**LỜI MỞ ĐẦU**

Khoa học công nghệ đang ngày càng phát triển, con người ứng dụng Internet vào cuộc sống, hoạt động sản xuất kinh doanh trở nên phổ biến và rộng rãi. Mạng Internet là phương thức truyền tải thông tin nhanh nhất tới khách hàng mọi lúc, mọi nơi. Website là một trong những công cụ hỗ trợ hiệu quả cho hoạt động quảng bá thông tin, quảng cáo sản phẩm dịch vụ kinh doanh và được xem là cánh tay đắc lực cho chiến lược Marketing của mỗi doanh nghiệp. Chính vì vậy, vai trò của website đối với doanh nghiệp là rất quan trọng cho sự tồn tại và phát triển của các đơn vị kinh doanh trong mọi lĩnh vực ở thời điểm hiện tại cũng như trong tương lai.

Website ở đây có thể được coi như nét đặc trưng của một công ty, là bộ mặt của doanh nghiệp mà từ đó mọi người có thể có những đánh giá nhất định, hiểu thêm được cách cơ cấu các bộ phận trong công ty doanh nghiệp sở hữu website đó. Khi sở hữu một website được thiết kế chuyên nghiệp sẽ mang đến cho hoạt động kinh doanh của bạn những lợi thế to lớn. Website giúp các đơn vị kinh doanh trong mọi lĩnh vực như: nhà hàng, khách sạn, bệnh viện, trường học, thời trang, dịch vụ, …cung cấp đày đủ thông tin, giới thiệu sản phẩm, dịch vụ đến khách hàng gần xa một cách nhanh chóng thông qua Internet. Website được xem là công cụ tối ưu hỗ trợ cho hoạt động Marketing góp phần quảng bá rộng rãi hình ảnh doanh nghiệp đem chất lượng sản phẩm, dịch vụ đến khách hàng nhằm xây dựng thương hiệu, tạo dựng uy tín, đồng thời nâng cao sức mạnh cạnh tranh cho các đơn vị kinh doanh.

Nhận thức được điều đó, em đã chọn đề tài “Xây dựng website bán hàng điện tử và đặt lịch sửa chữa tại nhà” để nghiên cứu, xây dựng và viết báo cáo, qua đó với sự giúp đỡ của lãnh đạo Công ty và các anh chị trong phòng kỹ thuật Công ty, em hy vọng sẽ hiểu thêm quá trình hoạt động website từ những bước cơ bản nhất.

Nội dung bài báo cáo gồm 3 chương:

* Chương 1: Giới thiệu tổng quan
* Chương 2: Phân tích thiết kế hệ thống
* Chương 3: Cài đặt và triển khai website bán hàng điện tử và đặt lịch sửa chữa tại nhà

**Mục lục**

|  |  |
| --- | --- |
|  | Trang |
| Danh mục các bảng |  |
|  |  |
| Danh mục hình vẽ |  |
| Hình 1.1 Mô hình xây dựng phần mềm bằng ngôn ngữ truyền thống  Hình 1.2  Hình 1.3  Hình 1.4  Hình 1.5  Hình 1.6  Hình 1.7  Hình 1.8  Hình 1.9  Hình 2.1  Hình 2.2  Hình 2.3  Hình 2.4  Hình 2.5  Hình 2.6  Hình 2.7  Hình 2.8  Hình 2.9  Hình 2.10  Hình 2.11  Hình 2.12  Hình 2.12  Hình 2.12  Hình 2.12  Hình 2.12  Hình 2.12  Hình 2.12  Hình 2.12  Hình 2.20  Hình 2.20  Hình 2.20  Hình 2.20  Hình 2.20  Hình 2.20  Hình 2.20  Hình 2.20  Hình 2.20  Hình 2.20  Hình 2.30  Hình 2.30  Hình 2.30  Hình 2.30  Hình 2.30  Hình 2.30  Hình 2.30  Hình 2.30  Hình 2.30  Hình 2.30  Hình 2.30 |  |

[Chương 1. Giới thiệu tổng quan 5](#_Toc6785802)

[1.1. Giới thiệu chung 5](#_Toc6785803)

[1.2. Nghiên cứu hiện trạng, giải quyết vấn đề 5](#_Toc6785804)

[1.3. Giới thiệu công nghệ sử dụng 5](#_Toc6785805)

[1.3.1.Giới thiệu tổng quan về công nghệ .NET 5](#_Toc6785806)

[1.3.1.1. Sự ra đời của .NET 5](#_Toc6785807)

[1.3.1.2. .NET Framework là gì? 7](#_Toc6785808)

[1.3.1.3. Một số ưu điểm chính của .Net Framework 11](#_Toc6785809)

[1.3.2.Visual Studio 2017 11](#_Toc6785810)

[1.3.2.1. Giới thiệu về Visual Studio 11](#_Toc6785811)

[1.3.2.2. Giới thiệu môi trường tích hợp (IDE) của Visual Studio 14](#_Toc6785812)

[1.3.3.Miscrosoft SQL Server 2017 14](#_Toc6785813)

[1.3.3.1. SQL Server là gì? 14](#_Toc6785814)

[1.3.3.2. Lịch sử phát triển. 15](#_Toc6785815)

[1.3.3.3. Đặc điểm của SQL Server 15](#_Toc6785816)

[1.3.3.4. Đối tượng làm việc của SQL Server 16](#_Toc6785817)

[1.3.4. Giới thiệu về mô hình MVC trong phát triển Web 16](#_Toc6785818)

[1.3.5. Giới thiệu về Entity Framework 21](#_Toc6785819)

[Chương 2. Phân tích thiết kế hệ thống 22](#_Toc6785820)

[2.1. Xác định các tác nhân và ca sử dụng 22](#_Toc6785821)

[2.1.1.Các tác nhân 22](#_Toc6785822)

[2.1.2.Ca sử dụng 22](#_Toc6785823)

[2.1.3 Các biểu đồ ca sử dụng 23](#_Toc6785824)

[2.1.3.1. Ca sử dụng 23](#_Toc6785825)

[2.1.3.2. Chi tiết ca sử dụng 23](#_Toc6785826)

[2.1.4.Mô tả chi tiết các ca sử dụng 29](#_Toc6785827)

[2.1.4.1. Tạo tài khoản 29](#_Toc6785828)

[2.1.4.2. Đăng nhập 30](#_Toc6785829)

[2.1.4.3. Đặt lịch sửa chữa 31](#_Toc6785830)

[2.1.4.4. Thêm giỏ hàng 31](#_Toc6785831)

[2.1.4.5. Tìm kiếm sản phẩm 32](#_Toc6785832)

[2.1.4.6. Chi tiết sản phẩm 32](#_Toc6785833)

[2.1.4.7. Quản lý nhóm hàng 33](#_Toc6785834)

[2.1.4.8. Quản lý bài viết 34](#_Toc6785835)

[2.1.4.9. Quản lý nhóm bài viết 34](#_Toc6785836)

[2.1.4.10. Quản lý tài khoản 35](#_Toc6785837)

[2.1.4.11. Quản lý đặt hàng 36](#_Toc6785838)

[2.1.4.12. Quản lý đặt lịch sửa chữa 37](#_Toc6785839)

[2.1.4.13. Quản lý nhóm sản phẩm 37](#_Toc6785840)

[2.1.4.14. Quản lý sản phẩm 38](#_Toc6785841)

[2.1.4.15. Quản lý nhà sản xuất 39](#_Toc6785842)

[2.2. Biểu đồ lớp chi tiết 40](#_Toc6785843)

[2.2.1.Đăng nhập 40](#_Toc6785844)

[2.2.2.Đăng ký 42](#_Toc6785845)

[2.2.3.Mua hàng 43](#_Toc6785846)

[2.2.4.Quản lý tài khoản 44](#_Toc6785847)

[2.2.5.Quản lý sản phẩm 45](#_Toc6785848)

[2.2.6.Quản lý nhóm sản phẩm 48](#_Toc6785849)

[2.2.7.Quản lý nhóm bài viết 51](#_Toc6785850)

[2.2.8.Quản lý bài viết 54](#_Toc6785851)

[2.2.9.Quản lý đặt hàng 57](#_Toc6785852)

[2.2.10.Quản lý đặt lịch 59](#_Toc6785853)

[2.2.11.Quản lý nhà sản xuất 61](#_Toc6785854)

[2.3. Cơ sở dữ liệu 63](#_Toc6785855)

[2.3.1.Cơ sở dữ liệu quan hệ 63](#_Toc6785856)

[2.3.2.Chi tiết bảng dữ liệu 64](#_Toc6785857)

[2.3.2.1. Bảng Wards 64](#_Toc6785858)

[2.3.2.2. Bảng Provinces 64](#_Toc6785859)

[2.3.2.3. Bảng Districts 65](#_Toc6785860)

[2.3.2.4. Bảng ArticleGroups 65](#_Toc6785861)

[2.3.2.5. Bảng Article 65](#_Toc6785862)

[2.3.2.6. Bảng Feedback 66](#_Toc6785863)

[2.3.2.7. Bảng GuaranteeTimeProduct 66](#_Toc6785864)

[2.3.2.8. Bảng Producer 66](#_Toc6785865)

[2.3.2.9. Bảng Order 67](#_Toc6785866)

[2.3.2.10. Bảng OrderRepair 67](#_Toc6785867)

[2.3.2.11. Bảng Product 68](#_Toc6785868)

[2.3.2.12. Bảng ProductGroup 69](#_Toc6785869)

[2.3.2.13. Bảng ProductWarrantyCard 69](#_Toc6785870)

[2.3.2.14. Bảng Promotion 69](#_Toc6785871)

[2.3.2.15. Bảng Role 69](#_Toc6785872)

[2.3.2.16. Bảng Service 70](#_Toc6785873)

[2.3.2.17. Bảng ServiceWarrantyCard 70](#_Toc6785874)

[2.3.2.18. Bảng User 71](#_Toc6785875)

[Chương 3. Cài đặt và triển khai website quản lý nhà hàng 72](#_Toc6785876)

[3.1. Giao diện của khách hàng 72](#_Toc6785877)

[3.2. Giao diện quản trị 72](#_Toc6785878)

[3.3. Một số giao diện khác 72](#_Toc6785879)

**Chương 1: Giới thiệu tổng quan**

* 1. **Giới thiệu chung**

Dự án tập trung vào việc xây dựng một website bán hàng và quảng bá dịch vụ sửa chữa, lắp đặt thiết bị điện tử, điện lạnh. Từ việc kinh doanh và cung cấp dịch vụ truyền thống, chủ yếu dựa vào bảng quản cáo, thụ động. Vì vậy, việc xây dựng một trang web giúp quảng bá được thương hiệu, giúp cho việc quản lý sản phẩm một cách nhanh chóng, tối ưu, tiết kiệm nhân lực.

