**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**BỘ MÔN TIN HỌC ỨNG DỤNG**

**∙∙∙🙟🕮🙝∙∙∙**

****

**LUẬN VĂN TỐT NGHIỆP**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN TIN HỌC ỨNG DỤNG**

**Đề tài :**

**XÂY DỰNG ỨNG DỤNG WEB**

**MUA BÁN CHIA SẺ ĐỒ CỦ TRỰC TUYẾN DÀNH CHO SINH VIÊN**

**Sinh viên: Phan Hải Dương**

**Mã số: B1809225**

**Khóa: K44**

**Cần Thơ, 10/2022**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**BỘ MÔN TIN HỌC ỨNG DỤNG**

**∙∙∙🙟🕮🙝∙∙∙**

****

**LUẬN VĂN TỐT NGHIỆP**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN TIN HỌC ỨNG DỤNG**

**Đề tài :**

**XÂY DỰNG ỨNG DỤNG WEB**

**MUA BÁN CHIA SẺ ĐỒ CỦ TRỰC TUYẾN DÀNH CHO SINH VIÊN**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn** | **Sinh viên thực hiện** |
| **ThS. Lê Văn Quan** | **Phan Hải Dương**  **Mã số: B1809225**  **Lớp: THUD – K44** |

**Cần Thơ, 10/2022**

**LỜI CẢM ƠN**

Lời nói đầu tiên em xin phép gửi lời cảm ơn đến gia đình đã giúp đỡ, tạo điều kiện về vật chất và tinh thần, động viên, khích lệ và giúp đỡ em trong quá trình thực hiện luận văn.

Em xin được gửi lời cảm ơn chân thành đến quý Thầy, Cô công tác tại Trường Đại học Cần Thơ, Khoa Công nghệ thông tin và Truyền thông đã hướng dẫn cho em trong suốt quá trình học tập vừa qua. Vốn kiến thức này sẽ là nền tảng vững chắc cho quá trình học tập và nghiên cứu của em trong tương lai.

Để hoàn thành được luận văn, em xin tỏ lòng biết ơn sâu sắc đến Thầy Lê Văn Quan, là người đã trực tiếp hướng dẫn em trong suốt quá trình thực hiện, người đã giải đáp thắc mắc, nhắc nhở, cho em những nhận xét, đáng giá vô cùng hữu ích. Nhờ đó em đã hoàn thành luận văn của mình.

Trong quá trình thực hiện luận văn, mặc dù đã cố gắng hoàn thiện đề tài thông qua tham khảo tài liệu, trao đổi và tiếp thu ý kiến đóng góp của Thầy, Cô và các bạn nhưng do vốn kiến thức còn hạn hẹp, chưa có nhiều kinh nghiệm, khả năng lý luận của bản thân còn nhiều hạn chế nên không thể tránh khỏi những thiếu sót. Kính mong sự chỉ dẫn và đóng góp của quý thầy cô để nội dung đề tài được hoàn thiện hơn.

Cuối cùng xin kính chúc quý thầy cô dồi dào sức khỏe và đạt được nhiều thành công trong quá trình học tập và làm việc.

Cần Thơ, ngày … tháng 05 năm 2022

Sinh viên thực hiện

Phan Hải Dương

**NHẬN XÉT CỦA GIẢNG VIÊN**

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

Cần Thơ, ngày tháng năm

(GVHD ký và ghi rõ họ tên)

**MỤC LỤC**

[PHẦN 1: GIỚI THIỆU 1](#_Toc118449827)

[1. ĐẶT VẤN ĐỀ 1](#_Toc118449828)

[2. TÓM TẮT LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ 1](#_Toc118449829)

[3. MỤC TIÊU ĐỀ TÀI 1](#_Toc118449830)

[4. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU 2](#_Toc118449831)

[4.1. Đối tượng nghiên cứu 2](#_Toc118449832)

[4.2. Phạm vi nghiên cứu 2](#_Toc118449833)

[4.3. Phương pháp nghiên cứu 2](#_Toc118449834)

[4.4. Chức năng sản phẩm và đặc điểm người dùng 4](#_Toc118449835)

[5. NỘI DUNG NGHIÊN CỨU 5](#_Toc118449836)

[5.1. Tổng quan về ASP.NET Core 5](#_Toc118449837)

[5.2. Entity Framework Core 8](#_Toc118449838)

[5.3. Mô hình MVC 10](#_Toc118449839)

[5.4. Các thành phần trong ASP.NET Core 13](#_Toc118449840)

[5.5. Bootstrap 15](#_Toc118449841)

[5.6. jQuery 16](#_Toc118449842)

[PHẦN 2: NỘI DUNG 16](#_Toc118449843)

[CHƯƠNG 1: MÔ TẢ BÀI TOÁN 16](#_Toc118449844)

[1.1. MỤC TIÊU 16](#_Toc118449845)

[1.2. PHẠM VI SẢN PHẨM 17](#_Toc118449846)

[1.3. MÔI TRƯỜNG VẬN HÀNH 17](#_Toc118449847)

[1.4. CÁC RÀNG BUỘC VỀ THỰC THI VÀ CÔNG NGHỆ SỬ DỤNG 17](#_Toc118449848)

[CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP 18](#_Toc118449849)

[2.1. GIẢI PHÁP PHÂN TÍCH, THIẾT KẾ MÔ HÌNH 18](#_Toc118449852)

[PHẦN 2: NỘI DUNG 21](#_Toc118449853)

[CHƯƠNG 1: MÔ TẢ BÀI TOÁN 21](#_Toc118449854)

[I. MÔ TẢ CHI TIẾT BÀI TOÁN 21](#_Toc118449855)

[II. PHÂN TÍCH, ĐÁNH GIÁ CÁC GIẢI PHÁP CÓ LIÊN QUAN 21](#_Toc118449856)

[1. LỰA CHỌN GIẢI PHÁP 21](#_Toc118449857)

[CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP 24](#_Toc118449858)

[I. MÔ HÌNH USECASE CÁC MỨC 24](#_Toc118449859)

[6. MÔ HÌNH TẬP THỰC THỂ CÁC MỐI QUAN HỆ 25](#_Toc118449860)

[8. MÔ HÌNH VẬT LÝ 25](#_Toc118449861)

[9. MÔ HÌNH CƠ SỞ DỮ LIỆU QUAN HỆ 25](#_Toc118449862)

[10. SƠ ĐỒ CHỨC NĂNG 25](#_Toc118449863)

[11. MÔ HÌNH TUẦN TỰ CÁC HOẠT ĐỘNG CHÍNH 25](#_Toc118449864)

[CHƯƠNG 3: KIỂM THỬ VÀ ĐÁNH GIÁ 25](#_Toc118449865)

[I. KIỂM THỬ 25](#_Toc118449866)

[12. ĐÁNH GIÁ 25](#_Toc118449867)

[13. PHẦN 3: KẾT LUẬN 25](#_Toc118449868)

[I. KẾT QUẢ ĐẠT ĐƯỢC 25](#_Toc118449869)

[II. HƯỚNG PHÁT TRIỂN 25](#_Toc118449870)

[III. TÀI LIỆU THAM KHẢO 25](#_Toc118449871)

**DANH MỤC BẢNG**

**TÓM TẮT**

Ngày nay, kinh tế xã hội phát triển, đời sống vật chất được nâng cao, cùng với sự phát triển vượt bậc của Internet, nhu cầu mua bán và giao dịch trực tuyến cũng tăng theo đó. Có rất nhiều các trang web rao vặt trực tuyến hoạt động rất thành công, là cầu nối giữa người mua và người bán, giúp giảm thiểu chi phí, thay thế cho các hình thức rao vặt thông thường khác như đăng tin trên báo, tờ rơi… Có thể kể tới các trang web nổi tiếng như: Ebay, Amazon, MSN.

Vì vậy tôi đã chọn đề tài xây dựng hệ thống “Website rao vặt”, sử dụng lợi thế về công nghệ và sự tiện dụng của Internet, hệ thống giúp người bán dễ dàng giới thiệu sản phẩm của mình và đưa ra cho người mua nhiều phương thức tiếp cận hàng hóa đồng thời hỗ trợ thanh toán trực tuyến qua hệ thống ví điện tử một cách an toàn và nhanh chóng. Tận dụng những ưu thế của hệ thống đấu giá trực tuyến, hệ thống giúp người bán dễ dàng đưa sản phẩm của mình ra đấu giá, người mua dễ dàng tham gia đấu giá đem lại nhiều lợi ích về kinh tế. Ngoài ra, với bản đồ số tích hợp tạo ra sự dễ dàng trong việc xác định vị trí và tìm đường đến sản phẩm.

Trong quá trình nghiên cứu thực hiện đề tài, tôi đã ứng dụng các kiến thức đã

học trên ghế trường và tích lũy được nhiều kinh nghiệm trong việc xây dựng mô hình hệ thống thông tin, lập trình bằng ngôn ngữ C# và hệ CSDL SQLSERVER.

Website tạo ra đáp ứng được mục tiêu của đề tài song bên cạnh đó vẫn còn một số vấn đề chưa giải quyết được: phương thức vận chuyển chưa linh động về giá, thông tin quản lý còn đơn giản… Tuy nhiên, hệ thống vẫn có khả năng tiếp tục phát triển, cung cấp nhiều tiện ích hơn cho người dùng.

Rất mong nhận được sự đóng góp ý kiến để đề tài hoàn thiện hơn !

**ABSTRACT**

Nowadays, economic and social development, living standards improved,

With the boom of the Internet, demand for sale and online transactions also increased accordingly. There are many online sitesof a very successful operation, they are the bridge between buyers and sellers, helping reduce costs, replacing the normal classifieds form such as published in newspapers, leaflets...It can be considered to the famous sites such as Ebay, Amazon, MSN.

