các bạn sau này để có thê tiếp thu những công nghệ mới mà nó được gói gọn vào những ngôn ngữ như : Ajax, Atlas ….

**\*Nhược điểm của Javascript**

* Javascript không có trình biên dịch riêng mà được diễn dịch và chạy bởi trình duyệt hỗ trợ nó. Chính vì thế, nếu trình duyệt không hỗ trợ, hoặc không bật JS, nó sẽ không chạy được.
* JS có thể làm ứng dụng web của bạn trở nên nặng nề hơn.
* Bảo mật kém. Không có khả năng giấu mã.

**Nguyên tắc lập trình trong JavaScript**

* Bất cứ ngôn ngữ lập trình nào cũng đều phải có nguyên tắc của nó, và Javascript cũng ko ngoại lệ. Để bắt đầu học lập trình 1 ngôn ngữ nào, trước tiên chúng ta cần phải biết quy tắc của nó.
* Một số quy tắc quan trọng trong JS mà chúng ta cần phải nhớ như sau:

- Lệnh Javascript phân biệt chữ in hoa và chữ thường

- Mội câu lệnh Javascript đều kết thúc bằng dấu chấm phẩy “;”

- Các điều kiện phải được khai báo trong cặp dấu ngoặc đơn ().

- Khi sử dụng lệnh điều khiển, nếu sử dụng nhiều hơn 1 lệnh, bạn phải sử dụng cặp dấu ngoặc nhọn {}

- Javascript sử dụng dấu chấm “.” để tham chiếu đến 1 phương thức hay thuộc tính của đối tượng

### *Ajax*

Ajax là một kỹ thuật viết tắt của chữ AJAX = *Asynchronous JavaScript* and *XML*, đây là một công nghệ giúp chung ta tạo ra những Web động mà hoàn toàn không reload lại trang nên rất mượt và đẹp. Đối với công nghệ web hiện nay thì ajax không thể thiếu, nó là một phần làm nên sự sinh động cho website. Ajax được viết bằng ngôn ngữ Javascript nên nó chạy trên client, tức là mỗi máy (user) sẽ chạy độc lập hoàn toàn không ảnh hưởng lẫn nhau. Hiện nay có nhiều thư viện javascript như **jQuery** hay **Angular** đều hỗ trợ kỹ thuật này nhằm giúp chúng ta thao tác dễ dàng hơn.

### *Bootstrap*

Bootstrap là một **Font-end Framework** được viết bằng SASS và biên dịch thành CSS, nó là một bộ thư viện mạnh mẽ được tích hợp nhiều CSS có sẵn giúp cho việc lập trình HTML & CSS trở nên đơn giản và dễ dàng hơn. Thông thường khi chúng ta lập trình bằng CSS thuần thì công việc kiểm tra tính tương thích trên các trình duyệt khác nhau và trên các thiết bị di động khác nhau rất là khó khăn, nhưng với Bootstrap thì lập trình viên không cần phải tốn nhiều công sức nữa vì mọi thứ đã có người tạo nên Bootstrap lo.

Bootstrap chia layout của một trang web ra thành 960 Grid và gồm 12 cột, mỗi cột 80Grid và đây chính lả chuẩn thiết kế HTML & CSS của Bootstrap. Nó có hỗ trợ hầu hết các module của một trang web như menu, tabs, tooltip, popup, ... Ngoài ra nó còn sử dụng thêm Javascript để xử lý các hiệu ứng cấp cao, javascript của Bootstrap sử dụng jquery nên để sử dụng được bắt buộc bạn phải bổ sung thêm thư viện jQuery nữa.

Các lợi ích khi sử dung bootstrap

* Phát triển giao diện nhanh chóng.
* Dễ học, dễ sử dụng
* Hỗ trợ javascript ,jquery

### *Hệ cơ sở dữ liệu SQL server*

SQL là một công cụ quản lý dữ liệu được sử dụng phổ biến ở nhiều lĩnh vực. Đầu tiên SQL được sử dụng trong các ngôn ngữ quản lý CSDL và chạy trên máy đơn lẻ. Do sự phát triển nhanh chóng của nhu cầu xây dựng những CSDL lớn theo mô hình khách/chủ (Client/Server), nhiều phần mềm sử dụng ngôn ngữ SQL đã ra đời mà điển hình là MS SQL Server, Oracle, Sybase... Trong mô hình khách/chủ, toàn bộ CSDL được tập trung lưu trữ trên máy chủ (Server), mọi thao tác xử lý dữ liệu được thực hiện trên máy chủ bằng các lệnh SQL. Máy trạm (Client) chỉ dùng để cập nhật dữ liệu cho máy chủ hoặc lấy thông tin từ máy chủ. Trên lĩnh vực đang phát triển hiện nay là Internet, ngôn ngữ SQL lại càng đóng vai trò quan trọng hơn. Nó được sử dụng để nhanh chóng tạo ra các trang Web động (Dynamic Web Page). Trang Web động thường có nội dung được lấy ra từ CSDL. SQL có thể được sử dụng như một chất keo kết dính giữa CSDLvà trang Web. Khi người dùng yêu cầu, SQL sẽ thực hiện việc truy cập thông tin trong CSDL trên máy chủ và hiển thị kết quả trên trang Web. Và SQL cũng là công cụ để cập nhật thông tin cho CSDLđó.

### *Truy vấn Linq*

Để giảm gánh nặng thao tác trên nhiều ngôn ngữ khác nhau và cải thiện năng suất lập trình, Microsoft đã phát triển giải pháp tích hợp dữ liệu cho .NET 19 Framework có tên gọi là LINQ (Language Integrated Query), đây là thư viện mở rộng cho các ngôn ngữ lập trình C# và Visual Basic.NET (có thể mở rộng cho các ngôn ngữ khác) cung cấp khả năng truy vấn trực tiếp dữ liệu Object, cơ sở dữ liệu và XML.

Điểm mạnh của LINQ là “viết truy vấn cho rất nhiều các đối tượng dữ liệu”. Từ cơ sở dữ liệu, XML, Data Object … thậm chí là viết truy vấn cho một biến mảng đã tạo ra trước đó. Vì thế ta có các khái niệm như là LinQ to SQL, LinQ to XML,….

#### *Các thành phần của LINQ*

###### Hình 1.2: Kiến trúc của linq trong .NET Framework 3.5

##### *LINQ to Objects*

“LINQ to Objects” ở đây có nghĩa là nói đến cách sử dụn LINQ đối với các đối tượng Collection mà đã được thực thi giao diện IEnumerable hoặc IEnumerable tức những collection có thể “liệt kê” ra được. Đây là trường hợp sử dụng đơn giản nhất của LINQ khi làm việc với dữ liệu.

##### *LINQ to SQL*

LINQ to SQL là một phiên bản hiện thực hóa của O/RM (object relational mapping) có bên trong .NET Framework 3.5, nó cho phép bạn mô hình hóa một cơ sở dữ liệu dùng các lớp .NET. Sau đó bạn có thể truy vấn cơ sở dữ liệu dùng LINQ, cũng như cập nhật/thêm/xóa dữ liệu từ đó.

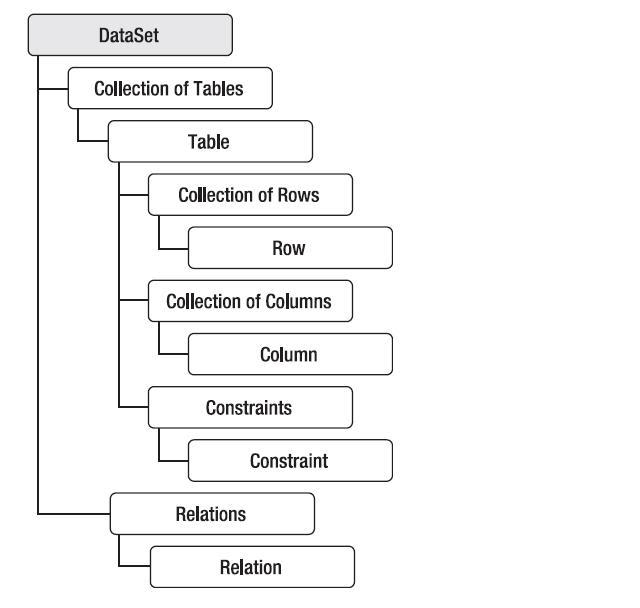
LINQ to SQL hỗ trợ đầy đủ transaction, view và các stored procedure (SP). Nó cũng cung cấp một cách dễ dàng để thêm khả năng kiểm tra tính hợp lệ của dữ liệu và các quy tắc vào trong mô hình dữ liệu của bạn.

##### *LINQ to XML*

Sử dụng LINQ với mục đích truy vấn file XML và truy vấn tiện lợi hơn nhiều so với việc dùng XmlDocument, Xpath và Xquery như trước kia.

##### *LINQ to Datasets*

DataSet trong ADO.NET là một bước phát triển lớn trong việc phát triển ứng dụng cơ sở dữ liệu đa hệ. Khi lấy và chỉnh sửa dữ liệu, duy trì liên tục kết nối tới Data Source trong khi chờ user yêu cầu thì rõ ràng là tốn tài nguyên máy rất nhiều. DataSet giúp ích ở đây rất lớn. Vì DataSet cho phép lưu trữ dữ liệu và chỉnh sửa tại ‘local cache’, hay gọi là offline mode. Có thể xem xét và xử lý thông tin trong khi ngắt kết nối. Sau khi chỉnh sửa và xem xong thì tạo một kết nối và update dữ liệu từ local vào Data Source.

Dữ liệu trong DataSet được lưu trữ dưới dạng một Collection các Tables và bạn cần phải xử lý thông qua các lớp DataTable (DataRow và DataColumn).

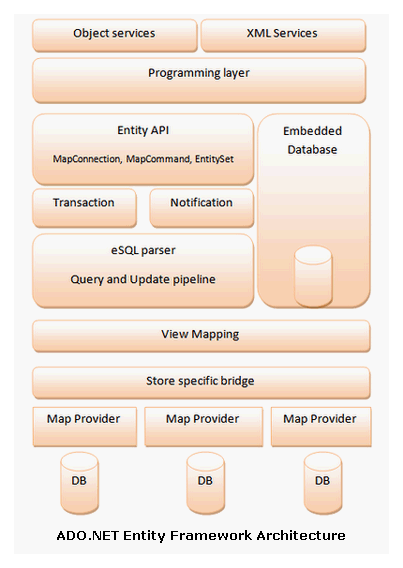
###### Hình 1.3: Các thành phần chính của mô hình MVC

**LINQ to DataSet** cho phép người lập trình sử dụng DataSets như một nguồn dữ liệu bình thường bằng các cú pháp truy vấn căn bản của LINQ

##### *LINQ to Entities*

**Khái Niệm về Entity Framework**

Entity Framework như giản đồ của các dữ liệu được lưu trữ trong cơ sở dữ liệu và trình bày giản đồ khái niệm ứng dụng của nó. Giản đồ này được ánh xạ từ cơ sỡ dữ liệu.