* 1. **Nghiên cứu hiện trạng, giải quyết vấn đề**

Hiện tại, việc kinh doanh sản phẩm điện tử online đã khá phổ biến, xong hầu hết chỉ cung cấp sản phẩm mà không chú trọng vào việc chăm sóc khách hàng hậu kỳ mua sản phẩm.

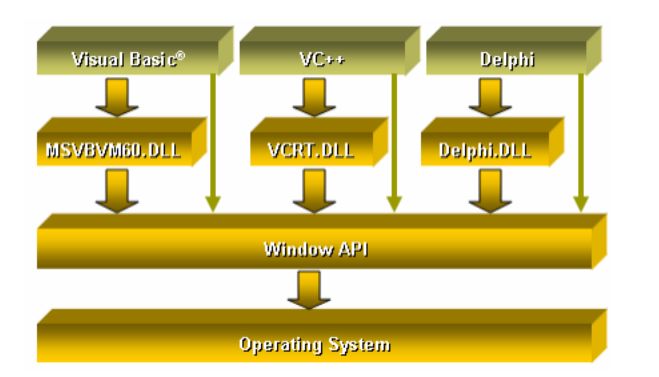
Đề xuất xây dựng một trang web giúp cho việc bán hàng thuận tiện, và có thể cung cấp dịch vụ bảo hành, sửa chữa cho khách hàng trong quá trình sử dụng sản phẩm và cả sản phẩm không do cửa hàng cung cấp.

* 1. Giới thiệu công nghệ sử dụng
     1. **Giới thiệu tổng quan về công nghệ .NET**
        1. **Sự ra đời của .NET**

Trước đây và cả ngày nay, trong lĩnh vực phát triển phần mềm có rất nhiều (hàng ngàn thậm chí hàng vạn) ngôn ngữ lâp trình được sử dụng để phát triển phần mềm (như Delphi, Ada, Cobol, Fortran, Basic, LISP, Prolog, Foxpro, Java, Pascal, C/C++, Visual Basic, VC++, C#...). Mỗi ngôn ngữ đều có những ưu và nhược điểm riêng, chẳng hạn Fortran là lựa chọn số một cho các tính toán khoa học; Prolog là lựa chọn rất tốt để phát triển các phần mềm thông minh (AI, Expert Systems…); Java có lợi thế phát triển các ứng dụng mạng, ứng dụng Mobile và độc lập hệ điều hành (Write One – Run Everywhere); Visual Basic tỏ ra dễ học và dễ phát triển các ứng dụng Winform; C# vượt trội bởi sự kết hợp giữa sức mạnh của C++ và sự dễ dàng của Visual Basic… Những ưu điểm có tính đặc thù của từng ngôn ngữ là điều đã được khẳng định. Tuy nhiên, điều mà ai cũng thấy rõ là rất khó để có thể tận dụng được sức mạnh của tất cả các ngôn ngữ lập trình trong một dự án phần mềm, chẳng hạn không thể hoặc rất khó khăn để viết một ứng dụng có sử dụng đồng thời cả ngôn ngữ Visual Basic và Java hay Foxpro với Delphi v.v… Nói cách khác, việc “liên thông” giữa các ngôn ngữ là gần như không thể. Cũng do sự khác biệt giữa các ngôn ngữ lập trình mà việc tiếp cận hay chuyển đổi sang ngôn ngữ lập trình mới sẽ tốn rất nhiều thời gian (Tuy rằng về tư tưởng và nguyên lý có tương tự nhau). Vì vậy, khi các dự án sử dụng ngôn ngữ lập trình khác nhau thì chi phí cho chuyển đổi/ học hỏi sẽ là rất lớn, gây lãng phí thời gian không cần thiết và chất lượng phần mềm chắc chắn không cao. Ngoài ra, cùng với sự phát triển như vũ bão của Internet thì mô hình phát triển ứng dụng cũng rất khác xưa. Các ứng dụng ngày nay không chỉ chạy riêng lẻ (stand-alone) trên máy tính PC mà còn có thể chạy trên môi trường mạng, cung cấp hay truy cập các dịch vụ từ xa (ứng dụng phân tán). Vai trò của phần mềm đã dần chuyển từ chỗ cung cấp các chức năng (Funtional) cụ thể sang cung cấp các dịch vụ (Services). 9 Từ những hạn chế trong quá trình phát triển phần mềm như đã nêu, đòi hỏi phải có một cách tiếp cận sao cho tối ưu nhất, vừa đảm bảo tốn ít chi phí chuyển đổi vừa đảm bảo nhiều người có thể tham gia cùng một dự án mà không nhất thiết phải viết trên cùng một ngôn ngữ lập trình, đồng thời ứng dụng phải hoạt động tốt trong môi trường mạng Internet. Đó chính là lý do để Microsoft cho ra công nghệ phát triển phần mềm mới .NET! Microsoft .NET là một nền tảng (Platform) phát triển ứng dụng mới và hoàn chỉnh nhất từ trước tới nay. Sự ra đời của Microsoft.NET có tính cách mạng, nó đem đến cho các nhà lập trình một phong cách phát triển phần mềm đột phá, khắc phục hầu hết các hạn chế trước đây của các ngôn ngữ lập trình. Việc sử dụng .NET không chỉ giúp phát triển các ứng dụng đơn lẻ mà còn có thể phát triển các ứng dụng phân tán ở qui mô rất lớn; .NET làm giảm thiểu thời gian phát triển ứng dụng, nâng cao rõ rệt chất lượng sản phẩm phần mềm. Phiên bản .NET đầu tiên (v 1.0) được Microsoft đưa ra thị trường vào năm 2001.

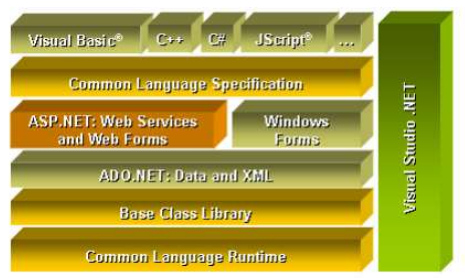
* + - 1. **.NET Framework là gì?**

Thông thường, mỗi ngôn ngữ lập trình đều có một tập các thư viện riêng, chẳng hạn: VC++ thì có thư viện chính là msvcrt.dll; Visual Basic thì có msvbvm60.dll …Các thư viện này chứa các hàm, thủ tục cơ bản của mỗi ngôn ngữ (ví dụ hàm, thủ tục xử lý xâu, xử lý toán học …). Tất cả những thứ này có ý nghĩa logic giống nhau nhưng về cách sử dụng hay cú pháp thì hầu như là khác nhau. Điều này khiến cho một lập trình viên C++ không thể áp dụng những kiến thức họ biết sang VB hoặc ngược lại. Hơn nữa, việc phát triển bộ thư viện riêng cho mỗi ngôn ngữ như vậy là quá dư thừa. Ý tưởng của Microsoft đó là KHÔNG xây dựng một tập thư viện riêng biệt cho từng ngôn ngữ lập trình mà sẽ xây dựng một bộ thư viện dùng CHUNG. Tập thư viện dùng chung này hình thành nên một bộ khung (Framework) để các lập trình viên viết ứng dụng trên bộ khung sẵn có đó. Bộ Khung này thực chất là một tập các thư viện được xây dựng sẵn, đáp ứng mọi nhu cầu phát triển các ứng dụng Desktop, Network, Mobile, web…