Therefore, I build a system infomation namely "Website classifieds", the advantages of using technology and the onvenience of the Internet, the system helps sellers to easily introduce their products and offer buyers many goods approach also supports online payments through E-wallet safely and quickly. Leveraging the advantages of online auction system, the system helps sellers toeasily put their products up for auction, buyers to bid easily bring many economic benefits. Also, digital map integration creates easy to identify and find a way to products location.

In the research process to implement the project, I have applied the knowledge learned in my school and accumulated much experience in modeling information systems, programming languages PHP and MYSQL database system.

Website created to meet the objectives of the project but there are still some unresolved issues: shipping method is not flexible on price, information management is simple... However, the system still able to continue to grow, providing greater convenience for users.

We hope to receive the suggestions for project more complete !

# PHẦN 1: GIỚI THIỆU

## ĐẶT VẤN ĐỀ

Việt Nam hiện nay có hơn 65 triệu người dùng Internet, với sự phát triển rông khắp đó, thương mại điện tử ở Việt Nam đang ngày một phát triển dữ dội. Cùng với sự phát triển đó, nhu cầu trao đổi, mua bán trực tuyến của người dùng Internet đang trở thành một loại hình dịch vụ phát triển mạnh mẽ.

Trong việc mua bán, đối với người kinh doanh có nguồn vốn ban đầu thấp, làm thế nào để vừa có được không gian bán hàng tốt vừa tiếp cận khách hàng dễ dàng mà không tốn quá nhiều chi phí cho việc chạy quảng cáo, thuê nhân viên, mặt bằng, trang trí cửa hàng, … còn đối với người tiêu dùng thì làm thế nào để dễ dàng tìm ra được một cửa hàng bán sản phẩm chất lượng, uy tín, tham khảo sản phẩm vào những thời gian rảnh rỗi mà không cần tốn quá nhiều công sức và thời gian di chuyển, chủ động trong việc thanh toán và vận chuyển.

Với tốc độ phát triển mạnh mẽ của công nghệ như hiện nay đã làm nâng cao chất lượng cuộc sống của con người, nhu cầu sử dụng các dịch vụ tiện ích ngày càng tăng cao. Việc mua bán truyền thống dần có những bất cập nhất định như tốn khá nhiều thời gian trong việc tìm kiếm và mua sản phẩm, giá của sản phẩm khá đắt do một số cửa hàng không sử dụng mã giảm giá, … trong khi đó mua bán online ngày càng thể hiện được những ưu điểm nổi bật như tìm kiếm thông tin sản phẩm dễ dàng, không tốn quá nhiều thời gian cho việc mua sản phẩm, có thể đặt mua sản phẩm vào bất kỳ thời gian nào và một ưu điểm nổi bật là giao hàng tận nơi, ...

Vì vậy đối với việc mua bán, chia sẻ đồ cũ và các sản phẩm có liên quan thì website rao vặt dành cho sinh viên với các công nghệ như ASP .Net Core – hệ quản trị cơ sở dữ liệu SQLServer hỗ trợ đa nền tảng được thiết kế theo hướng đối tượng với giao diện sử dụng Bootstrap đẹp mắt, gần gũi đối với những người kinh doanh và người tiêu dùng là một giải pháp hiệu quả và cần thiết.

## TÓM TẮT LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ

Với lĩnh vực giới thiệu và rao vặt online đã có một số các website áp dụng và nhận được nhiều sự quan tâm của người tiêu dùng như: chotot.com, muaban.net, raovatchotot.net … nhưng nhìn chung các website trên chưa thật sự gần gũi với người tiêu dùng bởi quy trình tìm hiểu và đặt mua sản phẩm chưa thật sự tối ưu và chưa tập trung vào phục vụ một nhóm đối tượng cụ thể.

## MỤC TIÊU ĐỀ TÀI

Mục tiêu của đề tài là tìm hiểu về nguyên lý và cách thức hoạt động của web service và web client, Ứng dụng các ngôn ngữ lập trình và các công nghệ để phát triển website như là: ASP .NET Core, C#, Javascript, HTML, CSS, SQLSERVER,… Bên cạnh đó hiểu được cách thức thiết kế và xây dựng chức năng của một website bán hàng. 2 Sau khi hiểu rõ về những kiến thức trên, tiếng hành xây dựng website quản trị với các chức năng như thêm, sửa, xóa, tìm kiếm các thông tin, thống kế sản phẩm bán được, các chức năng quản trị tài khoản cần thiết phục vụ cho người quản trị và xây dựng website dành cho khách hàng với các chức năng như xem danh sách sản phẩm, xem chi tiết sản phẩm, thêm vào giỏ hàng, đặt hàng, tìm kiếm sản phẩm, theo dõi đơn hàng, bình luận sản phẩm và các chức năng quản trị tài khoản.

## ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

### Đối tượng nghiên cứu

Xây dựng hệ thống “Website rao vặt sinh viên” để phục vụ nhu cầu rao vặt, mua bán, chía sẻ sản phẩm đã hoặc chưa sử dụng trực tuyến dành người dùng chủ yếu là sinh viên và những ai co nhu cầu liên quan trên toàn quốc.

### Phạm vi nghiên cứu

* Tìm hiểu về ngôn ngữ Javascript, HTML5, CSS.
* Tìm hiểu ASP .Net Framework, hệ quản trị SQLServer.
* Tìm hiểu về Web Service và Web Client

### Phương pháp nghiên cứu

#### Hướng giải quyết

* Tìm hiểu về quá trình rao vặt các sản phẩm phục vụ nhu cầu chủ yếu cho sinh viên qua hình thức online.
* Tham khảo một số website rao vặt để xây dựng một website gần gũi, đáp ứng được yêu cầu của người dùng. Đồng thời, xây dựng website quản trị nhằm hỗ trợ người quản lý dễ dàng quản lý thông tin.
* Tìm hiểu cách thiết kế và các chức năng cơ bản của một website rao vặt.
* Nghiên cứu về mô hình Unified Modeling Language (UML), vận dụng lý thuyết để phân tích hệ thống, xây dựng các mô hình, sơ đồ chức năng, lưu đồ giải thuật giải quyết vấn đề.
* Xây dựng mô hình cơ sở dữ liệu.
* Xây dựng tập dữ liệu.
* Thiết kế giao diện.
* Xây dựng các chức năng của đề tài.

#### Phương pháp thực hiện

Quá trình thực hiện đề tài cụ thể như sau:

|  |  |  |
| --- | --- | --- |
| **STT** | **Công việc** | **Thời gian (đơn vị: ngày)** |
| **1** | **Tìm hiểu và chọn đề tài** | **6** |
|  | Khảo sát thực tế | 4 |
| Lựa chọn đề tài | 2 |
| **2** | **Xác định yêu cầu và xây dựng kế hoạch thực hiện đề tài** | **9** |
|  | Thu thập thông tin về đề tài, tìm hiểu quy trình hiện có | 5 |
| Đánh giá rủi ro | 2 |
| Ước lượng thời gian thực hiện | 2 |
| **3** | **Phân tích thiết kế hệ thống** | **29** |
|  | Đặc tả dữ liệu | 4 |
| Đặc tả xử lý | 4 |
| Đặc tả chức năng | 5 |
| Thiết kế tổng thể hệ thống | 7 |
| Thiết kế cơ sở dữ liệu | 5 |
| Xây dựng kịch bản kiểm thử | 4 |
| **4** | **Thực hiện lập trình** | **74** |
|  | Thiết kế giao diện cho website quản trị | 6 |
| Xây dựng các chức năng quản trị | 35 |
| Thiết kế giao diện website cho khách hàng | 8 |
| Xây dựng các chức năng dành cho khách hàng | 25 |
| **5** | **Tích hợp và kiểm thử hệ thống** | **13** |
|  | Tích hợp các chức năng | 6 |
| Kiểm thử các chức năng theo kịch bản | 3 |
| Kiểm tra hệ thống và sửa lỗi | 4 |
| **6** | **Kiểm thử chấp nhận** | **6** |
|  | Demo cho giảng viên hướng dẫn | 1 |
| Chỉnh sửa và hoàn thiện | 5 |
| **7** | **Hoàn thành báo cáo** | **3** |

### Chức năng sản phẩm và đặc điểm người dùng

Sản phẩm có chức năng phân quyền sử dụng tài khoản với bốn đối tượng sử dụng: admin, khách hàng có đăng ký tài khoản và khách hàng không đăng ký tài khoản, đây là chức năng quan trọng hỗ trợ người dùng thao tác với các chức năng được cho phép đối với mỗi tài khoản trên website quản trị và website dành cho khách hàng.

Với website dành cho khách hàng, khách hàng dễ dàng xem danh sách sản phẩm theo danh mục, xem thông tin chi tiết sản phẩm, xem các gợi ý sản phẩm khuyến mãi 5 và sản phẩm bán chạy, tìm kiếm sản phẩm, đối với khách hàng có đăng ký tài khoản có thế thể thêm sản phẩm vào giỏ hàng, đặt mua sản phẩm, theo dõi đơn hàng, đánh giá sản phẩm và quản lý tài khoản cá nhân. Đối với web saite dành cho admin, admin dễ dàng quản lý những thông tin về khách hàng, thông tin cửa hàng, danh mục sản phẩm, sản phẩm, nhà cung cấp, hình thức vận chuyển, khuyến mãi, xem báo cáo thống kê.