Entity Framework là cơ sở dữ liệu độc lập và xây dựng dựa trên một mô hình chuẩn ADO.NET cho phép truy cập vào cơ sở dữ liệu của bên thứ ba thông qua việc sử dụng ADO.NET

###### Hình 1.4: Kiến trúc của Entity Framework

Kiến trúc của Entity Framework

LINQ to Entities cho phép chúng ta có thể truy vấn các thực thể bên trong Entity Framework.

Các Operator chuẩn trong LINQ

1.Select / SelectMany

2.Where

3.Sum / Min / Max / Average / Aggregate

4.Join / GroupJoin

5.Take / TakeWhile

6.Skip / SkipWhile

7.OfType

8.Concat

9.OrderBy / ThenBy

10.Reverse

11.GroupBy

12.Distinct

13.Union / Intersect / Except

14.EqualAll

15.First / FirstOrDefault / Last / LastOrDefault

16.Single

17.ElementAt

18.Any / All / Contains

19.Count

##### *LINQ to ADO.NET*

Nhều người vẫn hay thắc mắc về mối quan hệ giữa LINQ vs ADO.NET, liệu chúng có đối lập nhau hay hỗ trợ lẫn nhau ?

LINQ là tập mở rộng cho phép viết các câu truy vấn ngay trong các ngôn ngữ lập trình. Nó cho phép bạn làm việc với các kiểu tập hợp dữ liệu, như XML, collection, array,… và cả CSDL.

ADO.NET là công nghệ cho phép các ứng dụng có thể kết nối và làm việc với các loại CSDL khác nhau (truy vấn, cập nhật, thêm, xóa, gọi thủ tục…).

Bản thân LINQ không phải là một công nghệ được tạo ra để thay thế ADO.NET, bạn có thể làm việc với LINQ mà không cần đến cơ sở dữ liệu. Tuy nhiên, LINQ to SQL, là một phần mở rộng của LINQ, cho phép bạn có thể làm việc được với CSDL SQL Server, trong trường hợp này thì khi viết bạn có thể bỏ qua các câu lệnhADO.NET mà chỉ quan tâm tới cú pháp mà LINQ cung cấp

# Phân tích thiết kế hệ thống

## Mô hình hóa chức năng

### *Biểu đồ use case*





###### Hình 2.1: Biểu đồ Usecase

#### Các use case chính

###### Hình 2.2: Các usecase chính

1. **Xem chi tiết mặt hàng**: Cho phép khách hàng xem thông tin chi tiết về các mặt hàng.

2. **Xem mặt hàng theo loại mặt hàng**: Cho phép khách hàng xem thông tin các loại mặt hàng và các nhóm mặt hàng có trong từng loại mặt hàng.

3. **Xem tư vấn mặt hàng**: Cho phép khách hàng đọc các bài tư vấn trong tư vấn mặt hàng.

4. **Xem mặt hàng khuyến mại**: Cho phép khách hàng xem các mặt hàng được giảm giá bán với giá rẻ.

5. **Thêm vào giỏ hàng**: Cho phép khách hàng thêm hàng vào giỏ hàng.

6. **Xem Quản lý giỏ hàng**: Cho phép khách hàng xóa một mặt hàng trong giỏ, sửa số lượng hàng trong giỏ, xóa toàn bộ hàng trong giỏ hàng.

7. **Xem đơn hàng**: Cho phép khách hàng xem các đơn hàng đã đặt.

8. **Thanh toán**: Cho phép khách hàng thanh toán các hàng có trong giỏ hàng.

9. **Đăng ký**: Cho phép khách hàng đăng ký tài khoản thành viên.

10. **Đăng nhập**: Cho phép khách hàng đăng nhập tài khoản thành viên.

#### Các use case thứ cấp



###### Hình 2.3: Các usecase thứ cấp

* + 1. **Bảo trì loại mặt hàng**: Cho phép admin xem, thêm, sửa, xóa thông tin trong bảng LOAIMATHANG.
    2. **Bảo trì nhóm mặt hàng**: Cho phép admin xem, thêm, sửa, xóa thông tin trong bảng NHOMMATHANG.
    3. **Bảo trì mặt hàng**: Cho phép admin xem, thêm, sửa, xóa thông tin trong bảng MATHANG.
    4. **Bảo trì tư vấn**: Cho phép admin xem, thêm, sửa, xóa thông tin trong bảng TUVANMATHANG.
    5. **Bảo trì khuyến mại**: Cho phép admin xem, thêm, sửa, xóa thông tin trong bảng KHUYENMAI.
    6. **Quản lý tài khoản**: Cho phép admin xem, thêm, sửa, xóa thông tin trong bảng NHANVIEN
    7. **Quản lý đơn hàng (xem đơn hàng, xem chi tiết đơn hàng, sửa trạng thái đơn hàng, xóa đơn hàng)**: Cho phép admin xem thông tin trong bảng DONHANG, thêm, sửa trang thái đơn hàng trong bảng DONHANG, xóa thông tin trong bảng DONHANG.

#### Quan hệ giữa các use case

###### Hình 2.4: Quan hệ giữa các usecase

### *Mô tả chi tiết các use case*

#### *Mô tả use case Đăng nhập*

|  |
| --- |
| 1. Tên Use Case  Đăng nhập  2. Mô tả vắn tắt  Use case này cho phép người dùng đăng nhập vào tài khoản hệ thống đã được đăng kí.  3. Luồng các sự kiện  3.1.Luồng cơ bản   1. Use case này bắt đầu khi người dùng bấm vào nút “Đăng nhập” trong menu. Hệ thống sẽ hiển thị lên màn hình đăng nhập. Người dùng sẽ nhập thông tin đăng nhập gồm tên tài khoản và mật khẩu sau đó nhấn nút Đăng nhập. 2. Hệ thống sẽ truy cập vào bảng KHACHHANG để kiểm tra thông tin đăng nhập của người dùng, sau đó hiển thị ra màn hình kết quả đăng nhập. Use case kết thúc.   3.2. Các luồng rẽ nhánh   1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. 2. Tại bước 3 trong luồng cơ bản nếu thông tin đăng nhập không chính xác thì hệ thống sẽ báo lỗi và use case kết thúc.   4. Các yêu cầu đặc biệt  Không có.  5. Tiền điều kiện  Người dùng có tài khoản trong hệ thống.  6. Hậu điều kiện  Use case thành công thì người dùng có thể truy cập và sử dụng các chức năng hệ thống. Nếu không thành công thì trạng thái hệ thống giữ nguyên.  7. Điểm mở rộng  Không có. |

#### *Mô tả use case Đăng ký*

|  |
| --- |
| 1. Tên Use Case  Đăng Ký  2. Mô tả vắn tắt  Use case này cho phép người dùng đăng ký một tài khoản mới  3. Luồng các sự kiện  3.1.Luồng cơ bản   1. Use case này bắt đầu khi người dùng bấm vào nút “Đăng ký” trong menu. Hệ thống sẽ hiển thị lên màn hình đăng ký. Người dùng sẽ nhập thông tin đăng ký gồm họ tên, số điện thoại, địa chỉ, email, tên tài khoản và mật khẩu sau đó nhấn nút Đăng nhập. 2. Hệ thống sẽ thêm thông tin mới vào bảng KHACHHANG, sau đó hiển thị ra màn hình kết quả đăng nhập. Use case kết thúc.   3.2. Các luồng rẽ nhánh   1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. 2. Tại bước 3 trong luồng cơ bản nếu thiếu tài khoản hoặc mật khẩu thì hệ thống sẽ báo lỗi và use case kết thúc.   4. Các yêu cầu đặc biệt  Không có.  5. Tiền điều kiện  Không có.  6. Hậu điều kiện  Use case thành công thì người dùng có thể truy cập và sử dụng các chức năng hệ thống. Nếu không thành công thì trạng thái hệ thống giữ nguyên.  7. Điểm mở rộng  Không có. |

#### *Mô tả use case Bảo trì loại mặt hàng*

|  |
| --- |
| 1. Tên Use Case  Bảo trì loại mặt hàng.  2. Mô tả vắn tắt  Use case cho phép admin xem thêm, sửa và xóa các loại mặt hàng trong bảng.  3. Luồng các sự kiện  3.1.Luồng cơ bản   1. Use case này bắt đầu khi admin bấm vào nút “loại mặt hàng” trên thanh menu quản trị. Hệ thống sẽ lấy thông tin về các loại mặt hàng gồm: mã loại, tên loại và hiển thị lên màn hình. 2. Thêm loại mặt hàng:    * + 1. Admin bấm vào nút “Thêm loại mặt hàng” trên cửa sổ danh sách loại mặt hàng. Hệ thống hiển thị màn hình yêu cầu nhập thông tin cho loại mặt hàng gồm: mã loại, tên loại.        2. Admin nhập thông tin của loại mặt hàng gồm: tên loại và kích nút “Thêm”. Hệ thống sẽ sinh một mã loại mới, tạo loại mặt hàng trong bảng LOAIMATHANG và hiển thị danh sách loại mặt hàng đã cập nhật. 3. Sửa loại mặt hàng: 4. Admin kích vào nút “sửa” trên một loại mặt hàng. Hệ thống sẽ lấy thông tin cũ của loại mặt hàng gồm: mã loại, tên loại và hiển thị lên màn hình. 5. Admin cập nhật thông tin loại mặt hàng gồm: mã loại, tên loại và bấm nút “cập nhật”. Hệ thống sẽ sửa thông tin loại mặt hàng được chọn trong bảng LOAIMATHANG và hiển thị danh sách các loại mặt hàng đã được cập nhật. 6. Xóa loại mặt hàng: 7. Admin bấm vào nút “xóa” trên một loại mạt hàng. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa. 8. Người quản trị bấm vào nút “xác nhận” .Hệ thống sẽ xóa loại mặt hàng được chọn từ bảng LOAIMATHANG và hiển thị danh sách các loại mặt hàng được cập nhật. Use case kết thúc.   3.2. Các luồng rẽ nhánh   1. Tại bước 2b hoặc 3b trong luồng cơ bản, nếu admin nhập thông tin không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại, admin có thể nhập lại thông tin hoặc chọn nút “hủy bỏ” để thoát. 2. Tại bước 2b, 3b hoặc 4b trong luồng cơ bản, nếu admin bấm nút “hủy” thì hệ thống sẽ bỏ thao tác thêm, sửa hoặc xóa và use case kết thúc. 3. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.   4. Các yêu cầu đặc biệt  Use case chỉ cho phép admin thực hiện.  5. Tiền điều kiện  Admin cần đăng nhập hệ thống để thực hiện use case.  6. Hậu điều kiện  Thông tin phải cập nhật trong cơ sở dữ liệu khi use case thành công.  7. Điểm mở rộng  Không có. |