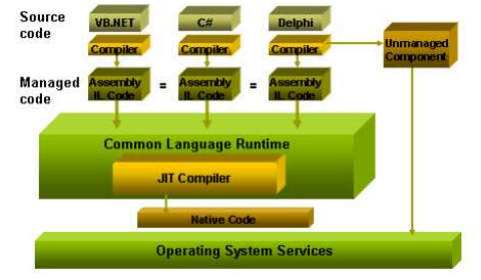


*Hình 1.1. Mô hình xây dựng phần mềm bằng ngôn ngữ truyền thống*

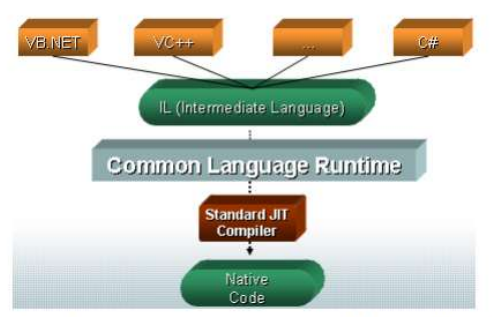
* Các thành phần và chức năng chính trong .Net Framework
  + Common Language Runtime (Trình thực thi ngôn ngữ chung): Sau khi ứng dụng được biên dịch ra file “Exe” (exe này khác với file exe thông thường. Nội dung của file exe này tuân theo một chuẩn/ngôn ngữ chung, dù là viết bằng C# hay VB.NET. Ngôn ngữ này gọi là ngôn ngữ chung), tiếp theo để file exe trung gian này có thể chạy được trên máy hiện hành thì cần phải được biên dịch ra mã máy tương ứng. Việc biên dịch và chạy được là nhờ Chương trình thực thi ngôn ngữ chung – CLR (Common Language Runtime).
  + Base Class Library: Là tập các thư viện chứa các lớp cơ bản để sử dụng trong tất cả các ngôn ngữ .NET. Ví dụ các lớp xử lý xâu, xử lý toán học…
  + ADO.NET: Là tập các thư viện chuyên dành cho thao tác với Cơ sở dữ liệu.
  + ASP.NET: Các thư viện dành cho phát triển các ứng dụng Web (webform).
  + Windows Forms: Các thư viện dành cho phát triển các ứng dụng Windows (winform). 11
  + Common Language Specification: Phần này có nhiệm vụ đặc tả ngôn ngữ chung để các chương trình viết trên các ngôn ngữ lập trình khác nhau phải tuân theo. Nói cách khác, biên dịch các chương trình viết trên các ngôn ngữ lập trình khác nhau về một ngôn ngữ thống nhất chung (Common Language). Nhờ điều này mà
  + Các ngôn ngữ lập trình.



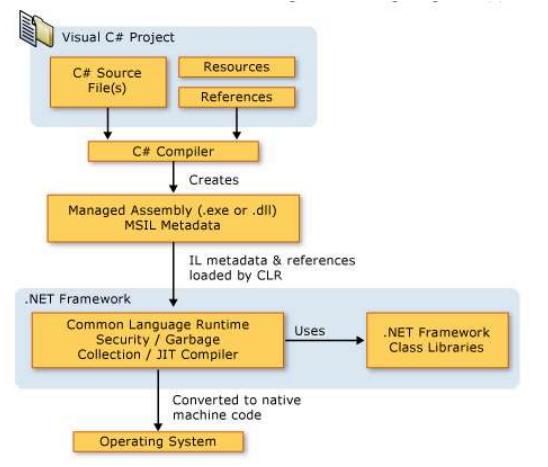
*Hình 1.2. Kiến trúc của .Net Framwork*



*Hình 1.3. Mô hình biên dịch và thực thi chương trình của ứng dụng .Net (1)*



*Hình 1.4. Mô hình biên dịch và thực thi chương trình của ứng dụng .Net (2)*



*Hình 1.5. Một cái nhìn khái quát về mô hình biên dịch và thực thi ứng dụng*

* + - 1. **Một số ưu điểm chính của .Net Framework**
* Tất cả các ngôn ngữ đều thừa hưởng một thư viện thống nhất. Khi sửa chữa hay nâng cấp thư viện này thì chỉ phải thực hiện một lần.
* Phong cách phát triển ứng dụng nhất quán và tương tự nhau giữa các ngôn ngữ lập trình. Có thể chuyển đổi sang ngôn ngữ lập trình .NET khác nhau một cách dễ dàng.
* Viết các ứng dụng webform không khác nhiều so với ứng dụng winform.
* Cung cấp một tập thư viện truy xuất CSDL thống nhất (ADO.NET) cho mọi ngôn ngữ .NET.
* Hỗ trợ cơ chế “Write one – Run everywhere” (Viết một lần chạy mọi nơi). Một ứng dụng viết bằng .NET có thể chạy trên bất cứ hệ điều hành nào mà không cần phải sửa lại code, miễn là máy đó có cài .NET framework.
* Cung cấp hệ thống kiểu chung (Common Type), do vậy đảm bảo tính thống nhất về kiểu dữ liệu giữa các ngôn ngữ lập trình.
* Cho phép sử dụng nhiều ngôn ngữ lập trình trong cùng một dự án.
* Kết thừa và sử dụng chéo giữa các ngôn ngữ lập trình dễ dàng như trên cùng một ngôn ngữ (Có thể viết một class trên C#, sau đó kế thừa trong VB.NET và ngược lại).
* Việc triển khai (Deploy) các ứng dụng dễ dàng. Chỉ cần Copy-and-run (copy là chạy). Không cần cài đặt và tránh được “địa ngục DLL” như trước đây
  + 1. **Visual Studio 2017**
       1. **Giới thiệu về Visual Studio**

Visual Studio 2017 và .NET Framework 4.6.1 đánh dấu thế hệ kế tiếp của các công cụ phát triển từ Microsoft.

Visual Studio 2017 và .NET Framework 4.6.1 tập trung vào những cột trụ cốt lỗi trong trải nghiệm phát triển phần mềm, cung cấp những nền tảng mới nhất, những trải nghiệm nhắm tới các loại ứng dụng nhất định, cùng nhiều cải thiện về kiến trúc lõi.

Visual Studio là bộ công cụ hoàn chỉnh cho phép xây dựng cả các ứng dụng cho máy để bàn lẫn các ứng dụng web doanh nghiệp theo nhóm. Ngoài khả năng xây dựng những ứng dụng desktop tốc độ cao, bạn còn có thể sử dụng các công cụ phát triển mạnh mẽ dựa trên thành phần cùng các công nghệ khác nhằm đơn giản hóa thiết kế, phát triển và triển khai các giải pháp doanh nghiệp theo nhóm.

Được thiết kế nhằm đáp ứng những yêu cầu mới nhất từ các nhà phát triển, Visual Studio cung cấp những đổi mới chủ chốt trong những lĩnh vực sau:

* Dân chủ hóa quá trình quản lý chu trình phát triển ứng dụng (ALM). ALM có vai trò quan trọng trong một tổ chức phát triển. Trước đây, không phải vai trò nào cũng có đóng góp như nhau trong quá trình này. Visual Studio Team System 2017 tiếp nối khả năng tổ chức nhằm xây dựng một nền tảng cân bằng chức năng cùng đóng góp trên toàn bộ tiến trình ALM.
* Hỗ trợ những xu hướng mới xuất hiện. Mỗi năm ngành công nghiệp phát triển phần mềm lại cho ra dời những công nghệ và xu hướng mới. Với Visual Studio 2017, Microsoft mang đến hỗ trợ các công cụ và framework cho những phát kiến mới nhất trong kiến trúc, phát triển và triển khai ứng dụng.
* Tạo cảm hứng cho nhà phát triển. Kể từ phiên bản phát hành đầu tiên của Visual Studio, Microsoft đã đặt ra tiêu chuẩn về sự hiệu quả và linh hoạt cho các nhà phát triển. Visual Studio 2017 tiếp tục truyền thống này bằng cách cải thiện đáng kể trải nghiệm cho mọi vai trò phát triển phần mềm.
* Lèo lái làn sóng nền tảng thế hệ mới. Microsoft tiếp tục đầu tư vào hệ điều hành đang dẫn đầu thị trường, các ứng dụng hiệu quả, cùng các nền tảng máy chủ nhằm không ngừng gia tăng giá trị khách hàng. Với Visual Studio 2010, khách hàng sẽ nhận được hỗ trợ công cụ yêu cầu để tạo ra các giải pháp không ngờ xung quanh những công nghệ này.

Có gì mới trong Visual Studio 2017:

* Microsoft Visual Studio 2017 đơn giản hóa quá trình phát triển giải pháp, giảm thiểu nguy cơ cũng như tăng cường kết quả trả về. Các công cụ cho mọi giai đoạn trong chu trình phát triển, từ thiết kế, phát triển đến kiểm định và triển khai, cho phép bạn thỏa sức thể hiện trí tưởng tượng và mang đến những giải pháp có ảnh hưởng lớn. Phiên bản này được phân phối dưới dạng ảnh đĩa ISO và có thể được ghi ra một đĩa trống cho quá trình cài đặt.
* Microsoft Visual Studio 2017 cung cấp một môi trường tích hợp các công cụ và kiến trúc máy chủ nhằm đơn giản hóa toàn bộ tiến trình phát triển ứng dụng. Tạo ra những kết quả kinh doanh với những tiến trình hiệu quả, tùy biến và có thể dự đoán cũng như tăng cường khả năng làm việc liên thông cùng khả năng theo dõi trong suốt chu trình phát triển với các phân tích chi tiết. Bất kể là tạo lập các giải pháp mới hay tăng cường các ứng dụng hiện có giải phóng sức sáng tạo của bạn với các công cụ dựng mẫu, kiến trúc và phát triển cho phép bạn hiện thực hóa tầm nhìn nhắm đến số lượng nền tảng và công nghệ luôn mở rộng, bao gồm điện toán đám mây và song song. Hiện thực hóa hiệu quả làm việc nhóm bằng cách khai thác các tính năng cộng tác tiên tiến cũng như sử dụng các công cụ kiểm định và dò lỗi tích hợp nhằm vừa đảm bảo chất lượng giải pháp vừa giảm thiểu phí tổn phát triển.