#### Chức năng của người quản trị

* Đăng nhập
* Đổi mật khẩu
* Đăng xuất
* Quản lý thông tin khách hàng
* Quản lý thông tin cửa hàng
* Quản lý các danh mục
* Quản lý thông tin sản phẩm
* Quản lý thông tin nhà cung cấp
* Báo cáo thống kê

#### Chức năng của khách hàng có đăng ký tài khoản

* Đăng nhập
* Đổi mật khẩu
* Đăng xuất
* Xem danh sách sản phẩm
* Xem thông tin giới thiệu sản phẩm
* Quản lý giỏ hàng cá nhân
* Đặt hàng
* Lựa chọn hình thức thanh toán
* Theo dõi đơn hàng
* Đánh giá sản phẩm
* Xem gợi ý sản phẩm
* Tìm kiếm sản phẩm theo từ khóa

#### Chức năng của khách hàng không có đăng ký tài khoản

* Đăng ký thành viên
* Xem thông tin sản phẩm
* Tìm kiếm sản phẩm

## NỘI DUNG NGHIÊN CỨU

### Tổng quan về ASP.NET Core

#### ASP.NET

ASP.NET lần đầu tiên được tung ra thị trường vào tháng 1 năm 2002 là một bộ phận của .NET Framework 1.0 nhằm thay thế cho ASP (Active Server Pages). Trên nền tảng ASP.NET, Microsoft xây dựng một số mô hình như: Web Form, WCF, MVC, Web API,... để hỗ trợ người lập trình. [6]

Mô hình Web Form ra đời năm 2002 hướng tới mô hình web “statefull” dựa trên sự kiện tương tự như Windows Forms không thích hợp cho các ứng dụng lớn, hạn chế khi test, mô hình stataful quá phức tạp, không kiểm soát được HTML,..và dần bị thay thế khi ASP.NET MVC ra đời. [6]

ASP.NET MVC ra đời vào năm 2009 hoạt động dựa trên mô hình kiến trúc MVC (Model-View-Control) mang lại nhiều lợi ích và được sử dụng rộng rãi. [6] ASP.NET Web API ra đời vào năm 2012 giúp phát triển ứng dụng dạng dịch vụ REST và dẫn thay thế WCF (Windows Communication Foundation, ra đời năm 2006). [6]

Tất cả các mô hình lập trình của ASP.NET đều được xây dựng trên cùng một nền tảng, sử dụng chung thư viện System.Web.dll và được hỗ trợ mạnh mẽ .NET Framework cho ra đời các ứng dụng web có thể sử dụng tất cả cá tính năng của cả .NET Framework và ASP.NET. [6]

Tuy nhiên, ASP.NET lại chịu sự giới hạn về sự phát triển của .NET Framework, gắn chặt với dịch vụ hosting của Windows sử dụng IIS (Internet Information Service) nên không thể chạy trên đa nền tảng và không phù hợp với các ứng dụng web hiện đại. [6]

#### Giới thiệu về .NET Core

* + **Khái niệm**

.NET Core là phiên bản mới của .NET Framework, là một nền tảng phát triển mục đích chung, miễn phí, mã nguồn mở được duy trì bởi Microsoft. Nó là một framework đa nền tảng chạy trên các hệ điều hành Windows, macOS và Linux. [12] .NET Core Framework có thể được sử dụng để xây dựng các loại ứng dụng khác nhau như thiết bị di động, máy tính để bàn, web, đám mây, IoT, máy học, microservices, trò chơi,... [12]

.NET Core là một platform nhẹ, nhanh và đa nền tảng. Nó bao gồm các tính năng cốt lõi cần thiết để chạy một ứng dụng .NET Core cơ bản. Các tính năng khác được cung cấp dưới dạng gói NuGet, người dùng có thể thêm nó vào ứng dụng của mình nếu cần. Bằng cách này, ứng dụng .NET Core tăng tốc hiệu suất, giảm dung lượng bộ nhớ và trở nên dễ bảo trì. [12]

* + **Đặc điểm của .NET Core**
* Đa nền tảng: Chạy trên các hệ điều hành Windows, macOS và Linux. Có thời gian chạy khác nhau cho mỗi hệ điều hành thực thi mã và tạo ra cùng một đầu ra. [12]
* Nhất quán trên các kiến trúc: Thực thi mã với cùng một hành vi trong các kiến trúc tập lệnh khác nhau, bao gồm x64, x86 và ARM. [12]
* Các công cụ dòng lệnh: Gồm các công cụ dòng lệnh dễ sử dụng, có thể được sử dụng để phát triển cục bộ và trong các tình huống tích hợp liên tục. [12]
* Triển khai linh hoạt: có thể cài đặt song song (cài đặt toàn người dùng hoặc toàn hệ thống). Có thể được sử dụng với các Docker Containers. [12]
* Hỗ trợ nhiều ngôn ngữ: Bạn có thể sử dụng các ngôn ngữ lập trình C#, F# và Visual Basic để phát triển các ứng dụng .NET Core. Bạn có thể sử dụng IDE yêu thích của mình, bao gồm Visual Studio 2017/2019, Visual Studio Code, Sublime Text, Vim, ... [12]
* Kiến trúc mô-đun: .NET Core hỗ trợ cách tiếp cận kiến trúc mô-đun sử dụng các gói NuGet. Có các gói NuGet khác nhau cho các tính năng khác nhau có thể được thêm vào dự án .NET Core nếu cần. Ngay cả thư viện .NET Core
* cũng được cung cấp dưới dạng gói NuGet. Gói NuGet cho mô hình ứng dụng .NET Core mặc định là Microsoft.NETCore.App. [12]
* Tương thích: .NET Core tương thích với .NET Framework, Xamarin và Mono, thông qua .NET Standard. [12]
* Nguồn mở: Nền tảng .NET Core là nguồn mở, sử dụng giấy phép MIT và Apache 2. .NET Core là một dự án .NET Foundation. [12]
* Được hỗ trợ bởi Microsoft: .NET Core được Microsoft hỗ trợ. [12]
  + **Thành phần của .NET Core**
* .NET Core runtime: cung cấp một hệ thống kiểu, tải lắp ráp, trình thu gom rác,

interop gốc và các dịch vụ cơ bản khác. Các thư viện khung .NET Core cung

cấp các kiểu dữ liệu nguyên thủy, các kiểu thành phần ứng dụng và các tiện ích cơ bản. [12]

* ASP.NET Core runtime: cung cấp khung để xây dựng các ứng dụng kết nối internet , điện toán đám mây hiện đại, chẳng hạn như ứng dụng web, ứng dụng IoT và phụ trợ di động. [12]
* .NET Core SDK và trình biên dịch ngôn ngữ (Roslyn và F #) cho phép trải nghiệm nhà phát triển .NET Core. [12]
* Dotnet command, được sử dụng để khởi chạy các ứng dụng .NET Core và các lệnh CLI. Nó chọn thời gian chạy và lưu trữ thời gian chạy, cung cấp chính sách tải lắp ráp và khởi chạy các ứng dụng và công cụ. [12]

#### Sơ lược về lịch sử của ASP.NET Core

Hàng triệu nhà phát triển sử dụng hoặc đã sử dụng ASP.NET 4.x để tạo ứng dụng web. ASP.NET Core là một bản thiết kế lại của ASP.NET 4.x, bao gồm những thay đổi về kiến trúc dẫn đến một modular framework gọn gàng hơn. [6]

#### Khái niệm ASP.NET Core

ASP.NET Core là một framework nguồn mở, hiệu suất cao, đa nền tảng để xây dựng các ứng dụng hiện đại, hỗ trợ đám mây, kết nối Internet. Với ASP.NET Core, bạn có thể:

* Xây dựng các ứng dụng và dịch vụ web, ứng dụng Internet of Things (IoT) và các phần mềm phụ trợ dành cho thiết bị di động. [6]
* Sử dụng các công cụ phát triển yêu thích của bạn trên Windows, macOS và Linux. [6]
* Triển khai lên đám mây hoặc tại chỗ. [6]
* Chạy trên .NET Core. [6]

#### Lợi ích của ASP.NET Core

* Thống nhất xây dựng web UI (User Interface) và web APIs:
* Mô hình Model-View-Controller (MVC) giúp kiểm tra các API web và
* các ứng dụng web. [6]
* Razor Pages là một mô hình lập trình dựa trên trang giúp việc xây dựng giao diện người dùng web dễ dàng hơn và hiệu quả hơn. [6]
* Đánh dấu Razor cung cấp một cú pháp hiệu quả cho các trang Razor vàgiao diện MVC. [6]
* Tag Helpers cho phép mã phía máy chủ tham gia vào việc tạo và hiển thị các phần tử HTML trong tệp Razor. [6]
* Hỗ trợ tích hợp cho nhiều định dạng dữ liệu và thương lượng nội dung cho phép các API web của bạn tiếp cận nhiều khách hàng, bao gồm cả trình duyệt và thiết bị di động. [6]
* Liên kết mô hình tự động ánh xạ dữ liệu từ các yêu cầu HTTP tới các tham số của phương thức hành động. [6]
* Xác thực mô hình tự động thực hiện xác thực phía máy khách và phía máy chủ. [6]
* Được lưu trữ lại để kiểm tra. [6]
* Razor Pages giúp tạo các kịch bản tập trung vào mã hóa trang dễ dàng và hiệu quả hơn. [6]
* Blazor cho phép bạn sử dụng C# trong trình duyệt cùng với JavaScript. Chia sẻ logic ứng dụng phía máy chủ và phía máy khách đều được viết bằng .NET. [6]
* Khả năng phát triển và chạy trên đa nền tảng Windows, macOS và Linux. [6]
* Mã nguồn mở và tập trung vào cộng đồng. [6]
* Tích hợp hiện đại giữa client-side framework và quy trình phát triển: ASP.NET Core tích hợp liền mạch với các thư viện và client-side framework phổ biến, bao gồm Blazor, Angular, React và Bootstrap. [6]
* Hỗ trợ lưu trữ các dịch vụ Remote Procedure Call (RPC) bằng gRPC. [6]
* Hệ thống cấu hình dựa trên môi trường, sẵn sàng cho đám mây. [6]
* Tích hợp dependency injection. [6]
* Modular của đường dẫn yêu cầu HTTP nhẹ và hiệu suất cao. [6]
* Khả năng lưu trữ trên: Kestrel, IIS, HTTP.sys, Nginx, Apache, Docker. [6]
* Lập phiên bản song song. [6]
* Công cụ đơn giản hóa việc phát triển web hiện đại. [6]