#### *Mô tả use case Xem mặt hàng theo loại mặt hàng*

|  |
| --- |
| 1. Tên Use Case  Xem mặt hàng theo loại mặt hàng  2. Mô tả vắn tắt  Use case này cho phép khách hàng xem thông tin loại mặt hàng và các nhóm mặt hàng có trong từng loại mặt hàng.  3. Luồng các sự kiện  3.1. Luồng cơ bản   1. Use case này bắt đầu khi khách hàng kích vào: “Các mặt hàng” trên thanh menu. Hệ thống hiện lên một submenu với tên loại mặt hàng trong bảng LOAIMATHANG, và danh sách nhóm mặt hàng của từng loại mặt hàng gồm có tên nhóm mặt hàng trong bảng NHOMMATHANG và hiển thị lên màn hình. 2. Khách hàng kích vào tên một nhóm mặt hàng trong danh sách nhóm mặt hàng của từng loại mặt hàng. Hệ thống sẽ lấy thông tin về danh sách mặt hàng có trong nhóm mặt hàng được chọn gồm có tên mặt hàng, giá bán, mô tả, thương hiệu, thành phần từ bảng MATHANG, ảnh minh họa từ bảng HINHANH. Use case kết thúc.   3.2. Các luồng rẽ nhánh   1. Tại bước 1 trong luông cơ bản, nếu trong bảng LOAIMATHANG chưa có dữ liệu, thì hệ thống sẽ hiển thị một thông báo: “Chưa có loại mặt hàng nào!” và use case kết thúc. 2. Tại bước 2 trong luồng cơ bản, nếu không tìm thấy MATHANG thuộc nhóm mặt hàng được chọn từ bảng MATHANG, thì hệ thống sẽ hiển thị một thông báo: “Không tìm thấy mặt hàng nào nhóm mặt hàng này!” và use case kết thúc. 3. ) Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.   4. Các yêu cầu đặc biệt  Không có  5. Tiền điều kiện  Không có  6. Hậu điều kiện  Không có  7. Điểm mở rộng  Không có. |

#### *Mô tả use case Bảo trì khuyến mại*

|  |  |
| --- | --- |
| 1. Tên Use Case  Bảo trì khuyến mại.  2. Mô tả vắn tắt  Use case này cho phép admin xem, thêm, sửa, xóa thông tin trong bảng KHUYENMAI.  3. Luồng các sự kiện  3.1. Luồng cơ bản  1) Use case này bắt đầu khi admin kích vào nút “Khuyến mại” trên menu quản trị. Hệ thống lấy thông tin chi tiết của các khuyến mại gồm: mã khuyến mại, giảm giá, ngày bắt đầu, ngày kết thúc từ bảng KHUYENMAI trong cơ sở dữ liệu và hiển thị danh sách khuyến mại lên màn hình.  2) Thêm khuyến mại:  a) Admin bấm vào nút “Thêm khuyến mại” trên cửa sổ danh sách khuyến mại. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho khuyến mại gồm : mã khuyến mại, giảm giá, ngày bắt đầu, ngày kết thúc.  b) Admin nhập thông tin của giảm giá, ngày bắt đầu, ngày kết thúc và kích vào nút “Tạo”. Hệ thống sẽ sinh một mã khuyến mại mới, tạo một khuyến mại trong bảng KHUYENMAI và hiển thị danh sách các khuyến mại đã được cập nhật.   1. Sửa khuyến mại:   a) Admin kích vào nút “Sửa” trên một dòng khuyến mại. Hệ thống sẽ lấy thông tin cũ của khuyến mại được chọn gồm: : mã khuyến mại, giảm giá, ngày bắt đầu, ngày kết thúc từ bảng KHUYENMAI và hiển thị lên màn hình.  b) Admin nhập thông tin mới giảm giá, ngày bắt đầu, ngày kết thúc. Hệ thống sẽ sửa thông tin của khuyến mại được chọn trong bảng KHUYENMAI và hiển thị danh sách các khuyến mại được cập nhật.   1. Xóa khuyến mại:   a) Admin kích vào nút “Xóa” trên một dòng khuyến mại. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa.  b) Admin kích vào nút “Đồng ý”. Hệ thống sẽ xóa khuyến mại được chọn khỏi bảng KHUYENMAI và hiển thị danh sách các khuyến mại đã cập nhật.  Use case kết thúc.  3.2. Các luồng rẽ nhánh  1) Tại bước 2b hoặc 3b trong luồng cơ bản, nếu admin nhập thông tin không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại. Admin có thể nhập lại thông tin để tiếp tục hoặc kích vào nút “Hủy bỏ” để kết thúc..  2) Tại bước 2b, 3b hoặc 4b trong luồng cơ bản nếu admin kích vào nút “Hủy bỏ” hệ thống sẽ bỏ qua thao tác thêm mới, sửa chữa hoặc xóa tương ứng và hiển thị danh sách các khuyến mại trong bảng KHUYENMAI.  3) Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.  4. Các yêu cầu đặc biệt  Use case chỉ cho phép admin thực hiện.  5. Tiền điều kiện  Admin cần đăng nhập hệ thống để thực hiện use case.  6. Hậu điều kiện  Nếu use case kết thúc thành công thì thông tin về cửa hàng sẽ được cập nhật trong cơ sở dữ liệu.  7. Điểm mở rộng  Không có. |  |

#### *Mô tả use case Bảo trì mặt hàng*

|  |  |
| --- | --- |
| 1. Tên use case: Bảo trì mặt hàng.  2. Mô tả vắn tắt:  Use case này cho phép người quản trị thêm, sửa, xóa thông tin trong bảng MATHANG.  3. Luồng các sự kiện  3.1. Luồng cơ bản  1) Use case này bắt đầu khi người quản trị kích vào nút “Mặt hàng” trên menu quản trị. Hệ thống lấy thông tin chi tiết của các mặt hàng bao gồm: mã mặt hàng, tên mặt hàng, giá bán, mô tả, thương hiệu, thành phần từ bảng MATHANG trong cơ sở dữ liệu và hiển thị lên màn hình.  2) Thêm mặt hàng  a) Admin vào nút “Thêm mới” trên cửa sổ danh sách các mặt hàng. Hệ thống hiển thị màn hình yêu cầu nhập thông tin mặt hàng mới.  b). Admin nhập thông tin mặt hàng và kích vào nút “Tạo”. Hệ thống sẽ tạo ra mặt hàng mới và thêm thông tin của mặt hàng vừa nhập trong bảng MATHANG và hiển thị ra thông tin mặt hàng vừa nhập.  3) Sửa mặt hàng  a) Admin kích vào nút “Sửa” trên một mặt hàng. Hệ thống sẽ lấy thông tin mặt hàng đó từ bảng MATHANG và hiển thị lên màn hình.  b) Admin nhập thông tin mới về mặt hàng và kích vào nút “Cập nhật”. Hệ thống sẽ sửa thông tin mặt hàng được chọn trong bảng MATHANG và hiển thị thông tin mặt hàng đã cập nhật.  4) Xóa mặt hàng  a) Admin vào nút “Xóa” trên một mặt hàng. Hệ thống sẽ hiển thị một thông báo xác nhận xóa.  b) Admin kích vào nút “Đồng ý”. Hệ thống sẽ xóa thông tin mặt hàng được chọn khỏi bảng MATHANG. Use case kết thúc.  3.2. Luồng rẽ nhánh  1) bước 2b hoặc 3b trong luồng cơ bản nếu admin nhập thông tin mặt hàng không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi yêu cầu nhập lại. Người quản trị có thể nhập lại để tiếp tục hoặc kích vào nút “Hủy bỏ” để kết thúc.  2) Tại bước 2b hoặc 3b trong luồng cơ bản nếu admin kích vào nút “Hủy bỏ” hệ thống sẽ bỏ qua thao tác thêm mới hoặc sửa chữa tương ứng và hiển thị danh sách các mặt hàng trong bảng MATHANG.  3) Tại bước 4b trong luồng cơ bản nếu admin kích vào nút “Không đồng ý” hệ thống sẽ bỏ qua thao tác xóa và hiển thị thông tin mặt hàng bảng MATHANG.  4) Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sử dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.  4. Các yêu cầu đặc biệt  Use case này chỉ cho phép admin thực hiện.  5. Tiền điều kiện  Admin cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case.  6. Hậu điều kiện  Nếu use case kết thúc thành công thì thông tin về mặt hàng sẽ được cập nhật trong cơ sở dữ liệu.  7. Điểm mở rộng  Không có. |  |

#### *Mô tả use case Xem chi tiết mặt hàng*

|  |
| --- |
| 1. Tên Use Case: Xem chi tiết mặt hàng.  2. Mô tả vắn tắt:  Use case này cho phép khách hàng xem thông tin chi tiết về các mặt hàng.  3. Luồng các sự kiện  3.1. Luồng cơ bản  1) Use case này bắt đầu khi khách hàng kích vào hình ảnh minh họa, hoặc tên của một mặt hàng. Hệ thống lấy thông tin về mặt hàng từ bảng MATHANG và hiển thị mã mặt hàng, tên mặt hàng, ảnh minh họa, xuất xứ, thành phần dinh dưỡng kích thước và giá từ bảng MATHANG. Use case kết thúc.  3.2. Các luồng rẽ nhánh  1) Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.  2) Trong luồng cơ bản, nếu không tìm thấy các thông tin chi tiết về mặt hàng được lấy từ bảng MATHANG, thì hệ thống sẽ hiển thị thông báo: “Không tìm thấy thông tin về mặt hàng này!” và use case kết thúc.  4. Các yêu cầu đặc biệt  Không có.  5. Tiền điều kiện  Không có.  6. Hậu điều kiện  Không có.  7. Điểm mở rộng  Không có. |