Hệ điều hành hỗ trợ:

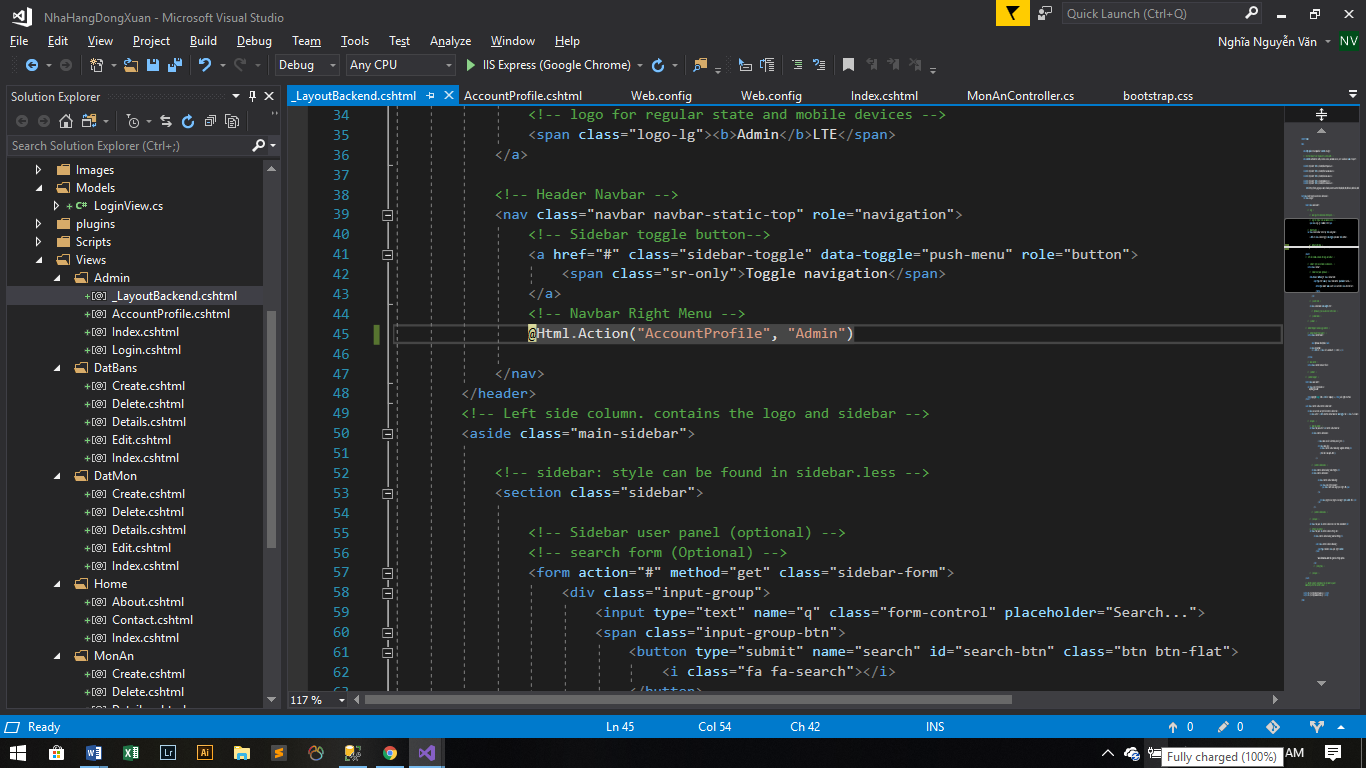
* Windows 10 (x86 và x64)
* Windows XP (x86) Service Pack 3 - mọi phiên bản trừ Starter Edition
* Windows XP (x64) Service Pack 2 - mọi phiên bản trừ Starter Edition
* Windows Vista (x86 & x64) Service Pack 1 - mọi phiên bản trừ Starter Edition
* Windows 7 (x86 và x64)
* Windows Server 2003 (x86 & x64) Service Pack 2
* Windows Server 2003 R2 (x86 & x64)
* Windows Server 2008 (x86 & x64) Service Pack 2
* Windows Server 2008 R2 (x64)

Yêu cầu phần cứng:

Computer with a 1.6GHz or faster processor 1024 MB RAM (1.5 GB if running in a virtual machine) 3 GB of available hard-disk space 5400 RPM hard drive 17 DirectX 9 capable video card running at 1024 x 768 or higher-resolution display

* + - 1. **Giới thiệu môi trường tích hợp (IDE) của Visual Studio**

Một điều thật tuyệt vời là Visual Studio sử dụng một trình IDE chung cho toàn bộ ngôn ngữ lập trình (ASP.NET, VB.NET, C#,…). Điều này đảm bảo tính nhất quán cho các ngôn ngữ trên nền .NET, giúp bạn chỉ cần “Học một lần nhưng áp dụng mọi nơi”.



*Hình 1.6. Cửa sổ giao diện chính của môi trường phát triển tích hợp.*

* + 1. **Miscrosoft SQL Server 2017**
       1. **SQL Server là gì?**

SQL Server là viết tắt của Structure Query Language, nó là một công cụ quản lý dữ liệu được sử dụng phổ biến ở nhiều lĩnh vực. Hầu hết các ngôn ngữ bậc cao đều có trình hỗ trợ SQL như Visual BASic, Oracle, Visual C...

Các chương trình ứng dụng và các công cụ quản trị CSDL cho phép người sử dụng truy nhập tới CSDL mà không cần sử dụng trực tiếp SQL. Nhưng những ứng dụng đó khi chạy phải sử dụng SQL.

* + - 1. **Lịch sử phát triển.**

SQL Server được phát triển từ ngôn ngữ SEQUEL2 bởi IBM theo mô hình Codd tại trung tâm nghiên cứu của IBM ở California, vào những năm 70 cho hệ thống QTCSDL lớn.

Đầu tiên SQL Server được sử dụng trong các ngôn ngữ quản lý CSDL và chạy trên các máy đơn lẻ. Song do sự phát triển nhanh chóng của nhu cầu xây dựng những CSDL lớn 23 theo mô hình khách chủ (trong mô hình này toàn bộ CSDL được tập trung trên máy chủ (Server)). Mọi thao tác xử lý dữ liệu được thực hiện trên máy chủ bằng các lệnh SQL máy trạm chỉ dùng để cập nhập hoặc lấy thông tin từ máy chủ). Ngày nay trong các ngôn ngữ lập trình bậc cao đều có sự trợ giúp của SQL. Nhất là trong lĩnh vực phát triển của Internet ngôn ngữ SQL càng đóng vai trò quan trọng hơn. Nó được sử dụng để nhanh chóng tạo các trang Web động.

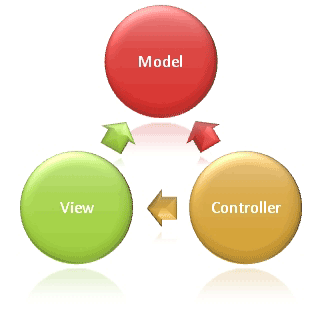
SQL đã được viện tiêu chuẩn quốc gia Mỹ (ANSI)và tổ chức tiêu chuẩn quốc tế (ISO) chấp nhận như một ngôn ngữ chuẩn cho CSDL quan hệ. Nhưng cho đến nay chuẩn này chưa đưa ra đủ 100% nên các SQL nhúng trong các ngôn ngữ lập trình khác nhau đã được bổ xung mở rộng cho SQL chuẩn cho phù hợp với các ứng dụng của mình. Do vậy có sự khác nhau rõ ràng giưã các SQL.

* + - 1. **Đặc điểm của SQL Server**
* SQL là ngôn ngữ tựa tiếng Anh.
* SQL là ngôn ngữ phi thủ tục, nó không yêu cầu ta cách thức truy nhập CSDL như thế nào. Tất cả các thông báo của SQL đều rất dễ sử dụng và ít khả năng mắc lỗi.
* SQL cung cấp tập lệnh phong phú cho các công việc hỏi đáp DL
  + Chèn, cập nhật, xoá các hàng trong một quan hệ
  + Tạo, sửa đổi, thêm và xoá các đối tượng trong của CSDL.
  + Điều khiển việc truy nhập tới cơ sở dữ liệu và các đối tượng của CSDL để đảm bảo tính bảo mật của cơ sở DL
  + Đảm bảo tính nhất quán và sự ràng buộc của CSDL.
* Yêu cầu duy nhất để sử dụng cho các hỏi đáp là phải nắm vững được các cấu trúc CSDL của mình.
  + - 1. **Đối tượng làm việc của SQL Server**

Là các bảng (tổng quát là các quan hệ) dữ liệu hai chiều. Các bảng này bao gồm một hoặc nhiều cột và hàng. Các cột gọi là các trường, các hàng gọi là các bản ghi. Cột với tên gọi và kiểu dữ liệu (kiểu dữ liệu của mỗi cột là duy nhất) xác định tạo nên cấu trúc của bảng (Ta có thể dùng lệnh Desc[ribe] TABLE-name để xem cấu trúc của bảng, phần tuỳ chọn [] có thể được bỏ trong Oracle). Khi bảng đã được tổ chức hệ thống cho một mục đích nào đó có một CSDL.