### Entity Framework Core

#### Giới thiệu về Entity Framework Core

Entity Framework (EF) Core là phiên bản nhẹ, có thể mở rộng, mã nguồn mở và đa nền tảng của công nghệ truy cập dữ liệu Entity Framework phổ biến. EF Core có thể hoạt động như một trình ánh xạ quan hệ đối tượng (O/RM), trong đó: [6]

* Cho phép nhà phát triển .NET làm việc với cơ sở dữ liệu bằng các đối tượng .NET.
* Loại bỏ sự cần thiết của hầu hết các mã truy cập dữ liệu thường cần được viết.

#### Entity Framework O/RM

Mặc dù EF Core giỏi trong việc tóm tắt nhiều chi tiết lập trình, nhưng có một số phương pháp hay nhất giúp tránh những cạm bẫy phổ biến trong ứng dụng sản xuất áp dụng cho bất kỳ O/RM. [6]

Kiểm tra chức năng và tích hợp: Điều quan trọng là phải tái tạo môi trường sản xuất càng chặt chẽ càng tốt để:

* Tìm sự cố trong ứng dụng chỉ hiển thị khi sử dụng phiên bản đặc biệt hoặc phiên bản máy chủ cơ sở dữ liệu. [6]
* Nắm bắt các thay đổi đột phá khi nâng cấp EF Core và các phụ thuộc khác. Ví dụ: thêm hoặc nâng cấp các khuôn khổ như ASP.NET Core, OData hoặc Automapper. Những phụ thuộc này có thể ảnh hưởng đến EF Core theo những cách không mong muốn. [6]

Kiến thức cấp độ trung cấp trở lên về máy chủ cơ sở dữ liệu cơ bản là điều cần thiết để thiết kế, gỡ lỗi, cấu hình và di chuyển dữ liệu trong các ứng dụng sản xuất hiệu suất cao. [6]

Kiểm thử hiệu suất và kiểm thử độ bền với việc tải dữ liệu. Việc sử dụng naïve của một số tính năng không được mở rộng. Ví dụ: nhiều collection Include, sử dụng nhiều tính năng tải chậm, truy vấn có điều kiện trên các cột không được lập chỉ mục, cập nhật và chèn lớn với các giá trị do cửa hàng tạo, thiếu xử lý đồng thời, mô hình lớn, chính sách bộ nhớ cache không đầy đủ. [6]

Đánh giá bảo mật: Ví dụ: xử lý các chuỗi kết nối và các bảo mật khác, quyền cơ sở dữ liệu cho hoạt động không triển khai, xác thực đầu vào cho SQL thô, mã hóa cho dữ liệu nhạy cảm. [6]

Đảm bảo ghi nhật ký và chẩn đoán là đủ và có thể sử dụng được. Ví dụ: cấu hình ghi nhật ký thích hợp, thẻ truy vấn và thông tin chi tiết về ứng dụng. [6]

Khôi phục lỗi. Chuẩn bị các phương án dự phòng cho các tình huống lỗi phổ biến như khôi phục phiên bản, máy chủ dự phòng, cân bằng tải và cân bằng quy mô, giảm thiểu DoS và sao lưu dữ liệu. [6]

Ứng dụng deployment và migration. Lập kế hoạch cho việc migration sẽ được áp dụng như thế nào trong khi deployment; làm điều đó khi khởi động ứng dụng có thể gặp phải các vấn đề đồng thời và yêu cầu quyền cao hơn mức cần thiết cho hoạt động bình thường. Sử dụng giai đoạn để tạo điều kiện khôi phục từ các lỗi nghiêm trọng trong quá trình migration. [6]

Kiểm tra chi tiết và thử nghiệm các migration được tạo ra. Các migration phải được kiểm tra kỹ lưỡng trước khi áp dụng cho dữ liệu xuất. Không thể dễ dàng thay đổi hình dạng của lược đồ và loại cột khi các bảng chứa dữ liệu xuất. [6]

#### Object Services

Object Service là một điểm vào chính cho việc truy cập dữ liệu từ CSDL và trả về. Object service có trách nhiệm trong việc cụ thể hóa quá trình chuyển đổi dữ liệu trả về từ một entity client data provider (lớp tiếp theo) tới một entity object structure. [8]

#### EntityClient Data Provider

Trách nhiệm chính của lớp này là chuyển đổi L2E hoặc những truy vấn Entity SQL vào một truy vấn SQL , nó được hiểu bởi CSDL cơ bản. Nó giao tiếp với ADO.Net data provider lần lượt gửi và nhận dữ liệu từ CSDL. [8]

#### EDM

EDM (Entity Data Model): EDM gồm ba phần chính – Conceptual model, Mapping và Storage model: [8]

* Conceptual Model: chứa các model class và những quan hệ của nó. Phần này sẽ độc lập với thiết kế bảng CSDL. [8]
* Storage Model: là database design model gồm các bảng, views, stored procedures, và những quan hệ của nó và các khóa. [8]
* Mapping: gồm có thông tin về cách làm thế nào Conceptual model nối với Storage model. [8]

### Mô hình MVC

#### Giới thiệu về mô hình MVC

MVC là viết tắt của Model – View – Controller. Là một trong những design parttern được vận hành để tách mã lệnh thành ba phần riêng biệt. Mỗi phần MVC sẽ có chức năng đặc thù để xử lý tác vụ mà request gửi tới. MVC làm cho mã lệnh trở nên trong sáng, dễ phát triển và dễ nâng cấp theo thời gian. [13]

Để làm việc tốt đối với MVC, chúng ta cần nắm thật vững kiến thức OOP (Object Oriented Programming). 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. [13]

#### Kiến trúc mô hình MVC

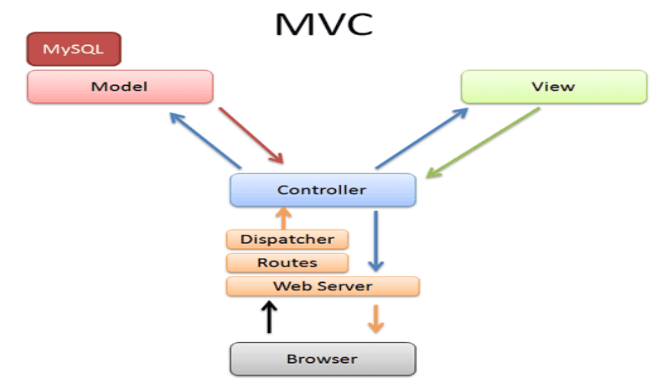
Trong kiến trúc MVC, một đối tượng đồ họa người dùng (GUI Component) bao gồm 3 thành phần cơ bản: Model, View và Controller. Model có trách nhiệm đối với toàn bộ dữ liệu cũng như trạng thái của đối tượng đồ họa. View chính là thể hiện trực quan của Model, hay nói cách khác chính là giao diện của đối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

Controller điều khiển biệc tương tác giữa đối tượng đồ họa với người sử dụng như những đối tượng khác. [13]

Model: Là thành phần chịu trách nhiệm xử lý các thao tác trên database và gửi trả kết quả thông qua view. [13]

View: Là thành 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. [13]

Controller: Là thành 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 trắc nhiệm gửi và nhận request từ hệ thống tới người sử dụng. [13]



Khi người sử dụng hoạc 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 hiện thị 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 dùng hoặc các đối tượng khác. [13]

#### Đặc điểm mô hình MVC

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 thay đổi trong một phần của ứng dụng mà không ảnh hưởng đến các thành phần khác. [13]

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 trong 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. [13]

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 dưa trên kiến trúc MVC. Đặc biệt là nền tảng ASP.NET MVC Framework, CodeIgniter Framework và Laravel Framework jau Struts Framework,... [13]

#### Ưu điểm và nhược điểm của mô hình MVC

**Ưu điểm:**

* Nó giúp dễ dàng quản lý độ phức tạp bằng cách chia ứng dụng thành model view và controller. [13]
* Nó không sử dụng trạng thái xem hoặc các biểu mẫu dựa trên máy chủ. Điều này làm cho khung MVC trở nên lý tưởng cho các nhà phát triển muốn toàn quyền kiểm soát hành vi của một ứng dụng. [13]
* Nó sử dụng mẫu Front Controller để xử lý các yêu cầu ứng dụng Web thông qua một controller duy nhất. Điều này cho phép bạn thiết kế một ứng dụng hỗ trợ cơ sở hạ tầng định tuyến phong phú. [13]
* Nó cung cấp hỗ trợ tốt hơn cho phát triển theo hướng thử nghiệm TDD (test- driven development). [13]
* Nó hoạt động tốt cho các ứng dụng Web được hỗ trợ bởi các nhóm lớn các nhà phát triển và nhà thiết kế Web, những người cần mức độ kiểm soát cao đối với hành vi của ứng dụng. [13]

**Nhược điểm:**

* MVC tận dụng mảng là thành phần chính cho việc truy vấn dữ liệu. Nhất làvới việc sử dụng việc sử dụng active record để viết ứng dụng. Chúng luôn cần người viết phải nắm vững mô hình mảng đa chiều. [13]
* Mặc dù MVC tỏ ra lợi thế hơn nhiều so với cách lập trình thông thường nhưng MVC luôn phải nạp 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. [13]
* MVC đòi hỏi người tiếp cận phải biết qua OOP (Object Oriented Programming), 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.