#### *Mô tả use case Thêm vào giỏ hàng*

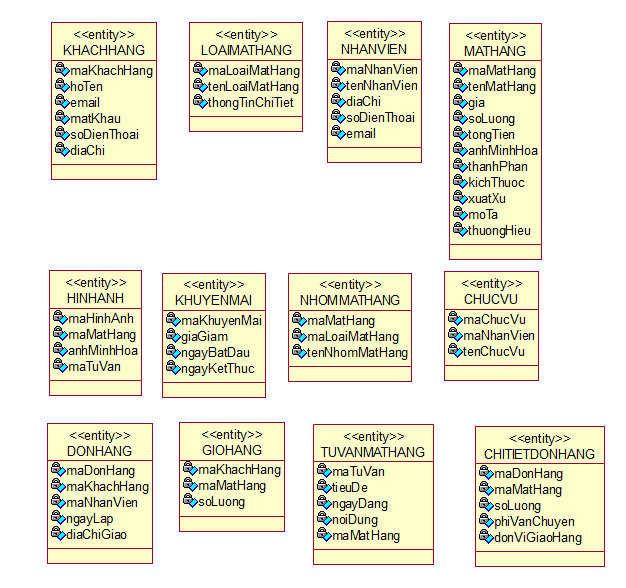
|  |
| --- |
| 11. Tên Use Case  Thêm vào giỏ hàng  2. Mô tả vắn tắt  Use case này cho phép khách hàng thêm những mặt hàng mà mình  muốn mua vào giỏ hàng của mình để tiến hành đặt hàng.  3. Luồng các sự kiện  3.1. Luồng cơ bản   1. Use case này bắt đầu khi khách hàng đã tìm kiếm được mặt hàng mà mình muốn mua bằng cách tìm kiếm (nhập từ khóa mặt hàng cần tìm) trên trang web hoặc từ danh sách mặt hàng trên Trang Chủ. 2. Khách hàng kích vào mặt hàng mà mình muốn mua bằng cách kích vào “biểu tượng giỏ hàng”. Hệ thống hiển thị thông chi tiết mặt hàng bao gồm mã mặt hàng, tên mặt hàng, giá, số lượng và tổng thành tiền. Khách hàng kiểm tra thông tin của mặt hàng. 3. Khách hàng chọn số lượng muốn mua và thêm vào giỏ hàng 4. Hệ thống hiển thị màn hình thông báo về việc thêm mặt hàng vào giỏ hàng và tổng số tiền thanh toán.   3.2. Các luồng rẽ nhánh   1. Tại bước 2 trong luồng cơ bản, khi khách hàng kích vào “biểu tượng giỏ hàng” nhưng mặt hàng đã hết hàng hoặc không khả dụng, hệ thống sẽ thông báo hết hàng và không cho phép người dùng thêm vào giỏ hàng. Use case kết thúc. 2. Tại bước 3 nếu chọn số lượng muốn mua quá với số lượng tối đa cho phép của một mặt hàng hệ thống sẽ thông báo lỗi và trở về số lượng là 1. Use case kết thúc. 3. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một số thông báo lỗi và use case kết thúc. 4. Các yêu cầu đặc biệt   4. Các yêu cầu đặc biệt  Không có  5. Tiền điều kiện  Không có  6. Hậu điều kiện  Không có  7. Điểm mở rộng  Không có. |

#### *Mô tả use case Quản lí tài khoản*

|  |  |
| --- | --- |
| 1. Tên Use Case  Quản lý tài khoản  2. Mô tả vắn tắt  Use case này cho phép admin xem, sửa, thêm, xóa dữ liệu trong bảng NHANVIEN  3. Luồng các sự kiện  3.1. Luồng cơ bản  1) Use case bắt đầu khi Admin kích vào “Tài khoản” trên menu quản trị. Hệ thống lấy thông tin từ bảng NHANVIEN trong cơ sở dữ liệu và hiện thị danh sách các tài khoản lên màn hình.  2) Sửa tài khoản:  a) Admin chọn một tài khoản và kích vào nút “Sửa”. Hệ thống sẽ lấy thông tin trong bảng NHANVIEN và sau đó hiển thị lên màn hình  b) Admin sửa các thông tin muốn sửa, Admin vào nút “Lưu”. Hệ thống sẽ thay đổi thông tin tài khoản và cập nhật vào bảng NHANVIEN và hiển thị danh sách tài khoản đã cập nhật.  3) Thêm tài khoản:  a) Admin vào nút “Thêm” trên cửa sổ danh sách tài khoản. Hệ thống hiện thị màn hình yêu cầu nhập thông tin tài khoản.  b) Admin nhập thông tin tài khoản và kích vào nút “Tạo”. Hệ thống sẽ tạo một tài khoản mới trong bảng NHANVIEN và hiện thị danh sách các tài khoản đã được cập nhật.  4) Xóa tài khoản:  a) Admin chọn một tài khoản kích vào nút ”Xóa”. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa.  b) Admin vào nút ”Đồng ý”. Hệ thống sẽ xóa tài khoản được chọn khỏi bảng NHANVIEN và hiện thị danh sách tài khoản đã cập nhật lên màn hình. Use case kết thúc.  3.2. Các luồng rẽ nhánh  1) Tại bước 4b trong luồng cơ bản, nếu Admin kích vào nút “Huỷ bỏ” thì hệ thống sẽ bỏ qua thao tác xoá và hiển thị lại danh sách tài khoản.  2) Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.  4. Các yêu cầu đặc biệt  Use case này chỉ cho phép một số vai trò như Admin, chủ hệ thống thực hiện.  5. Tiền điều kiện  Admin phải đăng nhập bằng tài khoản Admin mới có thể thực hiện được use case.  6. Hậu điều kiện  Nếu use case kêt thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.  7. Điểm mở rộng  Không có. |  |

## Phân tích use case

### *Biểu đồ các lớp Entity của hệ thống*



###### Hình 2.5: Biểu đồ các lớp entity của hệ thống

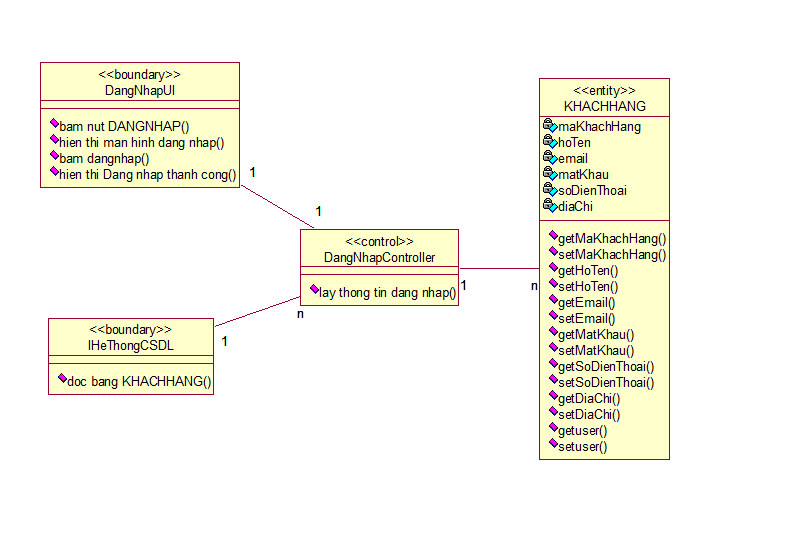
### *Phân tích các use case*

#### *Phân tích use case Đăng nhập*

##### Biểu đồ trình tự

###### Hình 2.6: Biểu đồ trình tự usecase Đăng nhập

##### Biểu đồ lớp phân tích



###### Hình 2.7: Biểu đồ lớp phân tích usecase Đăng nhập

#### *Phân tích use case Đăng ký*

##### Biểu đồ trình tự

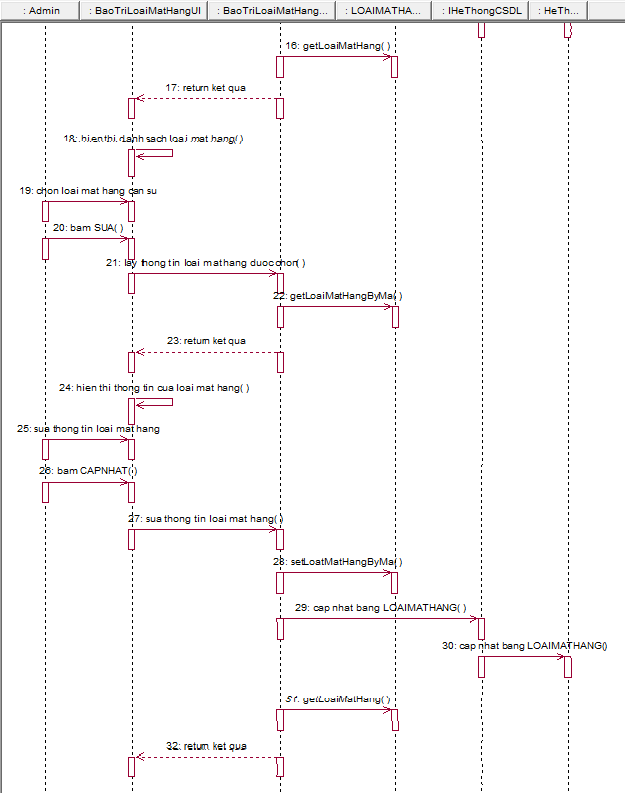
###### Hình 2.8: Biểu đồ trình tự usecase Đăng ký