* + 1. **Giới thiệu về mô hình MVC trong phát triển Web**
       1. **Mô hình MVC là gì**

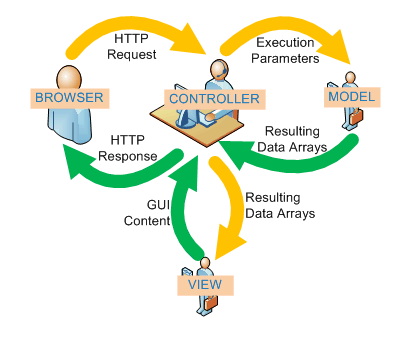
Mô hình MVC là một chuẩn mô hình và đóng vai trò quan trọng trong quá trình xây dựng – phát triển – vận hành và bảo trì một hệ thống hay một ứng dụng – phần mềm. Nó tạo ra một mô hình 3 lớp Model – View – Controller tách biệt và tương tác nhau, giúp các chuyên gia có thể dễ dàng dựa vào mô hình để trao đổi và xử lý những nghiệp vụ một cách nhanh chóng. Đây là một mô hình đã xuất hiện từ những năm 70 của thế kỷ 20 tại phòng thí nghiệm Xerox PARC ở Palo Alto, nó không phụ thuộc vào môi trường, nền tảng xây dựng hay ngôn ngữ phát triển. Chúng ta có thể áp dụng mô hình MVC vào các dự án trong môi trường Windows, Linux… và sử dụng bất kỳ ngôn ngữ nào như PHP, ASP, JSP…



*Hình 1.7. Mô hình hoạt động của MVC*

Mô hình MVC được chia làm 3 lớp xử lý gồm Model – View – Controller:

* Model: là nơi chứa những nghiệp vụ tương tác với dữ liệu hoặc hệ quản trị cơ sở dữ liệu (mysql, mssql, …) nó sẽ bao gồm các class/function xử lý nhiều nghiệp vụ như kết nối database, truy vấn dữ liệu, thêm – xóa – sửa dữ liệu…
* View: là nới chứa những giao diện như một nút bấm, khung nhập, menu, hình ảnh… nó đảm nhiệm nhiệm vụ hiển thị dữ liệu và giúp người dùng tương tác với hệ thống.
* Controller: là nới tiếp nhận những yêu cầu xử lý được gửi từ người dùng, nó sẽ gồm những class/ function xử lý nhiều nghiệp vụ logic giúp lấy đúng dữ liệu thông tin cần thiết nhờ các nghiệp vụ lớp Model cung cấp và hiển thị dữ liệu đó ra cho người dùng nhờ lớp View.
  + - 1. **Sự tương tác giữa các lớp trong mô hình MVC**
* **Controller – View**sẽ lấy những hình ảnh, nút bấm…hoặc hiển thị dữ liệu được trả ra từ Controller để người dùng có thể quan sát và thao tác. Trong sự tương tác này cũng có thể không có dữ liệu được lấy từ Model và khi đó nó chỉ chịu trách nhiệm hiển thị đơn thuần như hình ảnh, nút bấm…
* **Controller – Model** là luồng xử lý khi controller tiếp nhận yêu cầu và các tham số đầu vào từ người dùng, controller sẽ sử dụng các lớp/hàm trong Model cần thiết để lấy ra những dữ liệu chính xác.
* **View – Model** có thể tương tác với nhau mà không qua Controller, nó chỉ đảm nhận hiển thị dữ liệu chứ không phải qua bất kỳ xử lý nghiệp vụ logics nào. Nó giống như các vùng dữ liệu hiển thị tĩnh trên các website như block slidebar…
  + - 1. **Quy trình hoạt động của mô hình MVC trong dự án website**



*Hình 1.8. Quy trình hoạt động của MVC*

Mô hình trên là thể hiện mô hình MVC trong các dự án website và sẽ hoạt động theo quy trình sau:

* Người dùng sử dụng một trình duyệt web bất kỳ (Firefox, Chrome, IE, …) để có thể gửi những yêu cầu (HTTP Request) có thể kèm theo những dữ liệu nhập tới những Controller xử lý tương ứng. Việc xác định Controllerr xử lý sẽ dựa vào một bộ Routing điều hướng.
* Khi Controller nhận được yêu cầu gửi tới, nó sẽ chịu trách nhiệm kiểm tra yêu cầu đó có cần dữ liệu từ Model hay không? Nếu có, nó sẽ sử dụng các class/function cần thiết trong Model và nó sẽ trả ra kết quả (Resulting Arrays), khi đó Controller sẽ xử lý giá trị đó và trả ra View để hiển thị. Controller sẽ xác định các View tương ứng để hiển thị đúng với yêu cầu.
* Khi nhận được dữ liệu từ Controller, View sẽ chịu trách nhiệm xây dựng các thành phẩn hiển thị như hình ảnh, thông tin dữ liệu… và trả về GUI Content để Controller đưa ra kết quả lên màn hình trình duyệt.
* Trình duyệt sẽ nhận giá trị trả về (HTTP Response) và sẽ hiển thị với người dùng. Kết thúc một quy trình hoạt động.
  + - 1. **Ưu và nhược điểm của mô hình MVC**

Ưu điểm:

* Các dự án có thể áp dụng ngay mô hình MVC mà không phụ thuộc môi trường, nền tảng xây dựng hay ngôn ngữ lập trình phát triển;
* Quy hoạch các class/ function vào các thành phần riêng biệt Controller – Model – View, khi đó sẽ dễ dàng xây dựng – phát triển – quản lý – vận hành và bảo trì một dự án, tạo sự rõ ràng, trong sáng trong quá trình phát triển dự án, kiểm soát được các luồng xử lý và tạo ra các thành phần xử lý nghiệp vụ chuyên biệt hóa.
* Tạo thành mô hình chuẩn cho nhiều dự án, các chuyên gia sẽ tiếp cận – tìm hiểu những dự án đó một cách nhanh chóng và hiệu quả . Nếu bạn nắm rõ mô hình MVC của một dự án nào đó, thì khi tiếp cận với một dự án khác mà bạn chưa từng biết hoặc tiếp xúc, nhưng nó lại được xây dựng với mô hình MVC thì sẽ không khó khăn gì mà cực kỳ dễ dàng. Học một nhưng có thể hiểu và sử dụng được mười.
* Giúp các chuyên gia lập trình, nhà quản lý, nhà đầu tư, PM… có thể hiểu được dự án hoạt động ra sao hoặc giúp các lập trình viên dễ dàng quản lý – phát triển dự án. Nó không phải ngôn ngữ, nhưng khi họ cùng nhìn vào nó thì sẽ tự hiểu nó là gì, khi đó họ có thể trao đổi các yêu cầu và bàn bạc công việc.
* Đây là một mô hình chuẩn, nó tối ưu nhất hiện nay so với nhiều mô hình khác và được sử dụng trong nhiều dự án và nhiều lĩnh vực, đặc biệt trong công nghệ sản xuất ứng dụng – phần mềm. Các lập trình viên sử dụng mô hình chuẩn MVC để có thể dễ dàng phân phối và chuyển giao công nghệ.
* Đây là mô hình đơn giản, xử lý những nghiệp vụ đơn giản, và dễ dàng triển khai với các dự án nhỏ.

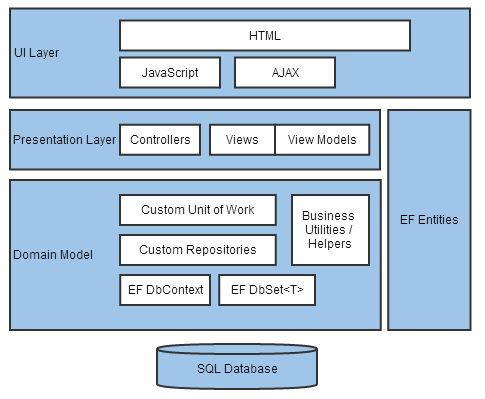
Nhược điểm:

Yêu cầu về chuyên môn khá cao, có kiến thức vững về các mô hình chuẩn. Khó triển khai với những dự án yêu cầu phúc tạp hơn. Hiện nay đang có một khái niệm mô hình mới đó là HMVC đang dần thay thế cho MVC

* + 1. **Giới thiệu về Entity Framework**
       1. **Entity Framwork là gì**

[Entity Framework](https://www.dammio.com/glossary/entity-framework) ([EF](https://www.dammio.com/glossary/ef)) là một framework ánh xạ quan hệ đối tượng (ORM) dành cho ADO.NET, là 1 phần của .NET Framework. [EF](https://www.dammio.com/glossary/ef) cho phép các nhà phát triển Web tương tác với dữ liệu quan hệ theo phương pháp hướng đối tượng đặc trưng. Lợi ích lớn nhất của [EF](https://www.dammio.com/glossary/ef) là giúp lập trình viên giảm thiểu việc lập trình mã nguồn cần thiết để truy cập và tương tác với cơ sở dữ liệu. [EF](https://www.dammio.com/glossary/ef) được Microsoft hỗ trợ phát triển lâu dài và bền vững, vì vậy [EF](https://www.dammio.com/glossary/ef) là 1 framework mạnh nhất hiện nay để phát triển ứng dụng Web với sự hỗ trợ đông đảo của các nhà phát triển Web.

* + - 1. **Vị trí của Entity Framework trong phát triển web**



*Hình 1.9. Vị trí của Entity Framework trong mô hình phát triển web*

Entity Framework có vị trí trung gian, đóng vai trò kết nối giữa cơ sở dữ liệu và các thành phần khác của 1 dự án Web khi cần đến.