### Các thành phần trong ASP.NET Core

#### Razor Pages

ASP.NET Core Razor Pages là một page-focused framework phía máy chủ cho phép xây dựng các trang web động theo hướng dữ liệu với sự tách biệt rõ ràng các mối quan hệ. Dựa trên phiên bản ASP.NET mới nhất của Microsoft - ASP.NET Core, Razor Pages hỗ trợ phát triển đa nền tảng và có thể được triển khai cho các hệ điều hành Windows, Unix và Mac. [7]

Razor Pages framework nhẹ và rất linh hoạt. Nó cung cấp cho nhà phát triển toàn quyền kiểm soát đối với HTML được hiển thị. Framework được xây dựng dựa trên ASP.NET Core MVC. Razor Pages là framework được khuyến nghị để tạo HTML phía máy chủ đa nền tảng trên .NET Core. Bạn không cần phải có bất kỳ kiến thức hoặc hiểu biết nào về MVC để làm việc với Razor Pages. [7]

Razor Pages sử dụng ngôn ngữ lập trình C# phổ biến cho lập trình phía máy chủ và cú pháp tạo mẫu Razor dễ học để nhúng C# vào HTML đánh dấu để tạo nội dung động cho trình duyệt. [7]

#### API

* **Khái niệm API**

API (viết tắt của Application Programming Interface - giao diện lập trình ứng dụng) là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác. API cung cấp khả năng truy xuất đến một tập các hàm hay dùng. Và từ đó có thể trao đổi dữ liệu giữa các ứng dụng. [13]

* **Ứng dụng của API**

Web API: là hệ thống API được sử dụng trong các hệ thống website. Hầu hết các website đều ứng dụng đến Web API cho phép bạn kết nối, lấy dữ liệu hoặc cập nhật cơ sở dữ liệu. Ví dụ: Bạn thiết kế chức nằng login thông Google, Facebook, Twitter, Github... Điều này có nghĩa là bạn đang gọi đến API của các ứng dụng trên. Hoặc như các ứng dụng di động đều lấy dữ liệu thông qua API. [13]

API trên hệ điều hành: Windows hay Linux có rất nhiều API, họ cung cấp các tài liệu API là đặc tả các hàm, phương thức cũng như các giao thức kết nối. Nó giúp lập trình viên có thể tạo ra các phần mềm ứng dụng có thể tương tác trực tiếp với hệ điều hành. [13]

API của thư viện phần mềm hay framework: API mô tả và quy định các hành động mong muốn mà các thư viện cung cấp. Một API có thể có nhiều cách triển khai khác nhau và nó cũng giúp cho một chương trình viết bằng ngôn ngữ này có thể sử dụng thư viện được viết bằng ngôn ngữ khác. [13]

#### RESTful HTTP services

* **Giới thiệu về RESTful HTTP services**

RESTful Web Service là các Web Service được viết dựa trên kiến trúc REST. REST đã được sử dụng rộng rãi thay thế cho các Web Service dựa trên SOAP và WSDL. RESTful Web Service nhẹ (lightweigh), dễ dàng mở rộng và bảo trì. [9]

REST định nghĩa các quy tắc kiến trúc để bạn thiết kế Web services, chú trọng vào tài nguyên hệ thống, bao gồm các trạng thái tài nguyên được định dạng như thế nào và được truyền tải qua HTTP, và được viết bởi nhiều ngôn ngữ khác nhau. Nếu tính theo số dịch vụ mạng sử dụng, REST đã nổi lên trong vài năm qua như là một mô hình thiết kế dịch vụ chiếm ưu thế. Trong thực tế, REST đã có những ảnh hưởng lớn và gần như thay thế SOAP và WSDL vì nó đơn giản và dễ sử dụng hơn rất nhiều. [9]

REST là một bộ quy tắc để tạo ra một ứng dụng Web Service, mà nó tuân thủ 4 nguyên tắc thiết kế cơ bản sau:

1. Sử dụng các phương thức HTTP một cách rõ ràng. [9]

2. Phi trạng thái. [9]

3. Hiển thị cấu trúc thư mục như các Urls. [9]

4. Truyền tải JavaScript Object Notation (JSON), XML hoặc cả hai. [9]

* **HTTP trong RESTful**

REST đặt ra một quy tắc đòi hỏi lập trình viên xác định rõ ý định của mình thông qua các phương thức của HTTP. Thông thường ý định đó bao gồm lấy dữ liệu, chèn dữ liệu, cập nhập dữ liệu hoặc xóa dữ liệu. Vậy khi bạn muốn thực hiện một trong các ý định trên hãy lưu ý các quy tắc sau:

* Để tạo một tài nguyên trên máy chủ, bạn cần sử dụng phương thức POST. [9]
* Để truy xuất một tài nguyên, sử dụng GET. [9]
* Để thay đổi trạng thái một tài nguyên hoặc để cập nhật nó, sử dụng PUT. [9]
* Để huỷ bỏ hoặc xoá một tài nguyên, sử dụng DELETE. [9]

Chú ý rằng các nguyên tắc ở trên là không bắt buộc, thực tế bạn có thể sử dụng phương thức GET để yêu cầu lấy dữ liệu, chèn, sửa hoặc xóa dữ liệu trên Server. Tuy nhiên REST đưa ra các nguyên tắc ở trên mục đích đưa mọi thứ trở lên rõ ràng và dễ hiểu. [9]

### Bootstrap

#### Giới thiệu về Bootstrap

Bootstrap là một nền tảng (framework) miễn phí, mã nguồn mở, dựa trên HTML, CSS và Javascript, nó được tạo ra để xây dựng các giao diện Website tương thích với tất cả các thiết bị có kích thước màn hình khác nhau. [10]

Bootstrap bao gồm những cái cơ bản có sẵn như: typography, forms, buttons, tables, navigation, modals, image carousels và nhiều thứ khác. Nó cũng có nhiều Component, Javascript hỗ trợ cho việc thiết kế Reponsive của bạn dễ dàng, thuận tiện và nhanh chóng hơn. [10]

Hiện nay Bootstrap là một trong những framework được sử dụng nhiều nhất trên thế giới để tạo ra các Responsive Website. Bootstrap đã tạo ra một tiêu chuẩn riêng, và rất được các lập trình viên ưu chuộng. Về cơ bản Bootstrap có 3 ưu điểm:

* Dễ sử dụng: Vì Bootstrap được xây dựng trên HTML, CSS và Javascript. [10]
* Responsive: Bootstrap đã xây dựng sẵn các "Responsive Css" tương thích với các thiết bị khác nhau, vì vậy bạn chỉ cần học cách sử dụng chúng. Tính năng này giúp tiết kiệm rất nhiều thời gian cho các người dùng khi tạo ra các Website thân thiện. [10]
* Tương thích với các trình duyệt: Nó tương thích với tất cả các trình duyệt (Chrome, Firefox, Internet Explorer, Safari, Opera). Tuy nhiên, với IE, Bootstrap 4 chỉ hỗ trợ từ IE10 trở lên. [10]

#### Giới thiệu về Responsive

Responsive Web Design (RWD) là thuật ngữ ám chỉ cách thiết kế trang web hiển thị tương thích với mọi kích thước thiết bị và là xu hướng mới theo đó quy trình thiết kế và phát triển web. Tức là bố cục trang web sẽ tự đáp ứng theo hành vi người dùng và môi trường hiển thị. Môi trường này chính là kích thước của trình duyệt, kích thước hoặc hướng xoay thiết bị. Và thiết bị ở đây đa phần là các thiết bị di động như smartphone hoặc tablet. Các bạn có thể thử thay đổi kích thước trình duyệt, xem trên di động, tablet,... mà vẫn đảm bảo giao diện hiển thị tốt. [10]

Để làm được điều đó chúng ta sẽ sử dụng linh hoạt kết hợp các kỹ thuật bao gồm flexible grid, responsive image và CSS media query. Khi người dùng chuyển từ máy tính xách tay của họ sang iPad hay iPhone, trang web sẽ tự động chuyển đổi để phù hợp với kích thước màn hình và kịch bản xử lý. Nói cách khác, các trang web cần phải có công nghệ tự động đáp ứng theo thiết bị của người dùng. Điều này sẽ loại bỏ sự cần thiết cho nhiều thiết kế web khác nhau và giảm thiểu thời gian cũng như chi phí thiết kế web. [10]

### jQuery

jQuery là một thư viện JavaScript đa tính năng, nhỏ gọn, nhanh, được tạo bởi John Resig vào năm 2006. jQuery đơn giản hóa việc duyệt tài liệu HTML, xử lý sự kiện, hoạt ảnh và tương tác Ajax để phát triển web nhanh chóng. Các phân tích web đã chỉ ra rằng, jQuery là thư viện JavaScript được triển khai rộng rãi nhất. [11]

jQuery là một bộ công cụ JavaScript được thiết kế để đơn giản hóa các tác vụ khác nhau bằng cách viết ít code hơn. Dưới đây liệt kê một số tính năng tối quan trọng được hỗ trợ bởi jQuery:

* Thao tác DOM − jQuery giúp dễ dàng lựa chọn các phần tử DOM để traverse (duyệt) một cách dễ dàng như sử dụng CSS, và chỉnh sửa nội dung của chúng bởi sử dụng phương tiện Selector mã nguồn mở, mà được gọi là Sizzle. [11]
* Xử lý sự kiện − jQuery giúp tương tác với người dùng tốt hơn bằng việc xử lý các sự kiện đa dạng mà không làm cho HTML code rối tung lên với các Event Handler. [11]
* Hỗ trợ AJAX − jQuery giúp bạn rất nhiều để phát triển một site giàu tính năng và phản hồi tốt bởi sử dụng công nghệ AJAX. [11]
* Hiệu ứng động − jQuery đi kèm với rất nhiều hiệu ứng động đẹp mà bạn có thể sử dụng cho các website của mình. [11]
* Gọn nhẹ − jQuery là thư viện gọn nhẹ - nó chỉ có kích cỡ khoảng 19KB (gzipped). [11]
* Được hỗ trợ hầu hết bởi các trình duyệt hiện đại − jQuery được hỗ trợ hầu hết bởi các trình duyệt hiện đại, và làm việc tốt trên IE 6.0+, FF 2.0+, Safari 3.0+, Chrome và Opera 9.0+. [11]
* Cập nhật và hỗ trợ các công nghệ mới nhất − jQuery hỗ trợ CSS3 Selector và cú pháp XPath cơ bản. [11]

## BỐ CỤC BÀI BÁO CÁO

Nội dung của đề tài gồm ba phần:

**Phần giới thiệu:** Thực hiện việc mô tả bài toán, mục tiêu cần đạt và hướng giải quyết.

**Phần nội dung:** Giải thích các khái niệm, phương pháp, kết quả nghiên cứu lý thuyết vận dụng vào đề tài, đặc tả hệ thống, vẽ các mô hình, các ràng buộc toàn vẹn, các sơ đồ chức năng, lưu đồ giải thuật giải quyết vấn đề. Phần nội dung gồm có ba chương:

**Chương 1:** Mô tả bài toán.

**Chương 2:** Thiết kế và cài đặt giải pháp.

**Chương 3:** Kiểm thử và đánh giá.

**Phần kết luận:** Thực hiện việc nhận xét kết quả đạt được, nêu lên ưu điểm cũng như nhược điểm và hướng phát triển của đề tài.

# PHẦN 2: NỘI DUNG

## CHƯƠNG 1: MÔ TẢ BÀI TOÁN

### MỤC TIÊU

Hiểu được khái niệm và cách thức hoạt động của web service và web client, hệ quản trị cơ sở dữ liệu mã nguồn mở SQLServer hỗ trợ đa nền tảng được thiết kế theo hướng đối tượng, ASP .NET Framework hỗ trợ thiết kế giao diện, trình thông dịch JavaScript.

Phân tích chính xác và thiết kế đầy đủ, chi tiết các mô hình hỗ trợ cho việc xây dựng đề tài.

Xây dựng website rao vặt giúp người dùng dễ dàng giới thiệu, rao bán và tiếp cận với sản phẩm như xem danh sách sản phẩm theo danh mục, xem thông tin chi tiết sản phẩm, xem các gợi ý sản phẩm khuyến mãi và sản phẩm bán chạy, tìm kiếm sản phẩm, đối với khách hàng có đăng ký tài khoản sẽ được thêm sản phẩm vào giỏ hàng, đặt mua sản phẩm, theo dõi đơn hàng, đánh giá sản phẩm và quản lý tài khoản cá nhân.

Xây dựng website quản lý thông tin để người quản trị dễ dàng quản lý những bài đăng sản phẩm, những thông tin về khách hàng, thông tin cửa hàng, danh mục sản phẩm, sản phẩm, nhà cung cấp, phương thức thanh toán, hóa đơn nhập hàng, khuyến mãi, xem báo cáo thống kê.

### PHẠM VI SẢN PHẨM

Sau khi hoàn thành quá trình nghiên cứu và xây dựng, website rao vặt dành cho sinh viên hỗ trợ người dùng có nhu cầu với các sản phẩm phục vụ nhu cầu chủ yếu cho sinh viên và các sản phẩm liên quan.

Đối với người quản trị, website giúp người quản trị quản lý các bài đăng, các thông tin về những đối tượng cần thiết trong quá trình kinh doanh như: sinh viên, học sinh, danh mục sản phẩm, các sản phẩm phục vụ học sinh, sinh viên, khuyến mãi, đánh giá của khách hàng, thống kê doanh thu một cách dễ dàng.

Đối với người tiêu dùng, website hỗ trợ khách hàng trong việc mua bán một cách nhanh chóng, tiện lợi, tiết kiệm được thời gian. Khách hàng tự chủ về phương thức thanh toán và hình thức vận chuyển trong quá trình đặt mua sản phẩm.

### MÔI TRƯỜNG VẬN HÀNH

Đề tài được xây dựng, phát triển và hoạt động trên PC có kết nối internet.

### CÁC RÀNG BUỘC VỀ THỰC THI VÀ CÔNG NGHỆ SỬ DỤNG

#### Các ràng buộc về thực thi

Đề tài được thực thi trên PC có kết nối internet, ứng với mỗi quyền sẽ có các chức năng nhất định.

Khách hàng được xem các thông tin sản phẩm và những gợi ý sản phẩm, tìm kiếm sản phẩm. Khách hàng có tài khoản sẽ được thêm sản phẩm vào giỏ hàng, đặt mua sản phẩm, theo dõi đơn hàng, đánh giá sản phẩm và quản lý tài khoản cá nhân.

Người quản trị quản lý những bài đăng, thông tin về khách hàng, thông tin các gian hàng, danh mục sản phẩm, sản phẩm, nhà cung cấp, xem báo cáo thống kê.

#### Công nghệ sử dụng

Các công nghệ để phát triển website như là:

* Powerdesigner 16.5.
* Canvas.
* ASP .NET Core.
* C#.
* SQLSERVER.
* Javascript, HTML, CSS,…

## CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP



### 2.1. Xây dựng các lớp thực thể

#### 2.1.1. Thực thể SANPHAM

Bảng sản phẩm (SANPHAM): Đây là thực thể quan trọng lưu thông tin sản phẩm. Mỗi sản phẩm sẽ do một người dùng đăng lên trang để giới thiệu, rao bán gồm có các thông tin sau: Mã sản phẩm, mã loại sản phẩm, mã người đăng, tên sản phẩm, giá bán, ngày đăng tin, thời gian đã sử dụng sản phẩm, số lượng còn lại, số lượng đã bán, hãng sản xuất, mô tả sản phẩm, lượt xem, tình trạng sản phẩm.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| SP\_MSSP | char(36) | Khóa chính | Mã sản phẩm |
| LOAI\_MALOAI | int | Khóa ngoại | Mã loại sản phẩm |
| SV\_MSSV | char(8) | Khóa ngoại | Mã người đăng |
| SP\_TENSP | nvarchar(200) |  | Tên sản phẩm |
| SP\_NGAYDANG | datetime |  | Ngày đăng tin |
| SP\_THOIGIANSUDUNG | int |  | Thời gian đã sử dụng sản phẩm |
| SP\_GIA | float |  | Giá bán |
| SP\_CONLAI | int |  | Số lượng còn lại |
| SP\_DABAN | int |  | Số lượng đã bán |
| SP\_HANGSX | nvarchar(200) |  | Hãng sản xuất |
| SP\_MOTA | nvarchar(4000) |  | Mô tả sản phẩm |
| SP\_LUOTXEM | int |  | Lượt xem |
| SP\_TINHTRANG | bit |  | Tình trạng sản phẩm |

#### 2.1.2. Thực thể SINHVIEN

Bảng sinh viên (SINHVIEN): Đây là thực thể quan trọng nhất vì nó lưu thông tin của người dùng. Mỗi người dùng có thể đăng ký một tài khoản, có một hoặc nhiều hóa đơn, đăng bán hoặc đặt một hoặc nhiều sản phẩm và gồm có các thông tin sau: Mã người dùng, mật khẩu, họ tên người dùng, tên hiển thị, ngày tạo tài khoản, lần hoạt động cuối, địa chỉ người dùng, số điện thoại người dùng, email người dùng, tình trạng người dùng, quyền truy cập người dùng.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| SV\_MSSV | char(8) | Khóa chính | Mã người dùng |
| LOAI\_SV\_MATKHAU | char(64) |  | Mật khẩu |
| SV\_HOTEN | nvarchar(200) |  | Họ tên người dùng |
| SV\_TENHIENTHI | nvarchar(200) |  | Tên hiển thị |
| SV\_NGAYTAOTK | datetime |  | Ngày tạo tài khoản |
| SV\_LANHDCUOI | date |  | Lần hoạt động cuối |
| SV\_DIACHIGIAOHANG | nvarchar(500) |  | Địa chỉ người dùng |
| SV\_SDT | char(10) |  | Số điện thoại người dùng |
| SV\_EMAIL | nvarchar(200) |  | Email người dùng |
| SV\_TINHTRANG | bit |  | Tình trạng người dùng |
| SV\_ADMIN | bit |  | Quyền truy cập người dùng |

#### 2.1.3. Thực thể LOAIMATHANG

Bảng thể loại mặt hàng (LOAIMATHANG): Thực thể chứa thông tin thể loại mặt hàng. Mỗi thể loại thuộc một danh mục sản phẩm, có một hoặc nhiều sản phẩm thuộc thể loại mặt hàng và gồm các thông tin sau: Mã loại mặt hàng, tên loại mặt hàng.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| MH\_MAMH | int | Khóa chính | Mã loại mặt hàng |
| MH\_TENMH | nvarchar(200) |  | Tên loại mặt hàng |