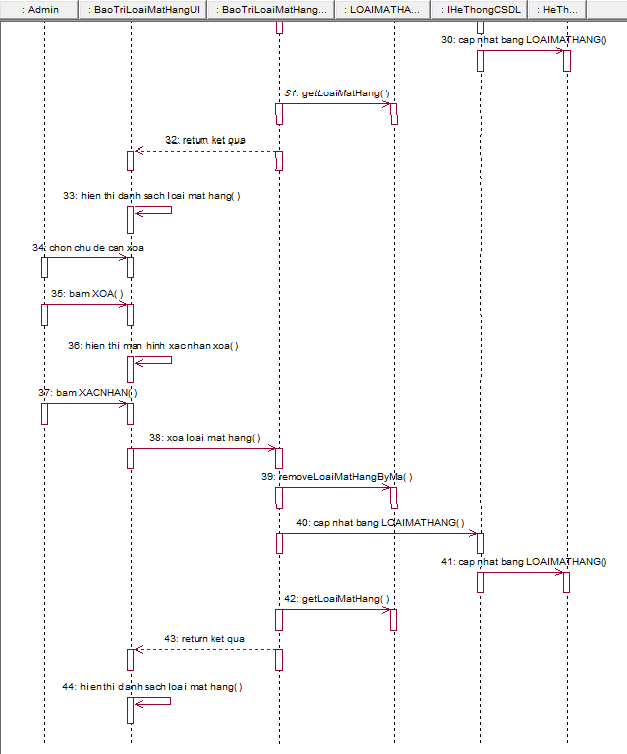
##### Biểu đồ lớp phân tích

###### Hình 2.9: Biểu đồ lớp phân tích usecase Đăng ký

#### *Phân tích use case Bảo trì loại mặt hàng*

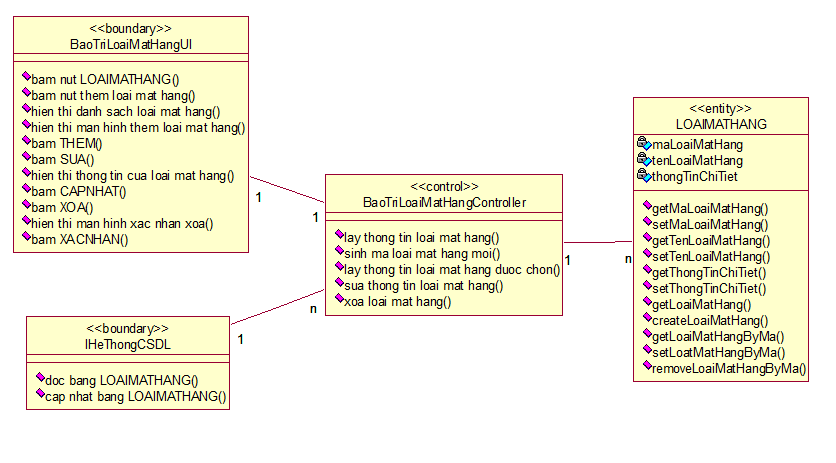
##### Biểu đồ trình tự





###### Hình 2.10: Biểu đồ trình tự usecase Bảo trì loại mặt hàng

##### Biểu đồ lớp phân tích



###### Hình 2.11: Biểu đồ lớp phân tích usecase Bảo trì loại mặt hàng

#### *Phân tích use case Xem mặt hàng theo loại mặt hàng*

##### Biểu đồ trình tự

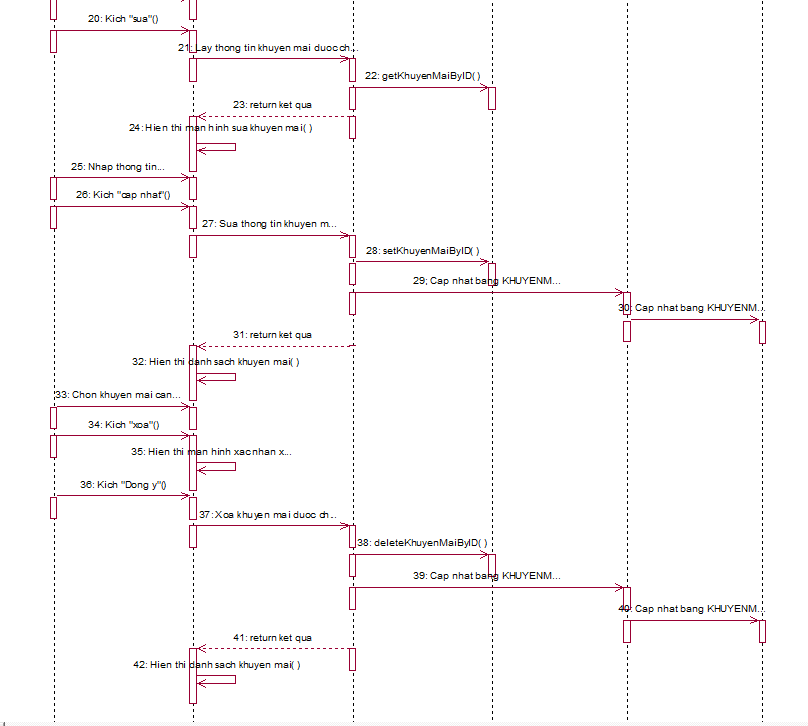
###### Hình 2.12: Biểu đồ trình tự usecase Xem mặt hàng theo loại mặt hàng

##### Biểu đồ lớp phân tích

###### Hình 2.13: Biểu đồ lớp phân tích usecase Xem mặt hàng theo loại mặt hàng

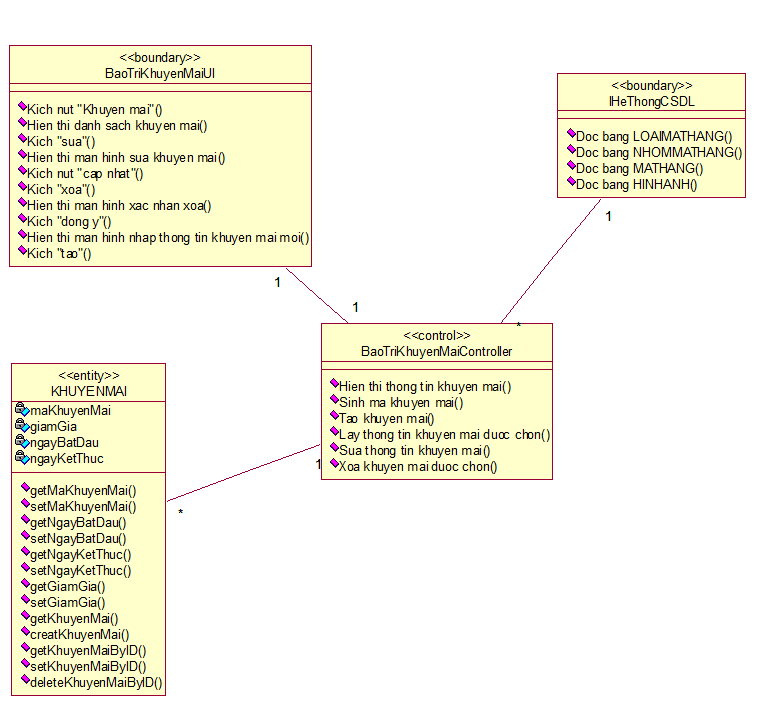
#### *Phân tích usecase Bảo trì khuyến mại*

##### Biểu đồ trình tự



###### Hình 2.14: Biểu đồ trinh tự usecase Bảo trì khuyến mại

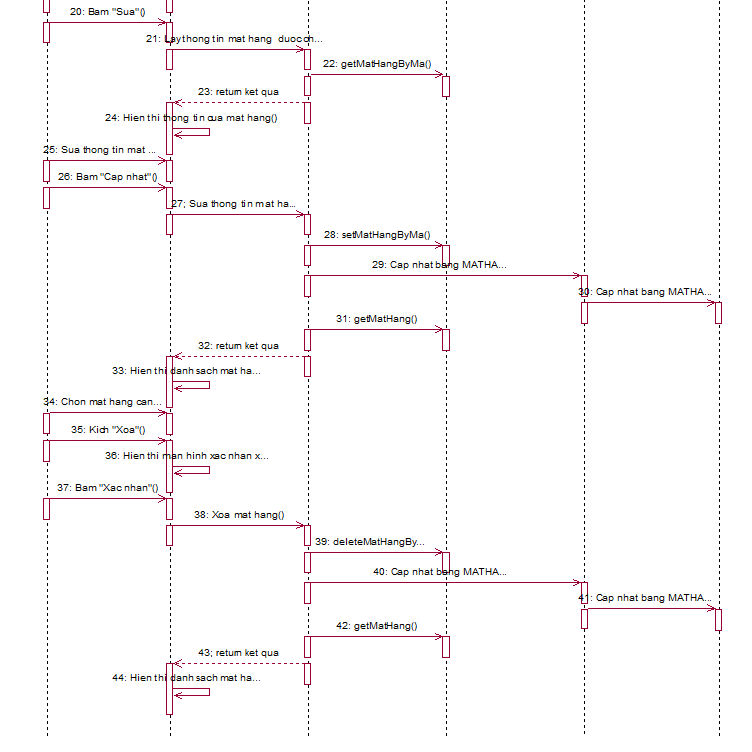
##### Biểu đồ lớp phân tích



###### Hình 2.15: Biểu đồ lớp phân tích usecase Bảo trì khuyến mại

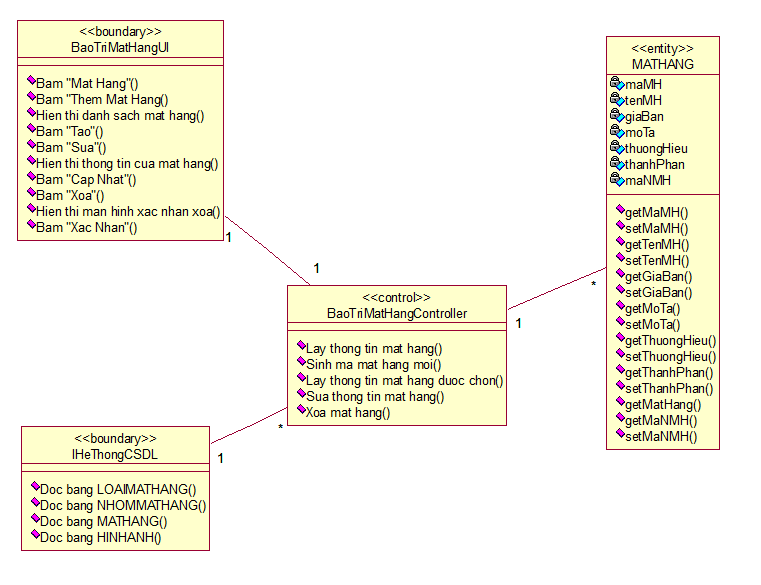
#### *Phân tích use case Bảo trì mặt hàng*

##### Biểu đồ trình tự



###### Hình 2.16: Biểu đồ trinh tự usecase Bảo trì mặt hàng

##### Biểu đồ lớp phân tích



###### Hình 2.17: Biểu đồ lớp phân tích usecase Bảo trì mặt hàng

#### *Phân tích use case Xem chi tiết mặt hàng*

##### Biểu đồ trình tự

###### Hình 2.18: Biểu đồ trình tự usecase Xem chi tiết mặt hàng

##### Biểu đồ lớp phân tích

###### Hình 2.19: Biểu đồ lớp phân tích usecase Xem chi tiết mặt hàng

#### *Phân tích use case Thêm vào giỏ hàng*

##### Biểu đồ trình tự

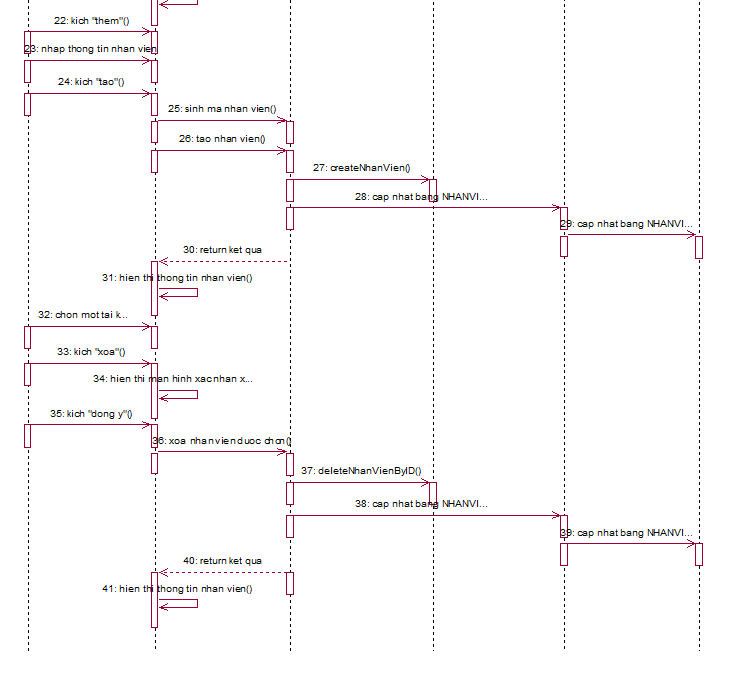
###### Hình 2.20: Biểu đồ trình tự usecase Thêm vào giỏ hàng