**Chương 2. Phân tích thiết kế hệ thống**

* 1. **Xác định các tác nhân và ca sử dụng**
     1. **Các tác nhân**
* Admin: Người quản trị hệ thống
* User: Thành viên đã có tài khoản
* Guest: Khách hàng chưa đăng ký tài khoản, khách vãng lai
  + 1. **Ca sử dụng**
* Admin:
  + Chắc năng đăng nhập
  + Chức năng quản lý sản phẩm (thêm, sửa, xóa)
  + Chức năng quản lý bài viết (thêm, sửa, xóa)
  + Chức năng quản lý tài khoản (thêm, sửa, xóa)
  + Quản lý nhóm sản phẩm
  + Quản lý nhà sản xuất
  + Quản lý đặt lịch sửa chữa
  + Quản lý thông tin đặt hàng
  + Quản lý nhân viên
* Guest
  + Đăng ký tài khoản
  + Xem, tìm kiếm sản phẩm
  + Đặt hàng
  + Mua hàng và thanh toán
  + Đóng góp ý kiến
  + Đặt lịch sửa chữa
* User: Có đầy đủ các chức năng của Guest và thêm một số chức năng
  + Đăng nhập
    1. **Các biểu đồ ca sử dụng**
       1. **Ca sử dụng**



*Hình 2.1. Ca sử dụng*

* + - 1. **Chi tiết ca sử dụng**
         1. **Đăng nhập**



*Hình 2.2. Ca sử dụng đăng nhập*

* + - * 1. **Đăng ký**



*Hình 2.3. Ca sử dụng đăng ký*

* + - * 1. **Thêm giỏ hàng**



*Hình 2.4. Ca sử dụng thêm giỏ hàng*

* + - * 1. **Tìm kiếm sản phẩm**



*Hình 2.5. Ca sử dụng tìm kiếm sản phẩm*

* + - * 1. **Chi tiết sản phẩm**



*Hình 2.6. Ca sử dụng chi tiết sản phẩm*

* + - * 1. **Quản lý nhóm hàng**



*Hình 2.7. Ca sử dụng quản lý nhóm hàng*

* + - * 1. **Quản lý bài viết**



*Hình 2.8. Ca sử dụng quản lý bài viết*

* + - * 1. **Quản lý nhóm bài viết**



*Hình 2.9. Ca sử dụng quản lý nhóm bài viết*

* + - * 1. **Quản lý tài khoản**



*Hình 2.10. Ca sử dụng quản lý tài khoản*

* + - * 1. **Quản lý đặt hàng**



*Hình 2.11. Ca sử dụng quản lý đặt hàng*

* + - * 1. **Quản lý đặt lịch sửa chữa**



*Hình 2.12. Ca sử dụng quản lý đặt lịch sửa chữa*

* + - * 1. **Quản lý sản phẩm**



*Hình 2.13. Ca sử dụng quản lý sản phẩm*

* + - * 1. **Quản lý nhà sản xuất**



*Hình 2.14. Quản lý nhà sản xuất*

* + 1. **Mô tả chi tiết các ca sử dụng**
       1. **Tạo tài khoản**

1. Mô tả tóm tắt

* Tên ca sử dụng: Tạo tài khoản/Đăng ký
* Mục đích: Giúp khách hàng tạo tài khoản để đăng nhập vào hệ thống, để sử dụng dịch vụ của cửa hàng tốt hơn.
* Tác nhân: Khách hàng

1. Tiền điều kiện: Không
2. Các luồng sự kiện

* Luồng sự kiện chính:
  + Ca sử dụng bắt đầu khi khách hàng click vào “Đăng ký”. Màn hình view đăng ký tài khoản hiện lên.
  + Khách hàng điền thông tin vào các trường nhập liệu và click vào “Đăng ký tài khoản”.
  + Nếu khách hàng nhập đúng dữ liệu, sẽ có thông báo “Đăng ký tài khoản thành công” và màn hình tự động quay về màn hình trang chủ.
* Luồng rẽ nhánh
  + Nếu khách hàng nhập dữ liệu không đúng, sau khi click vào “Đăng ký tài khoản” sẽ có thông báo nhập sai và hiện thị đỏ những trường nhập sai dữ liệu.

1. Hậu điều kiện

* Sau khi có tài khoản, khách hàng đăng nhập để được những ưu đãi dành cho khách hàng thân thiết như: tích điểm sau mỗi lần thanh toán, nhận thông báo về các ưu đãi mới của cửa hàng.
  + - 1. **Đăng nhập**
  1. Mô tả tóm tắt
* Tên ca sử dụng: Đăng nhập.
* Mục đích: Quản trị viên, khách hàng đăng nhập vào để sử dụng hệ thống.
* Tác nhân: Người dùng có tài khoản của cửa hàng.
  1. Tiền điều kiện: Có tài khoản của cửa hàng.
  2. Các luồng sự kiện
* Luồng sự kiện chính
  + Người dùng click vào nút “Đăng nhập”, màn hình hiện lên form đăng nhập.
  + Người dùng nhập tài khoản vào form rồi click “Đăng nhập”, nếu tài khoản chính sác, màn hình sẽ quay về trang chủ nếu là tài khoản người dùng khách hàng, quay về trang quản lý nếu tài khoản người dùng là nhân viên nhà hàng.
* Luồng rẽ nhánh
  + Nếu tài khoản không chính xác, màn hình hiện lên thông báo “Tài khoản hoặc mật khẩu không đúng, nhập lại để đăng nhập”. Người dùng phải nhập lại tài khoản để đăng nhập
  1. Hậu điều kiện: Sau khi đăng nhập, người dùng có thể đăng nhập để sử dụng dịch vụ, các công việc của cửa hàng.
     + 1. **Đặt lịch sửa chữa**

1. Mô tả tóm tắt

* Tên ca sử dụng: Đặt lịch
* Mục đích: Khách hàng có thể đặt lịch sửa chữa thiết bị điện tử, điện lạnh tại nhà
* Tác nhân: Khách hàng.

1. Tiền điều kiện: Không
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Form đặt lịch hiện ra
  + Chọn các mục và nhập vào các trường nhập liệu của form
  + Click vào đặt lịch
* Luồng rẽ nhánh
  + Nếu thông tin nhập không đúng, thông báo lỗi hiện lên
  + Quay về luồng sự kiện chính

1. Hậu điều kiện: Đặt lịch sửa chữa tại nhà thành công
   * + 1. **Thêm giỏ hàng**
2. Mô tả tóm tắt

* Tên ca sử dụng: Mua hàng
* Mục đích: Khách hàng có thể mua 1 hoặc nhiều sản phẩm
* Tác nhân: Khách hàng.

1. Tiền điều kiện: Không
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Cách hàng tìm kiếm được sản phẩm cần mua
  + Chọm thêm vào giỏ hàng
  + Chọn mua hàng
  + Nhập thông tin liên hệ
  + Nếu thông tin chính xác, mua hàng thành công
* Luồng rẽ nhánh
  + Nếu thông tin liên hệ không chính xác, quay về bước nhập thông tin của luồng sự kiện chính

1. Hậu điều kiện: Mua hàng thành công
   * + 1. **Tìm kiếm sản phẩm**
2. Mô tả tóm tắt

* Tên ca sử dụng: Tìm kiếm sản phẩm
* Mục đích: Người dùng có thể tìm kiếm được sản phẩm theo thông tin sản phẩm như tên sản phẩm, hãng sản xuất, v.v...
* Tác nhân: Khách hàng.

1. Tiền điều kiện: Không
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Nhập từ khóa tìm kiếm vào ô tìm kiếm
  + Chọn tìm kiếm
  + Nếu thông tin nhập có liên quan tới sản phẩm, sản phẩm hoặc danh sách sản phẩm hiện lên
* Luồng rẽ nhánh
  + Nếu từ khóa tìm kiếm không có liên quan, thông báo không có sản phẩm, người dùng quay về luồng sự kiện chính

1. Hậu điều kiện: Tìm kiếm được sản phẩm
   * + 1. **Chi tiết sản phẩm**
2. Mô tả tóm tắt

* Tên ca sử dụng: Chi tiết sản phẩm
* Mục đích: Người dùng có thể xem được thông tin chi tiết của sản phẩm
* Tác nhân: Người dùng

1. Tiền điều kiện: Không
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Chọn sản phẩm cần xem chi tiết
  + Trang thông tin chi tiết sản phẩm hiện lên
* Luồng rẽ nhánh

1. Hậu điều kiện: Có thể xem được thông tin chi tiết của sản phẩm
   * + 1. **Quản lý nhóm hàng**
2. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý nhóm hàng
* Mục đích: Quản lý được nhóm sản phẩm
* Tác nhân: Quản trị viên

1. Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Đăng nhập vào hệ thống
  + Chọn quản lý nhóm hàng trên menu
  + Trang nhóm sản phẩm hiện ra
  + Chọn 1 nhóm hàng để xem chi tiết
  + Chọn sửa để sửa thông tin nhóm hàng
  + Chọn xóa, màn hình xác nhận xóa hiện lên, nếu người dùng muốn xóa chọn xác nhận, nếu không chọn “x” hoặc “thoát” để đóng cửa sổ
  + Chọn thêm nhóm sản phẩm
  + Màn hình thêm nhóm sản phẩm hiện lên, nhập thông tin vào
  + Chọn lưu để lưu lại nhóm sản phẩm
* Luồng rẽ nhánh
  + Nếu tên nhóm sản phẩm khi sửa hoặc thêm mới tồn tại, thôn báo nhóm đã tồn tại
  + Quay về luồng sự kiện chính

1. Hậu điều kiện: Quản lý thêm sửa xóa được nhóm sản phẩm
   * + 1. **Quản lý bài viết**
2. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý bài viết
* Mục đích: Người dùng có thể quản lý thêm, sửa, xóa bài viết
* Tác nhân: Quản trị viên

1. Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Chọn “Quản lý bài viết” trên menu
  + Danh sách bài viết hiện ra, người dùng chọn bài viết bất kỳ để xem chi tiết
  + Chọn “Xóa”, màn hình xác nhận xóa bài viết hiện lên, chọn “Xác nhận” để xóa bài viết
  + Chọn “Sửa”, màn hình thông tin chi tiết của bài viết hiện lên. Người dùng nhập thông tin cần sửa, chọn “Lưu” để lưu lại thông tin bài viết
  + Chọn “Thêm bài viết”, màn hình thêm bài viết hiện lên, người dùng nhập thông tin và chọn “Lưu” để lưu lại bài viết mới
* Luồng rẽ nhánh
  + Nếu thông tin nhập không đúng, thông báo lỗi hiện lên, người dùng quay lại bước nhập liệu của luồng sự kiện chính