#### 2.1.4. Thực thể LOAISANPHAM

Bảng thể loại sản phẩm (LOAISANPHAM): Thực thể chứa thông tin thể loại sản phẩm. Mỗi loại sản phẩm thuộc một loại mặt hàng, có một hoặc nhiều sản phẩm thuộc thể loại sản phẩm và gồm các thông tin sau: Mã loại sản phẩm, mã loại mặt hàng, mã giá vận chuyển, tên loại sản phẩm.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| LOAI\_MALOAI | int | Khóa chính | Mã loại sản phẩm |
| MH\_MAMH | int | Khóa ngoại | Mã loại mặt hàng |
| SHIP\_MA | int | Khóa ngoại | Mã giá vận chuyển |
| LOAI\_TENLOAI | nvarchar(200) |  | Tên loại sản phẩm |

#### 2.1.5. Thực thể CHAT

Bảng chat (CHAT): Thực thể chứa thông tin các cuộc trò chuyện giữ những người dùng khi cần trao đổi, mua bán một hoặc nhiều sản phẩm và gồm các thông tin sau: mã người gửi tin nhắn, mã người nhận tin nhắn, thời điểm chat, nội dung chat, tình trạng đã đọc tin nhắn.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| SV\_MSSV\_G | char(8) | Khóa chính | Mã người gửi tin nhắn |
| SV\_MSSV\_N | char(8) | Khóa chính | Mã người nhận tin nhắn |
| CHAT\_THOIGIAN | datetime | Khóa chính | Thời điểm chat |
| CHAT\_NOIDUNG | nvarchar(4000) |  | Nội dung chat |
| CHAT\_DADOC | bit |  | Tình trạng đã đọc tin nhắn |

#### 2.1.6. Thực thể CHITIETHOADON

Bảng chi tiết hóa đơn (CHITIETHOADON): Thực thể chứa thông tin hóa đơn. Mỗi hóa đơn được tạo ra khi có một người dùng đặt một hoặc nhiều sản phẩm và gồm các thông tin: Mã hóa đơn, mã sản phẩm, số lượng sản phẩm, tổng hóa đơn.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| SP\_MSSP | char(36) | Khóa chính | Mã sản phẩm |
| HD\_MSHD | char(36) | Khóa chính | Mã hóa đơn |
| CTHD\_SOLUONG | int |  | Số lượng sản phẩm |
| SP\_GIABAN | float |  | Tổng hóa đơn |

#### 2.1.7. Thực thể GIOHANG

Bảng giỏ hàng (GIOHANG): Thực thể chứa thông tin về giỏ hàng các sản phẩm đã được người dùng thêm vào giỏ hàng. Mỗi giỏ hàng gồm các thông tin sau: Mã loại mặt hàng, tên loại mặt hàng.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| SV\_MSSV | char(8) | Khóa chính | Mã người dùng |
| SP\_MSSP | char(36) | Khóa chính | Mã sản phẩm |
| GH\_SOLUONG | int |  | Số lượng sản phẩm |

#### 2.1.8. Thực thể HINHANH

Bảng hình ảnh (HINHANH): Thực thể chứa thông tin biểu diễn hình ảnh sản phẩm. Mỗi hình ảnh chỉ dành cho một sản phẩm và gồm các thông tin sau: Mã hình ảnh, mã sản phẩm, mã người dùng, link hình ảnh.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| HA\_MSHA | char(36) | Khóa chính | Mã hình ảnh |
| SV\_MSSV | char(8) | Khóa ngoại | Mã người dùng |
| SP\_MSSP | char(36) | Khóa ngoại | Mã sản phẩm |
| HA\_LINK | varchar(500) |  | Link hình ảnh |

#### 2.1.9. Thực thể HOADONMUA

Bảng hóa đơn mua (CHITIETHOADON): Thực thể chứa thông tin hóa đơn. Mỗi hóa đơn được tạo ra khi có một người dùng đặt một hoặc nhiều sản phẩm và gồm các thông tin: Mã hóa đơn, tên người nhận, tình trạng hóa đơn, ngày đặt hàng, tổng hóa đơn, địa chỉ nhận hàng, số điện thoại người nhận, tên người nhận.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| HD\_MSHD | char(36) | Khóa chính | Mã hóa đơn |
| TT\_MSTT | int | Khóa ngoại | Tình trạng hóa đơn |
| HD\_NGAYMUA | datetime |  | Ngày đặt hàng |
| HD\_TONGGIA | float |  | Tổng hóa đơn |
| HD\_DIACHI | nvarchar(500) |  | Địa chỉ nhận hàng |
| HD\_SDT | char(10) |  | Số điện thoại người nhận |
| HD\_NGUOINHAN | nvarchar(200) |  | Tên người nhận |

#### 2.1.10. Thực thể KHUYENMAI

Bảng khuyến mãi (KHUYENMAI): Thực thể chứa thông tin các khuyễn mãi của các sản phẩm đang đăng bán. Mỗi khuyến mãi dành cho một hoặc nhiều sản phẩm và gồm các thông tin sau: Mã khuyến mãi, mã sản phẩm, phần trăm khuyến mãi.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| KM\_MSKM | char(36) | Khóa chính | Mã khuyến mãi |
| SP\_MSSP | char(36) |  | Mã sản phẩm |
| KM\_PHANTRAM | float |  | Phần trăm khuyến mãi |

#### 2.1.11. Thực thể NHAXNETNGUOIBAN

Bảng nhận xét người bán (NHANXETNGUOIBAN): Thực thể chứa thông tin phản hồi, nhận xét góp ý của người dùng. Mỗi nhận xét gồm các thông tin sau: Mã phản hồi, họ tên khách hàng phản hồi, email khách hàng phản hồi, ngày phản hồi, nội dung phản hồi, trạng thái phản hồi.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| KM\_MSKM | char(36) | Khóa chính | Mã khuyến mãi |
| SP\_MSSP | char(36) |  | Mã sản phẩm |
| KM\_PHANTRAM | float |  | Phần trăm khuyến mãi |

#### 2.1.12. Thực thể TIENSHIP

Bảng tiền vận chuyển (TIENSHIP): Thực thể gồm các thông tin sau: Mã loại vận chuyển, giá của loại vận chuyển.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| SHIP\_MA | int | Khóa chính | Mã vận chuyển |
| SHIP\_GIA | float |  | Giá vận chuyển |

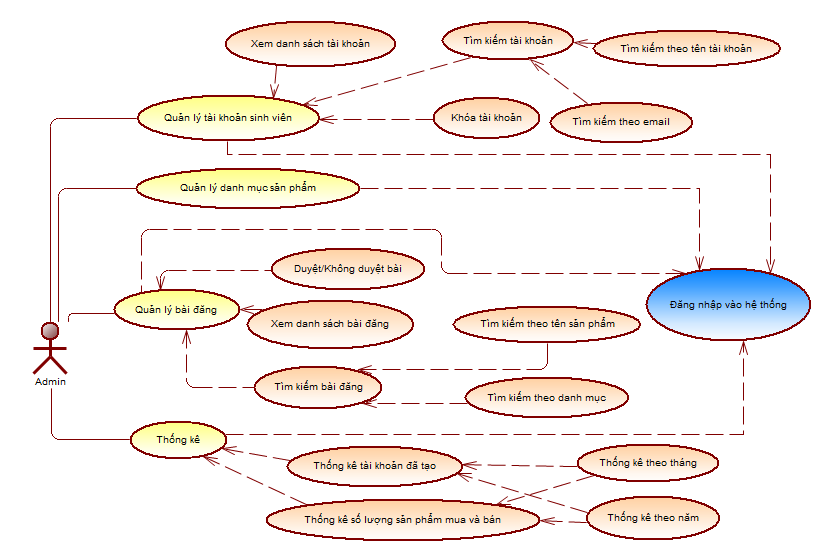
#### 2.1.13. Thực thể TINHTRANGHOADON

Bảng tính trạng hóa đơn (TINHTRANG): Thực thể chứa tình trạng của đơn hàng mà người mua đã đặt gồm các thông tin sau: Mã tình trạng đơn hàng, tình trạng đơn hàng.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| TT\_MSTT | int | Khóa chính | Mã tình trạng |
| TT\_TRANGTHAI | nvarchar(20) |  | Tên tình trạng |