##### Biểu đồ lớp phân tích

###### Hình 2.21: Biểu đồ lớp phân tích usecase Thêm vào giỏ hàng

#### *Phân tích use case Quản lý tài khoản*

##### Biểu đồ trình tự



###### Hình 2.22: Biểu đồ trình tự usecase Quản lý tài khoản

##### Biểu đồ lớp phân tích

###### Hình 2.23: Biểu đồ lớp phân tích usecase Quản lý tài khoản

### *Biểu đồ các lớp phân tích*

#### *Nhóm use case chính*

###### Hình 2.24: Biểu đồ các lớp phân tích nhóm usecase chính

#### *Nhóm use case thứ cấp*

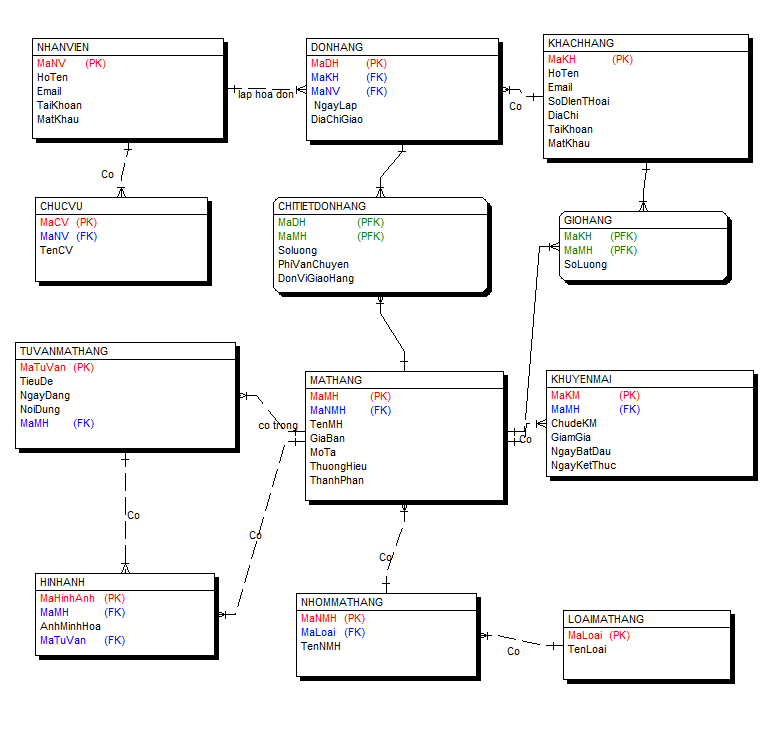
###### Hình 2.25: Biểu đồ các lớp phân tích nhóm usecase thứ cấp

## Mô hình hóa dữ liệu

### *Các yêu cầu về dữ liệu*

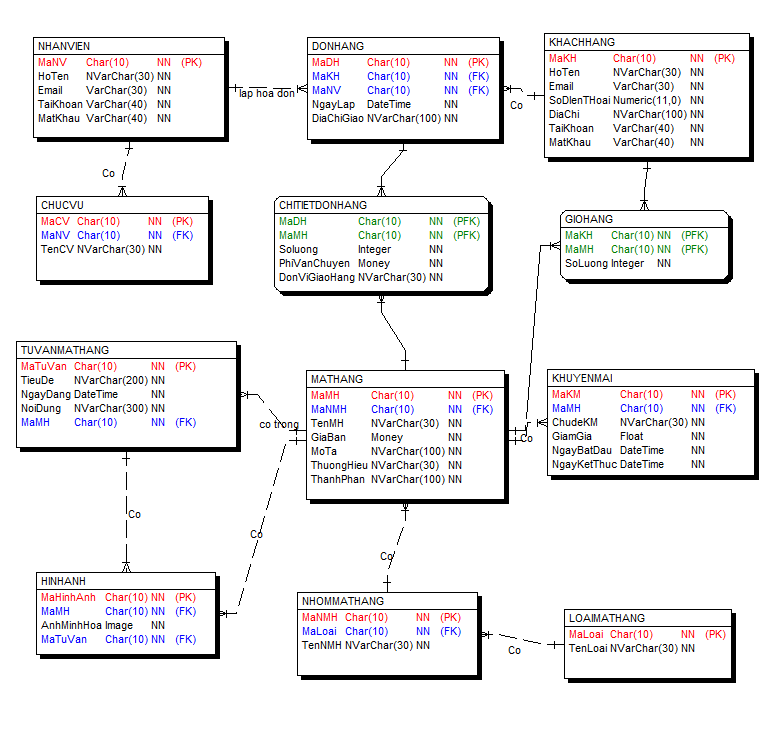
* Loại mặt hàng: gồm mã loại, tên loại. Trong 1 loại mặt hàng có 0,1 hoặc nhiều nhóm mặt hàng, mỗi nhóm mặt hàng cần thuộc về 1 loại mặt hàng.
* Nhóm mặt hàng: gồm mã nhóm mặt hàng, tên nhóm mặt hàng. Trong 1 nhóm mặt hàng có 0, 1 hoặc nhiều mặt hàng. Mỗi mặt hàng cần thuộc về 1 nhóm mặt hàng.
* Mặt hàng: gồm mã mặt hàng, tên mặt hàng, giá bán, mô tả, thương hiệu, thành phần. Mỗi mặt hàng có 1 hoặc nhiều hình ảnh, mỗi hình ảnh thuộc về 1 mặt hàng. Mỗi mặt hàng thuộc 1 hoặc nhiều tư vấn mặt hàng, mỗi tư vẫn mặt hàng cần thuộc về 1 mặt hàng. Mỗi mặt hàng có 1 hoặc nhiều khuyến mại, mỗi khuyến mại cần thuộc về 1 mặt hàng.
* Tư vấn tranh: gồm mã tư vấn, tiêu đề, giới thiệu, ngày đăng và số lượt xem, nội dung. Mỗi tư vấn tranh có 1 hoặc nhiều hình ảnh minh họa, mỗi hình ảnh minh họa cần thuộc về 1 tư vấn.
* Khuyến mãi: gồm mã khuyến mãi, chủ đề khuyến mại, ngày bắt đầu, ngày kết thúc, giảm giá.
* Hình ảnh: gồm mã hình ảnh, ảnh minh họa.
* Đơn hàng: gồm mã đơn hàng, ngày lập, địa chỉ giao hàng. Mỗi đơn hàng có 1 hoặc nhiều mặt hàng. Mỗi mặt hàng có trong 1 hoặc nhiều đơn hàng.
* Khách hàng: gồm mã khách hàng, họ tên, email,tài khoản, mật khẩu, số điện thoại, địa chỉ. Mỗi khách hàng có thể mua 1 hoặc nhiều mặt hàng. Mỗi mặt hàng được mua bới 1 hoặc nhiều khách hàng. Mỗi khách hàng có 1 hoặc nhiều đơn hàng, mỗi đơn hàng cần thuộc về 1 khách hàng.
* Nhân viên: gồm mã nhân viên, họ tên, tài khoản, mật khẩu, email. Mỗi nhân viên lập 1 hoặc nhiều hóa đơn. Mỗi hóa đơn cần được lập bởi 1 nhân viên. Mỗi nhân viên có 1 hoặc nhiều chức vụ.
* Chức vụ: gồm mã chức vụ, tên chức vụ