1. Hậu điều kiện: Có thể thêm, sửa, xóa bài viết
   * + 1. **Quản lý nhóm bài viết**
2. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý nhóm bài viết
* Mục đích: Người dùng có thể quản lý thêm, sửa, xóa nhóm bài viết
* Tác nhân: Quản trị viên

1. Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Chọn “Quản lý nhóm bài viết”, màn hình hiện lên danh sách nhóm bài viết
  + Người dùng click vào nhóm bài viết cần xem để xem chi tiết
  + Chọn “Sửa”, màn hình hiển thị lên thông tin nhóm bài viết, người dùng nhập lại thông tin cần sửa và chọn “Cập nhật”
  + Chọn “Xóa”, màn hình xác nhận hiện lên, chọn “Xác nhận” để xóa nhóm bài viết, chọn “Thoát” hoặc “x” để thoát
  + Chọn “Thêm nhóm bài viết”, màn hình thêm nhóm bài viết hiện lên, nhập thông tin nhóm bài viết và chọn “Lưu”
* Luồng rẽ nhánh
  + Nếu thông tin nhập vào không đúng, màn hình hiển thị thông báo lỗi, quay về lại bước nhập dữ liệu của luồng sự kiện chính

1. Hậu điều kiện: Thêm, sửa, xóa được nhóm bài viết
   * + 1. **Quản lý tài khoản**
2. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý tài khoản
* Mục đích: Người dùng có thể quản lý thêm, sửa, xóa tài khoản
* Tác nhân: Quản trị viên

1. Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Chọn “Quản lý tài khoản”, màn hình hiện lên danh sách tài khoản khách hàng
  + Người dùng click vào tài khoản cần xem để xem chi tiết, màn hình thông tin chi tiết hiện lên
  + Chọn “Sửa”, màn hình hiển thị lên thông tài khoản, người dùng nhập lại thông tin cần sửa và chọn “Cập nhật”
  + Chọn “Xóa”, màn hình xác nhận hiện lên, chọn “Xác nhận” để xóa tài khoản, chọn “Thoát” hoặc “x” để thoát
  + Chọn “Thêm tài khoản”, màn hình thêm tài khoản hiện lên, nhập thông tin tài khoản và chọn “Lưu”
* Luồng rẽ nhánh
  + Nếu thông tin nhập vào không đúng, màn hình hiển thị thông báo lỗi, quay về lại bước nhập dữ liệu của luồng sự kiện chính

1. Hậu điều kiện: Thêm, sửa, xóa được tài khoản khách hàng
   * + 1. **Quản lý đặt hàng**
2. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý đặt hàng
* Mục đích: Người dùng có thể quản lý sửa, xóa thông tin đặt hàng của khách hàng
* Tác nhân: Quản trị viên

1. Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Chọn “Quản lý đơn hàng”, màn hình hiện lên danh sách đơn hàng của khách hàng
  + Người dùng click vào đơn hàng cần xem để xem chi tiết
  + Chọn “Sửa”, màn hình hiển thị lên thông tin đơn hàng, người dùng nhập lại thông tin cần sửa và chọn “Cập nhật”
  + Chọn “Xóa”, màn hình xác nhận hiện lên, chọn “Xác nhận” để xóa đơn hàng, chọn “Thoát” hoặc “x” để thoát
* Luồng rẽ nhánh
  + Nếu thông tin nhập vào không đúng, màn hình hiển thị thông báo lỗi, quay về lại bước nhập dữ liệu của luồng sự kiện chính

1. Hậu điều kiện: Sửa, xóa được nhóm bài viết
   * + 1. **Quản lý đặt lịch sửa chữa**
2. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý đặt lịch sửa chữa tại nhà
* Mục đích: Người dùng có thể quản lý sửa, xóa thông tin đặt lịch sửa chữa của khách hàng
* Tác nhân: Quản trị viên

1. Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Chọn “Quản lý đặt lịch”, màn hình hiện lên danh sách đặt lịch sửa chữa
  + Người dùng click vào đặt lịch cần xem để xem chi tiết
  + Chọn “Sửa”, màn hình hiển thị lên thông tin đặt lịch, người dùng nhập lại thông tin cần sửa và chọn “Cập nhật”
  + Chọn “Xóa”, màn hình xác nhận hiện lên, chọn “Xác nhận” để xóa đơn hàng, chọn “Thoát” hoặc “x” để thoát
* Luồng rẽ nhánh
  + Nếu thông tin nhập vào không đúng, màn hình hiển thị thông báo lỗi, quay về lại bước nhập dữ liệu của luồng sự kiện chính

1. Hậu điều kiện: Sửa, xóa được nhóm bài viết
   * + 1. **Quản lý nhóm sản phẩm**

* Mô tả tóm tắt
* Tên ca sử dụng: Quản lý nhóm sản phẩm
* Mục đích: Người dùng có thể quản lý thêm, sửa, xóa thông tin đặt hàng của khách hàng
* Tác nhân: Quản trị viên
* Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
* Các luồng sự kiện
* Luồng sự kiện chính
  + Chọn “Quản lý nhóm sản phẩm”, màn hình hiện lên danh sách nhóm sản phẩm
  + Người dùng click vào nhóm sản phẩm cần xem để xem chi tiết
  + Chọn “Sửa”, màn hình hiển thị lên thông tin nhóm sản phẩm, người dùng nhập lại thông tin cần sửa và chọn “Cập nhật”
  + Chọn “Xóa”, màn hình xác nhận hiện lên, chọn “Xác nhận” để xóa đơn hàng, chọn “Thoát” hoặc “x” để thoát
  + Chọn “Thêm”, màn hình thêm nhóm sản phẩm hiện lên, người dùng nhập vào thông tin nhóm sản phẩm và chọn “Lưu”
* Luồng rẽ nhánh
  + Nếu thông tin nhập vào không đúng, màn hình hiển thị thông báo lỗi, quay về lại bước nhập dữ liệu của luồng sự kiện chính
* Hậu điều kiện: Thêm, sửa, xóa được nhóm sản phẩm
  + - 1. **Quản lý sản phẩm**

1. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý sản phẩm
* Mục đích: Người dùng có thể thêm, sửa, xóa được sản phẩm
* Tác nhân: Quản trị viên

1. Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Chọn “Quản lý sản phẩm” trên menu, màn hình hiển thị danh sách sản phẩm
  + Click vào sản phẩm cần xem để xem chi tiết, màn hình chi tiết sản phẩm hiển thị lên
  + Chọn “Sửa”, màn hình hiển thị lên thông tin sán phẩm, người dùng nhập vào thông tin cần sửa và chọn “Cập nhật”
  + Chọn “Xóa”, màn hình xác nhận hiện lên, chọn “Xác nhận” để xóa sản phẩm, chọn “x” hoặc “Thoát” để kết thúc
  + Chọn “Thêm sản phẩm”, màn hình hiển thị lên trang nhập liệu, người dùng nhập thông tin sản phẩm và chọn “Lưu”
* Luồng rẽ nhánh
  + Nếu thông tin nhập không đúng, thông báo lỗi hiện lên, người dùng quay lại bước nhập liệu tại luồng sự kiện chính

1. Hậu điều kiện: Thêm, sửa, xóa được sản phẩm
   * + 1. **Quản lý nhà sản xuất**
2. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý nhà sản xuất
* Mục đích: Người dùng có thể thêm, sửa, xóa được nhà sản xuất
* Tác nhân: Quản trị viên

1. Tiền điều kiện: Đăng nhập vào hệ thống bằng tài khoản Admin
2. Các luồng sự kiện

* Luồng sự kiện chính
  + Chọn “Quản lý nhà sản xuất” trên menu, màn hình hiển thị danh sách nhà sản xuất
  + Click vào nhà sản xuất cần xem để xem chi tiết, màn hình chi tiết nhà sản xuất hiện lên
  + Chọn “Sửa”, màn hình hiển thị lên thông tin nhà sản xuất, người dùng nhập vào thông tin cần sửa và chọn “Cập nhật”
  + Chọn “Xóa”, màn hình xác nhận hiện lên, chọn “Xác nhận” để xóa sản phẩm, chọn “x” hoặc “Thoát” để kết thúc
  + Chọn “Thêm sản phẩm”, màn hình hiển thị lên trang nhập liệu nhà sản xuất, người dùng nhập thông tin sản phẩm và chọn “Lưu”
* Luồng rẽ nhánh
  + Nếu thông tin nhập không đúng, thông báo lỗi hiện lên, người dùng quay lại bước nhập liệu tại luồng sự kiện chính