### 2.2. Sơ đồ Use Case

#### 2.2.1. Sơ đồ Use Case của actor người quản trị

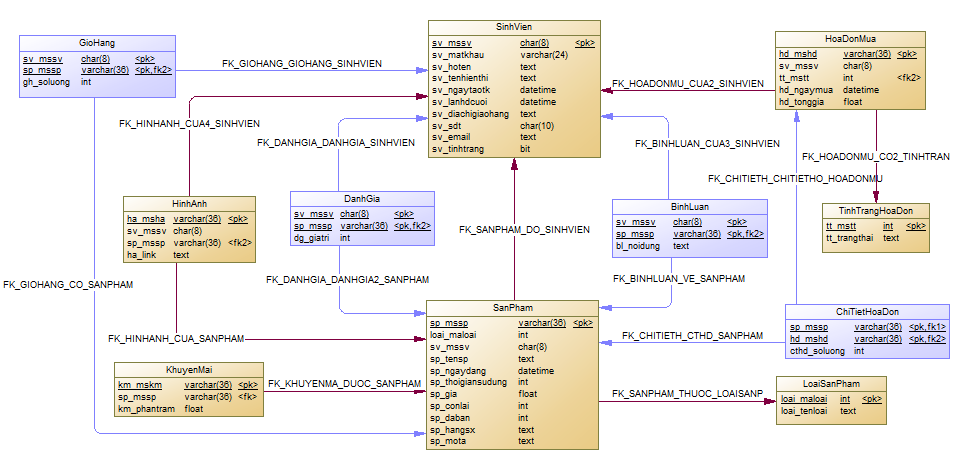


#### 2.2.2. Sơ đồ Use Case của actor khách hàng

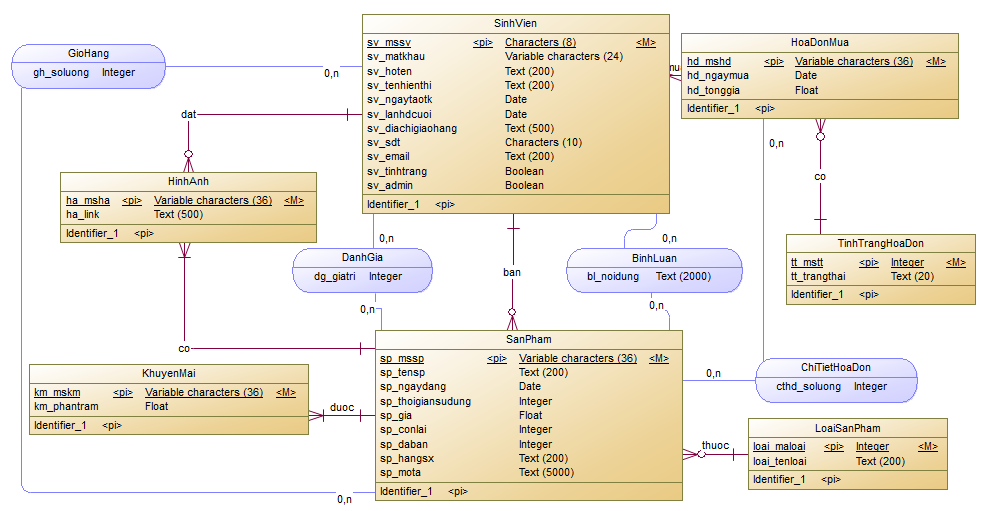


### 2.3. Thiết kế cơ sở dữ liệu

#### 2.3.1. Mô hình MLD



#### 2.3.2. Mô hình thực thể mối quan hệ (ERD – Entity Relationship Diagram)



BỐ CỤC NIÊN LUẬN

Bố cục của quyển niên luận bao gồm các phần:

Phần 1: GIỚI THIỆU

* Đặt vấn đề
* Lịch sử giải quyết vấn đề
* Mục tiêu đề tài
* Đối tượng và phạm vi nghiên cứu
* Nội dung nghiên cứu
* Những đóng góp chính
* Bố cục niên luận

Phần 2: NỘI DUNG

* Chương 1: CƠ SỞ LÝ THUYẾT
* Chương 2: MÔ TẢ BÀI TOÁN
* Chương 3: THIẾT KẾ VÀ CÀI ĐẶT BÀI TOÁN
* Chương 4: KẾT QUẢ THỰC HIỆN

Phần 3: KẾT LUẬN

* Kết quả đạt được
* Hướng phát triển

# PHẦN 2: NỘI DUNG

## CHƯƠNG 1: MÔ TẢ BÀI TOÁN

### MÔ TẢ CHI TIẾT BÀI TOÁN

### PHÂN TÍCH, ĐÁNH GIÁ CÁC GIẢI PHÁP CÓ LIÊN QUAN

### LỰA CHỌN GIẢI PHÁP

#### Microsoft Visual Studio 2019

Microsoft Visual Studio là một môi trường phát triển tích hợp (IDE) từ Microsoft. Microsoft Visual Studio còn được gọi là "Trình soạn thảo mã nhiều người sử dụng nhất thế giới ", được dùng để lập trình C++ và C# là chính. Nó được sử dụng để phát triển chương trình máy tính cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như Windows API, Windows Forms, Windows Presentation Foundation, Windows Store và Microsoft Silverlight. Nó có thể sản xuất cả hai ngôn ngữ máy và mã số quản lý.

Visual Studio bao gồm một trình soạn thảo mã hỗ trợ IntelliSense cũng như cải tiến mã nguồn. Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng giao diện ứng dụng, thiết kế web, thiết kế lớp và thiết kế giản đồ cơ sở dữ liệu. Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các hệ thống quản lý phiên bản (như Subversion) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các miền ngôn ngữ cụ thể hoặc bộ công cụ dành cho các khía cạnh khác trong quy trình phát triển phần mềm.

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có C, C++ và C++/CLI (thông qua Visual C++), VB.NET (thông qua Visual Basic.NET), C# (thông qua Visual C#) và F# (như của Visual Studio 2010). Hỗ trợ cho các ngôn ngữ khác như J++/J#, Python và Ruby thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ XML/XSLT, HTML/XHTML, JavaScript và CSS. [2]

#### ASP .NET MVC

ASP.NET MVC là một khung ứng dụng web được phát triển bởi Microsoft, thực hiện mô hình model – view – controller (MVC). Nó là phần mềm mã nguồn mở, ngoài thành phần ASP.NET Web Forms và là phần mềm độc quyền. ASP.NET Core kể từ đó đã được phát hành, hợp nhất ASP.NET, ASP.NET MVC, ASP.NET Web API và ASP.NET Web Pages (một nền tảng chỉ sử dụng các trang Razor). [3]

[ASP.NET MVC](https://techmaster.vn/khoa-hoc/25503/xay-dung-mang-xa-hoi-chia-se-anh-voi-aspnet-mvc) là một framework tuyệt vời hỗ trợ pattern MVC cho ASP.NET. Nếu bạn muốn hiểu ASP.NET MVC làm việc như thế nào, bạn cần phải có một sự hiểu biết rõ ràng về mô hình MVC. MVC là cụm từ viết tắt của Model-View-Controller, nó phân chia pattern của ứng dụng thành 3 phần - model, controller và view.

Model giúp lưu trữ dữ liệu của ứng dụng và trạng thái của nó. Nó là một cơ sở dữ liệu hoặc cũng có thể chỉ là một tập tin XML.

View được coi là một giao diện người dùng được sử dụng bởi khách truy cập trang web của bạn để nhìn thấy các dữ liệu. Các trang ASPX thường được sử dụng để hiển thị view trong các ứng dụng ASP.NET MVC.

Controller chịu trách nhiệm xử lý các tương tác của người dùng với trang web. Nó được sử dụng để xác định loại view nào cần phải được hiển thị. Controller cũng được sử dụng cho mục đích giao tiếp với model.

Framework này là khá nhẹ và cung cấp khả năng kiểm thử, trong đó tích hợp với các tính năng hiện có của ASP.NET như xác thực (authentication) dựa trên membership và cả các master page.

Hiện nay Techmaster cũng đang có khóa học ["Xây dựng mạng xã hội chia sẻ ảnh với ASP.NET MVC"](https://techmaster.vn/khoa-hoc/25503/xay-dung-mang-xa-hoi-chia-se-anh-voi-aspnet-mvc). Giảng viên là anh Nguyễn Thanh Tùng, kiến trúc sư giải pháp .NET tại công ty MISA. Anh Tùng còn là chuyên gia công nghệ Microsoft (Microsoft Most Valuable Professional).

#### Microsoft SQL Server Management Studio 2018

SQL Server Management Studio (SSMS) là một môi trường tích hợp để quản lý bất kỳ cơ sở hạ tầng SQL nào, từ SQL Server đến Azure SQL Database. SSMS cung cấp các công cụ để cấu hình, giám sát và quản trị các phiên bản của SQL Server và cơ sở dữ liệu. SSMS dùng để triển khai, giám sát và nâng cấp các thành phần tầng dữ liệu được các ứng dụng của người sử dụng, xây dựng các truy vấn và tập lệnh.

SSMS có thể dùng để truy vấn, thiết kế và quản lý cơ sở dữ liệu và kho dữ liệu của người dùng, cho dù chúng ở đâu - trên máy tính cục bộ hoặc trên đám mây. [4]

#### Bootstrap

Bootstrap là bộ công cụ mã nguồn mở front-end phổ biến nhất thế giới, có các biến Sass và mixin, hệ thống lưới đáp ứng, các thành phần được tạo sẵn rộng rãi và các plugin JavaScript mạnh mẽ. [5]



Hình 3: Logo Bootstrap

## CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP

### MÔ HÌNH USECASE CÁC MỨC

#### Mô hình usecase mức 0

### MÔ HÌNH TẬP THỰC THỂ CÁC MỐI QUAN HỆ

#### Mô hình ER

#### Mô tả các quan hệ chính yếu

### MÔ HÌNH VẬT LÝ

### MÔ HÌNH CƠ SỞ DỮ LIỆU QUAN HỆ

#### Mô hình cơ sở dữ liệu quan hệ

#### Mô tả cơ sở dữ liệu

### SƠ ĐỒ CHỨC NĂNG

### MÔ HÌNH TUẦN TỰ CÁC HOẠT ĐỘNG CHÍNH

## CHƯƠNG 3: KIỂM THỬ VÀ ĐÁNH GIÁ

### KIỂM THỬ

### ĐÁNH GIÁ

### PHẦN 3: KẾT LUẬN

### KẾT QUẢ ĐẠT ĐƯỢC

### HƯỚNG PHÁT TRIỂN

### TÀI LIỆU THAM KHẢO