### *Biểu đồ thực thể liên kết mức logic*



###### Hình 2.26: Biểu đồ thực thể liên kết mức logic

### *Biểu đồ thực thể liên kết mức vật lý*

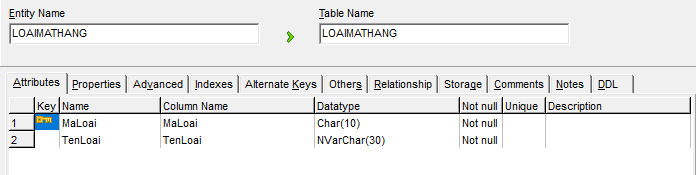


###### Hình 2.27: Biểu đồ thực thể liên kết mức vật lý

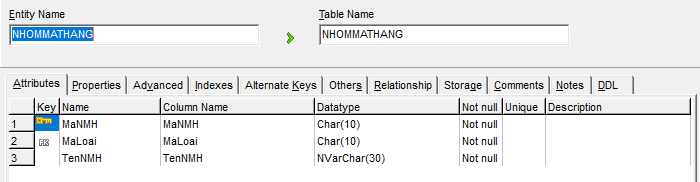
### *Mô hình cơ sở dữ liệu mức vật lý*

#### *Thiết kế bảng*

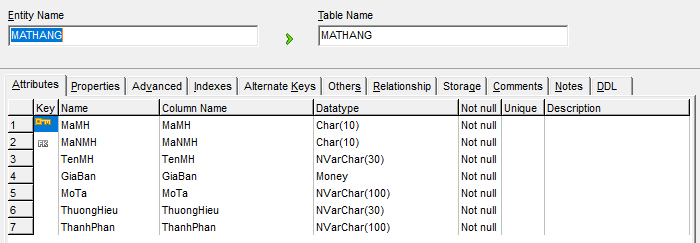
Bảng 2.1: Loại mặt hàng



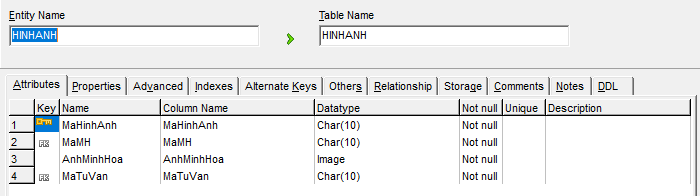
Bảng 2.2: Nhóm mặt hàng



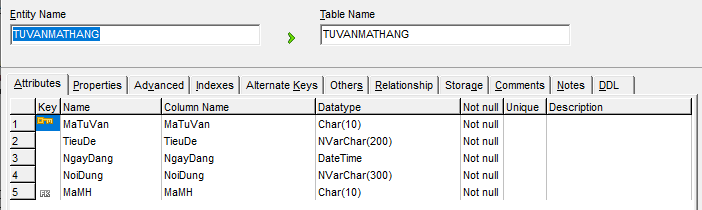
Bảng 2.3: Mặt hàng



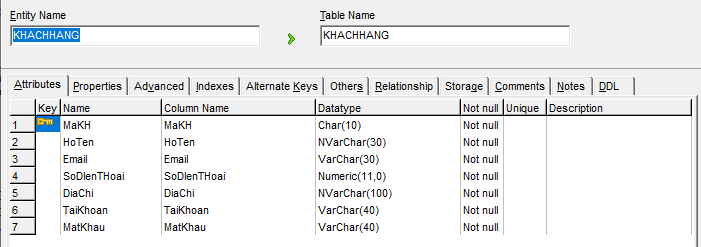
Bảng 2.4: Hình ảnh



Bảng 2.5: Tư vấn mặt hàng



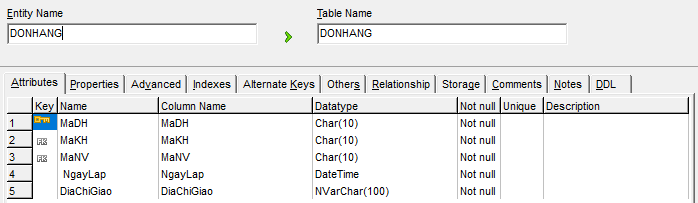
Bảng 2.6: Khách hàng



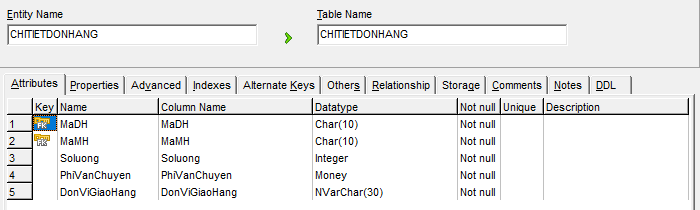
Bảng 2.7: Giỏ hàng



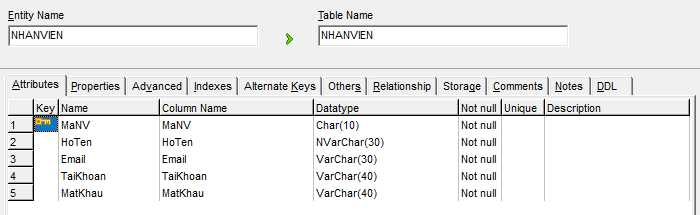
Bảng 2.8: Đơn hàng



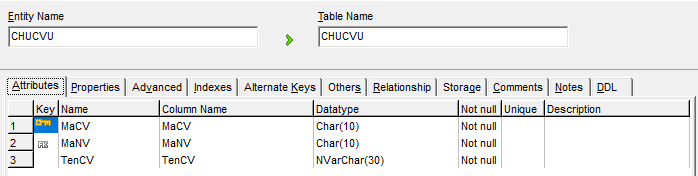
Bảng 2.9: Chi tiết đơn hàng



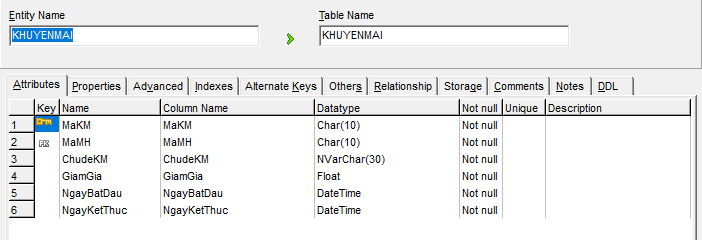
Bảng 2.10: Nhân viên



Bảng 2.11: Chức vụ



Bảng 2.12: Khuyến mại



# Cài đặt và triển khai

## Công cụ và môi trường cài đặt

### *Visual Studio Code*

Visual Studio Code là một [trình soạn thảo mã nguồn](https://vi.wikipedia.org/wiki/Tr%C3%ACnh_so%E1%BA%A1n_th%E1%BA%A3o_m%C3%A3_ngu%E1%BB%93n) được phát triển bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft) dành cho [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), [Linux](https://vi.wikipedia.org/wiki/Linux) và [macOS](https://vi.wikipedia.org/wiki/MacOS). Nó hỗ trợ chức năng debug, đi kèm với [Git](https://vi.wikipedia.org/wiki/Git_(ph%E1%BA%A7n_m%E1%BB%81m)), có chức năng nổi bật cú pháp ([syntax highlighting](https://vi.wikipedia.org/w/index.php?title=Syntax_highlighting&action=edit&redlink=1)), tự hoàn thành mã thông minh, [snippets](https://vi.wikipedia.org/w/index.php?title=Snippets&action=edit&redlink=1), và [cải tiến mã nguồn](https://vi.wikipedia.org/wiki/C%E1%BA%A3i_ti%E1%BA%BFn_m%C3%A3_ngu%E1%BB%93n). Nó cũng cho phép tùy chỉnh, người dùng có thể thay đổi theme, phím tắt, và các tùy chọn khác. Nó miễn phí và là [phần mềm mã nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_t%E1%BB%B1_do_ngu%E1%BB%93n_m%E1%BB%9F) theo [giấy phép MIT](https://vi.wikipedia.org/wiki/Gi%E1%BA%A5y_ph%C3%A9p_MIT),[[4]](https://vi.wikipedia.org/wiki/Visual_Studio_Code#cite_note-4) mặc dù bản phát hành của [Microsoft](https://vi.wikipedia.org/wiki/Microsoft) là theo [giấy phép phần mềm](https://vi.wikipedia.org/wiki/Gi%E1%BA%A5y_ph%C3%A9p_ph%E1%BA%A7n_m%E1%BB%81m) miễn phí.[[5]](https://vi.wikipedia.org/wiki/Visual_Studio_Code#cite_note-5)

Visual Studio Code được dựa trên [Electron](https://en.wikipedia.org/wiki/Electron_(software_framework)), một nền tảng được sử dụng để triển khai các ứng dụng [Node.js](https://vi.wikipedia.org/wiki/Node.js) [máy tính cá nhân](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh_c%C3%A1_nh%C3%A2n) chạy trên động cơ bố trí Blink. Mặc dù nó sử dụng nền tảng Electron[[6]](https://vi.wikipedia.org/wiki/Visual_Studio_Code#cite_note-ars-electron-6) nhưng phần mềm này không phải là một bản khác của [Atom](https://vi.wikipedia.org/wiki/Atom), nó thực ra được dựa trên trình biên tập của [Visual Studio Online](https://vi.wikipedia.org/wiki/Microsoft_Visual_Studio) (tên mã là "Monaco").[[7]](https://vi.wikipedia.org/wiki/Visual_Studio_Code#cite_note-7)

Trong cuộc khảo sát vào năm 2022 trên [Stack Overflow](https://vi.wikipedia.org/wiki/Stack_Overflow), Visual Studio Code được xếp hạng là trình biên tập mã phổ biến nhất, với 74.48% của 71,010 người trả lời tuyên bố sử dụng nó.

Các bước cài đặt:

Bước 1: Chọn vào file cài đặt để bắt đầu cài đặt phần mềm Code

Bước 2: Chọn I Agree để đồng ý các điều khoản của Visual Studio Code.

Bước 3: Tích chọn những ô trong Other (hình minh họa)

Bước 4: Chọn Next lần lượt ở những bước tiếp theo.

Bước 5: Kiểm tra Add to PATH và chọn Install.

Bước 6: Phần mềm VS Code 2023 sẽ được cài đặt. Chọn Finish để kết thúc.

### *Visual Studio 2019*

Visual studio là một phần mềm hỗ trợ đắc lực hỗ trợ công việc lập trình website. Công cụ này được tạo lên và thuộc quyền sở hữu của ông lớn công nghệ Microsoft. Năm 1997, phần mềm lập trình nay có tên mã Project Boston. Nhưng sau đó, Microsoft đã kết hợp các công cụ phát triển, đóng gói thành sản phẩm duy nhất.

Visual Studio là hệ thống tập hợp tất cả những gì liên quan tới phát triển ứng dụng, bao gồm  trình chỉnh sửa mã, trình thiết kế, gỡ lỗi. Tức là, bạn có thể viết code, sửa lỗi, chỉnh sửa thiết kế ứng dụng dễ dàng chỉ với 1 phần mềm Visual Studio mà thôi.  Không dừng lại ở đó, người dùng còn có thể thiết kế giao diện, trải nghiệm trong Visual Studio như khi phát triển ứng dụng Xamarin, UWP bằng XAML hay Blend vậy.

Phần mềm Visual studio được chia thành 2 phiên bản Visual Studio Enterprise và Visual Studio Professional, các phiên bản cao cấp có tính phí này được sử dụng nhiều bởi các [công ty chuyên về lập trình](https://mona.software/). Bên cạnh đó, Microsoft cũng cho ra mắt phiên bản Community (phiên bản miễn phí) của gói phần mềm, cung cấp cho người dùng những tính năng cơ bản nhất, phù hợp với các đối tượng lập trình không chuyên, mới tiếp cận tìm hiểu về công nghệ (đối tượng nghiên cứu, nhà phát triển cá nhân, hỗ trợ dự án mỡ, các tổ chức phi doanh nghiệp dưới 5 người dùng).

Các bước cài đặt:

**Bước 1:** vào trang [Download Visual Studio](https://visualstudio.microsoft.com/downloads/). Ở đây mình **chọn bản Community**.

**Bước 2:** **nhấn đúp chuột** vào file vừa mới tải về, sau đó chọn **Run** để bắt đầu cài đặt Visual Studio.

**Bước 3:** Chọn **Continue** và chờ vài phút để tải file cài đặt về.

**Bước 4:** chọn **các mục, các ngôn ngữ lập trình** mà cần sử dụng. Ở đây mình chọn mục ASP.NET and web development

**Bước 5:** chọn **vị trí lưu file** của Visual Studio. Sau đó chọn **Install** để bắt đầu quá trình tải và cài đặt.

**Bước 6:** Chờ Visual Studio tải và cài đặt các gói tài nguyên.**chọn Start after installation** để khởi động Visual Studio sau khi tải xong

### *Microsoft SQL server*

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS) ) sử dụng câu lệnh SQL (Transact-SQL) để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server có thể kết hợp “ăn ý” với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server….

Các bước cài đặt:

**Bước 1:** Truy cập trang chủ để tải file cài đặt SQL Server 2022. Click vào **download** ở phần **developer** để tải phần mềm.

**Bước 2:** Mở file đã **download**.

**Bước 3:** Sau khi mở ứng dụng có 3 lựa chọn tùy mục đích khác nhau cho người dùng:

Bước 4: chọn chế độ **Basic**, bạn hãy chọn basic và ấn **Accept** sau đó ấn nút **Install** để tiến hành cài đặt.

Cài đặt SQL Server Management Studio (SSMS):

**Bước 1:** Bạn chọn nút **Install SSMS**. Hệ thống sẽ tự động đưa bạn đến link **download**.

**Bước 2:** Click vào “**Download SQL Server Management Studio (SSMS)**” để tải file SSMS

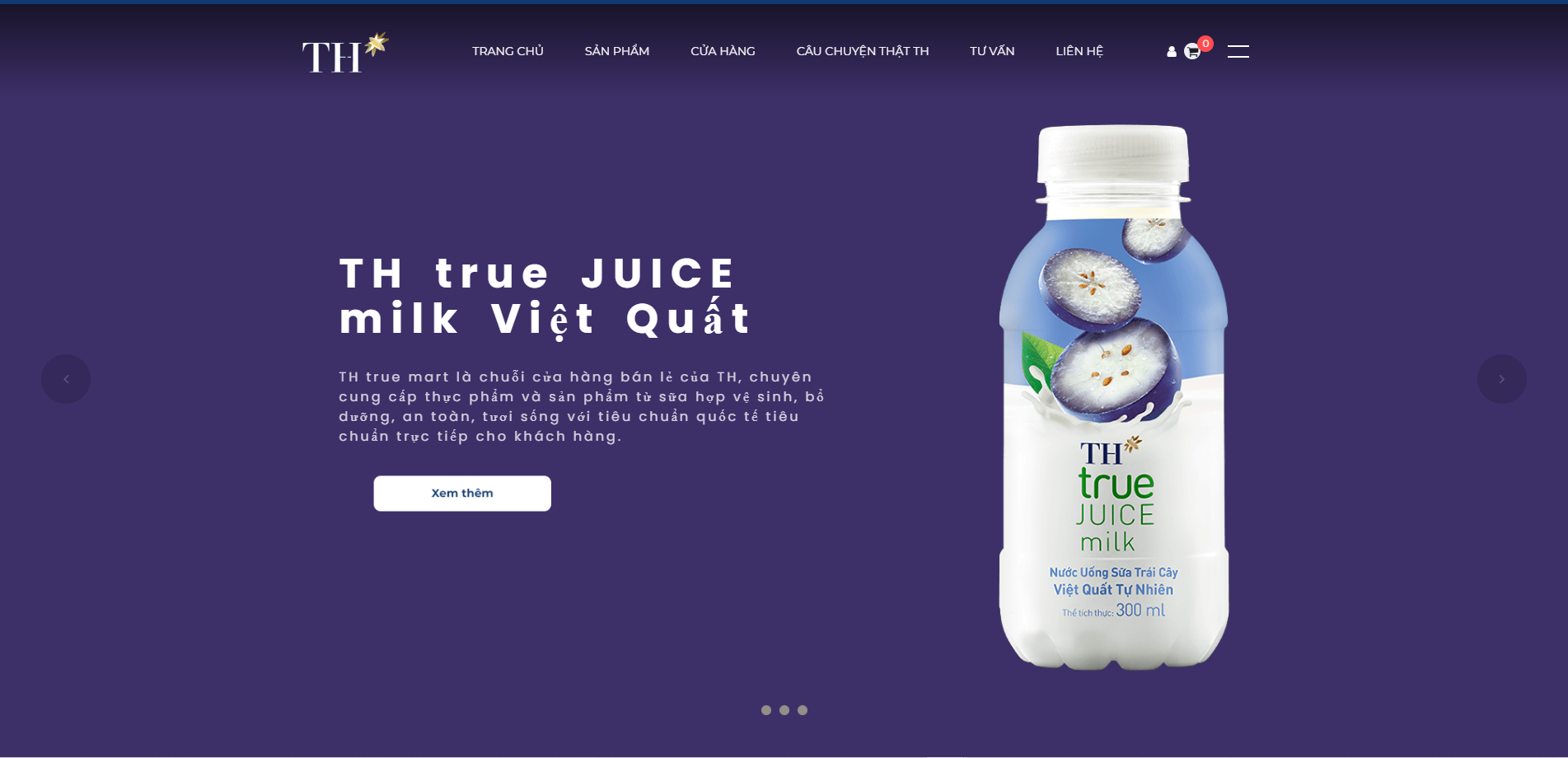
**Bước 3:** Mở file vừa download, bấm **Install** để cài giao diện cho phần mềm

Bước 4: Cuối cùng, chọn Close để hoàn tất.

## Giao diện sản phẩm

### *Khách hàng*

* “Trang chủ” khách hàng có thể ấn trên thanh menu để link đến các trang trong web



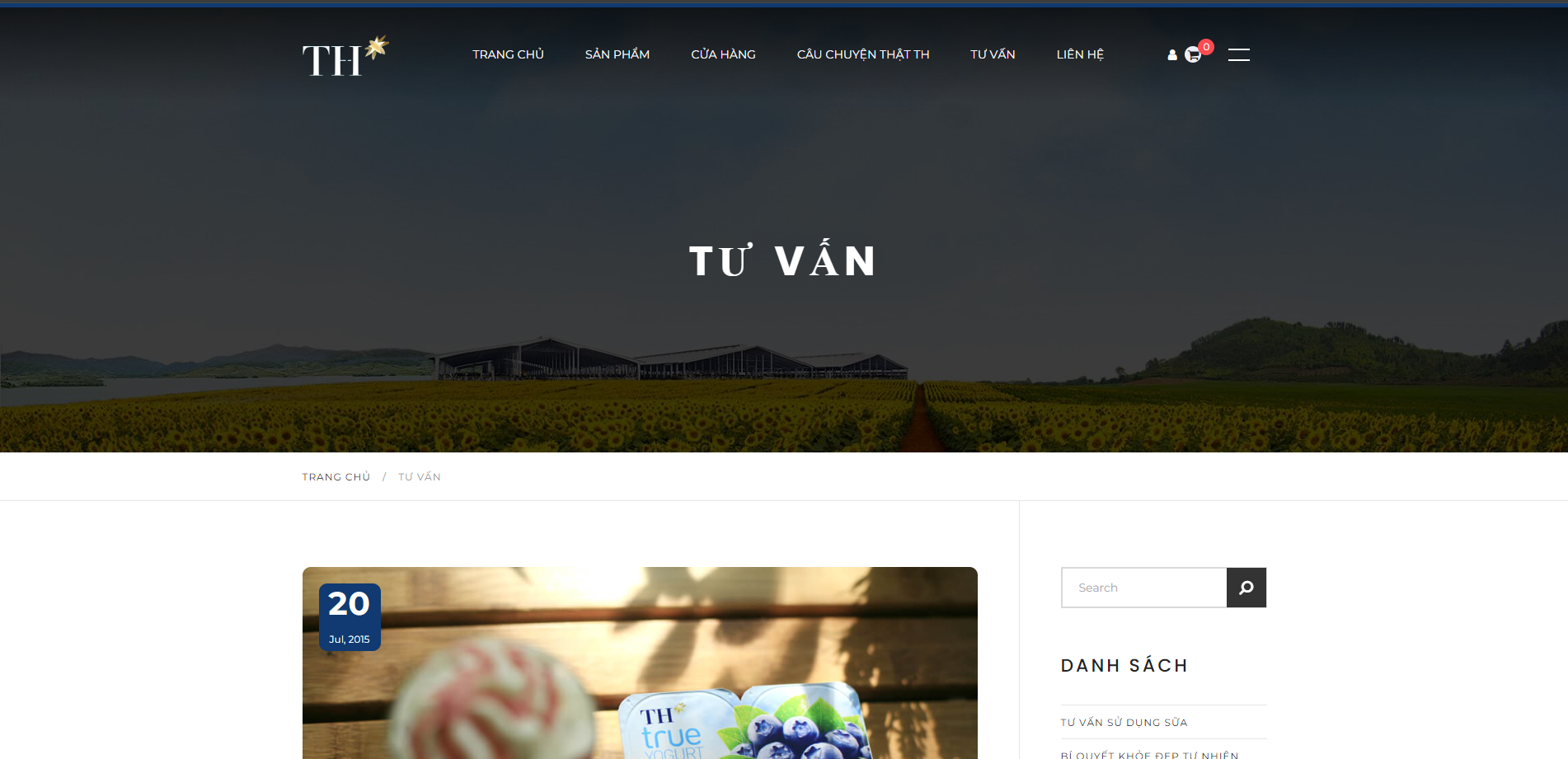
###### Hình 3.1: Giao diện Trang chủ

* “Câu chuyện thật TH” khách hàng có thể đọc hiểu về công ty TH hiểu về chất thật ý nghĩa của tên TH



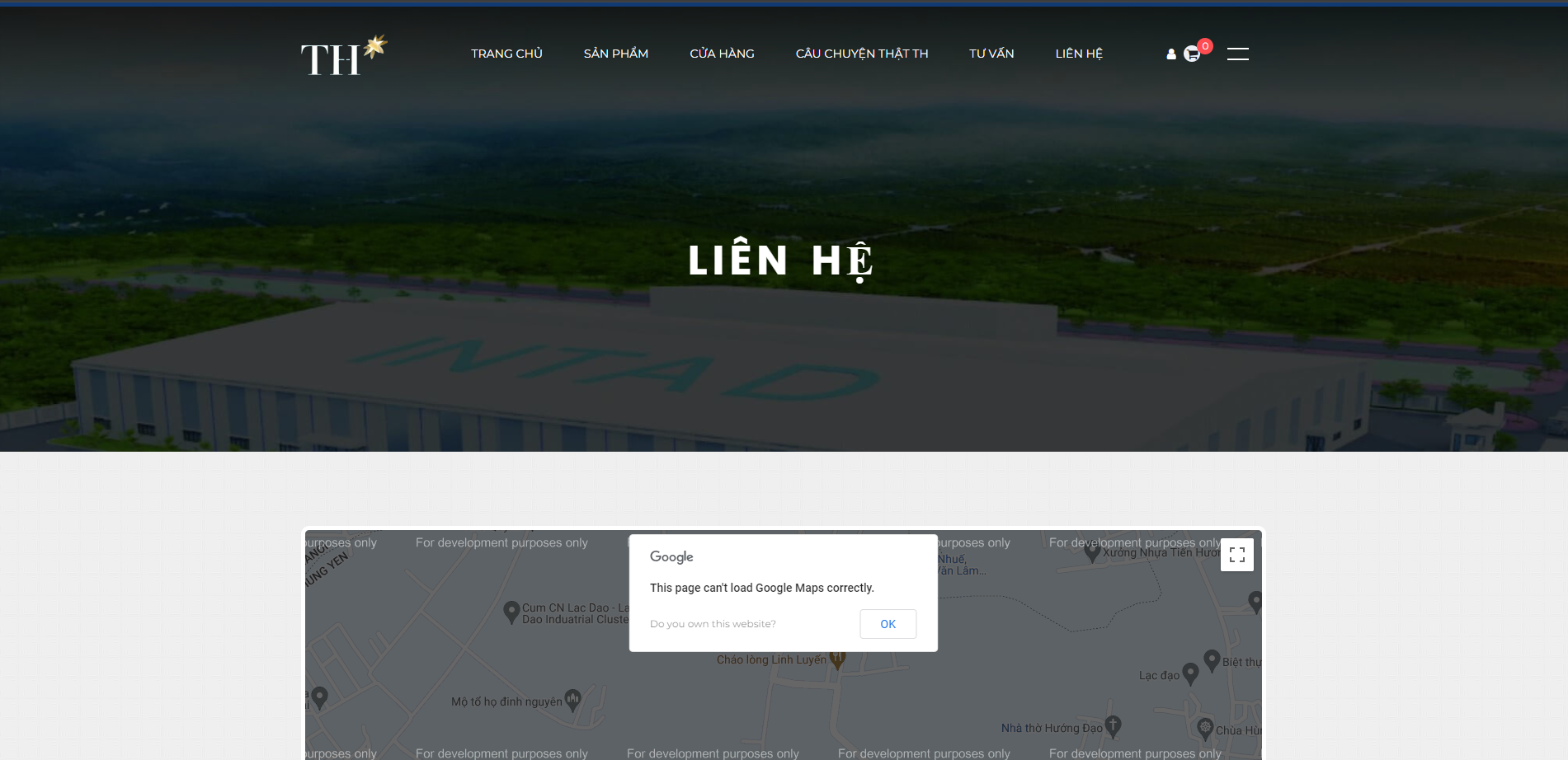
###### Hình 3.2: Giao diện Trang Câu chuyện thật TH

* “Tư vấn” khách vào đọc các bài viết tư vấn sử dụng sữa hoặc các bài viết liên quan đến sản phẩm TH



###### Hình 3.3: Giao diện Trang Tư vấn

* “Liên hệ” Khách vào xem map và gửi mail liên hệ với admin trao đổi



###### Hình 3.4: Giao diện Trang Liên hệ