1. Hậu điều kiện: Thêm, sửa, xóa được nhà sản xuất
   1. **Biểu đồ lớp chi tiết**
      1. **Đăng nhập**

* Biểu đồ VOPC



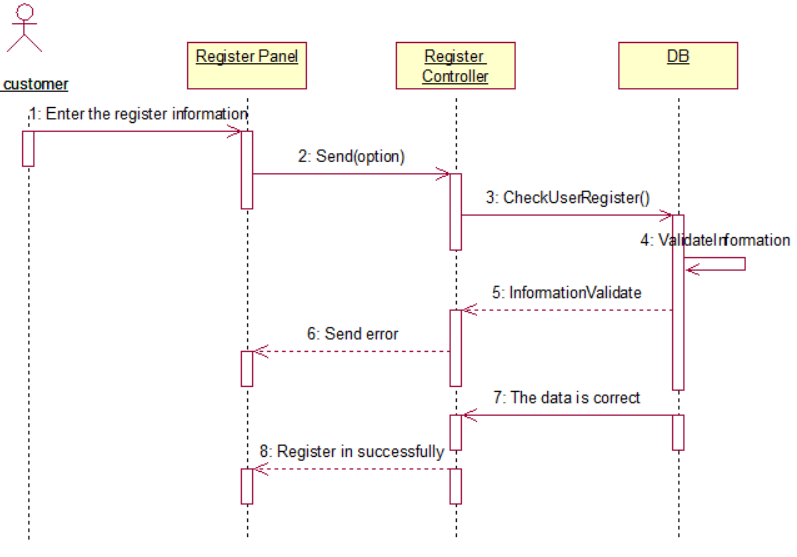
*Hình 2.15. Biểu đồ VOPC đăng nhập*

* Biểu đồ Basic Flow



*Hình 2.16. Biểu đồ Basic Flow đăng nhập*

* + 1. **Đăng ký**



*Hình 2.17. Biểu đồ Basic Flow đăng ký*

* + 1. **Mua hàng**



*Hình 2.18. Biểu đồ basic flow mua hàng*

* + 1. **Quản lý tài khoản**



*Hình 2.19. Quản lý tài khoản*

* + 1. **Quản lý sản phẩm**
* Thêm sản phẩm



*Hình 2.20. Sequence Diagram thêm sản phẩm*

* Sửa sản phẩm



*Hình 2.21. Sequence Diagram sửa sản phẩm*

* Xóa sản phẩm



*Hình 2.22. Sequence Diagram xóa sản phẩm*

* + 1. **Quản lý nhóm sản phẩm**
* Thêm nhóm sản phẩm



*Hình 2.23. Sequence Diagram thêm nhóm sản phẩm*

* Sửa nhóm sản phẩm



*Hình 2.24. Sequence Diagram sửa nhóm sản phẩm*

* Xóa nhóm sản phẩm



*Hình 2.25. Sequence Diagram thêm nhóm sản phẩm*

* + 1. **Quản lý nhóm bài viết**
* Thêm nhóm bài viết



*Hình 2.26. Sequence Diagram thêm nhóm bài viết*

* Sửa nhóm bài viết



*Hình 2.27. Sequence Diagram sửa nhóm bài viết*

* Xóa nhóm bài viết



*Hình 2.28. Sequence Diagram xóa nhóm bài viết*

* + 1. **Quản lý bài viết**
* Thêm bài viết



*Hình 2.29. Sequence Diagram thêm bài viết*

* Sửa bài viết



*Hình 2.30. Sequence Diagram sửa bài viết*

* Xóa



*Hình 2.31. Sequence Diagram xóa bài viết*

* + 1. **Quản lý đặt hàng**
* Sửa đặt hàng



*Hình 2.32. Sequence Diagram sửa đặt hàng*

* Xóa



*Hình 2.33. Sequence Diagram xóa đặt hàng*

* + 1. **Quản lý đặt lịch**
* Sửa đặt lịch



*Hình 2.34. Sequence Diagram sửa đặt hàng*

* Xóa đặt lịch



*Hình 2.35. Sequence Diagram xóa đặt hàng*

* + 1. **Quản lý nhà sản xuất**
* Thêm nhà sản xuất



*Hình 2.36. Sequence Diagram thêm nhà sản xuất*

* Sửa nhà sản xuất



*Hình 2.37. Sequence Diagram sửa nhà sản xuất*

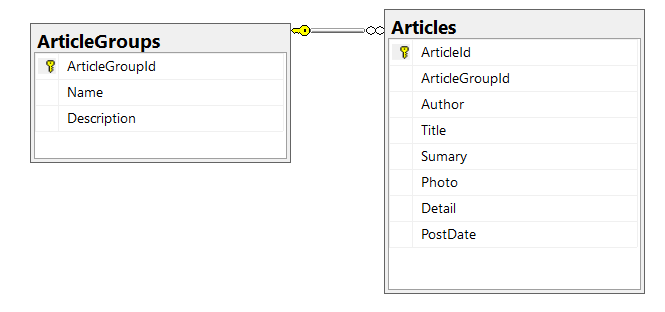
* Xóa nhà sản xuất



*Hình 2.38. Sequence Diagram xóa nhà sản xuất*

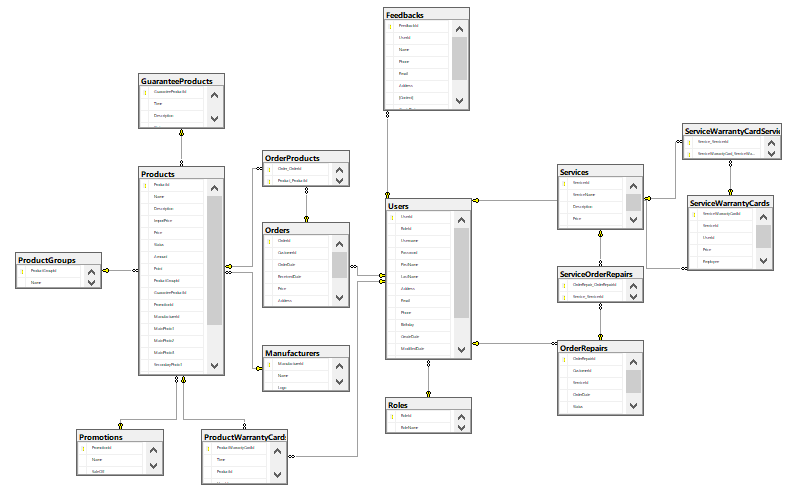
* 1. **Cơ sở dữ liệu**
     1. **Cơ sở dữ liệu quan hệ**

Nhóm dữ liệu tin tức



*Hình 2.39. Quan hệ dữ liệu nhóm bảng tin tức*

Nhóm dữ liệu sản phẩm



*Hình 2.40. Quan hệ dữ liệu chính*

* + 1. **Chi tiết bảng dữ liệu**
       1. **Bảng Wards**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| WardId | INT | 11 | PRIMARY KEY |  |
| DistrictId | INT | 11 | FOREIGN KEY |  |
| WardName | NVARCHAR | 100 | NOT NULL |  |
| WardType | NVARCHAR | 100 | NUT NULL |  |

* + - 1. **Bảng Provinces**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ProvinceId | INT | 11 | PRIMARY KEY |  |
| ProvinceName | NVARCHAR | 100 | NOT NULL |  |
| ProvinceType | NVARCHAR | 100 | NOT NULL |  |

* + - 1. **Bảng Districts**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| DistrictId | INT | 11 | PRIMARY KEY |  |
| ProvinceId | INT | 11 | FOREIGN KEY |  |
| DistrictName | NVARCHAR | 100 | NOT NULL |  |
| DistrictType | NVARCHAR | 100 | NOT NULL |  |

* + - 1. **Bảng ArticleGroups**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ArticleGroupId | INT | 11 | PRIMARY KEY |  |
| Name | NVARCHAR | 255 | NOT NULL |  |
| Description | NVARCHAR | 255 | NOT NULL |  |

* + - 1. **Bảng Article**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ArticleId | INT | 11 | PRIMARY KEY | Not null |
| ArticleGroupId | INT | 11 | FOREIGN KEY | Not null |
| Author | NVARCHAR | 255 | NOT NULL |  |
| Title | NVARCHAR | 255 | NOT NULL |  |
| Sumary | TEXT |  | NOT NULL |  |
| Photo | NVARCHAR |  | NOT NULL |  |
| Detail | TEXT |  | NOT NULL |  |
| PostDate | DATETIME |  | NOT NULL |  |

* + - 1. **Bảng Feedback**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| FeedbackId | INT | 11 | PRIMARY KEY |  |
| User | NVARCHAR |  | NOT NULL |  |
| Name | NVARCHAR |  | NOT NULL |  |
| Phone | NVARCHAR |  | NOT NULL |  |
| Email | NVARCHAR |  | NOT NULL |  |
| Address | NVARCHAR |  | NOT NULL |  |
| Content | NVARCHAR |  | NOT NULL |  |
| CreateDate | DATETIME |  | NOT NULL |  |
| Status | BINARY |  | NOT NULL |  |

* + - 1. **Bảng GuaranteeTimeProduct**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| GuaranteeProductId | INT | 11 | PRIMARY KEY |  |
| Time | INT | 5 | NOT NULL |  |
| Description | NVARCHAR |  |  |  |
| Status | BINARY |  | NOT NULL |  |

* + - 1. **Bảng Producer**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ManufacturerId | INT | 11 | PRIMATY KEY |  |
| Name | NVARCHAR | 255 |  |  |
| Logo | NVARCHAR | 255 |  |  |

* + - 1. **Bảng Order**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| OrderId | INT | 11 | PRIMARY KEY |  |
| CustomerId | INT | 11 | FOREIGN KEY |  |
| OrderDate | DATETIME |  |  |  |
| ReceivedDate | DATETIME |  |  |  |
| Price | FLOAT | 15 |  |  |
| Address | NVARCHAR | 255 |  |  |
| Status | BINARY |  |  |  |

* + - 1. **Bảng OrderRepair**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| OrderRepairId | INT | 11 | PRIMARY KEY |  |
| CustomerId | INT | 11 | FOREIGN KEY |  |
| ServiceId | INT | 11 | FOREIGN KEY |  |
| OrderDate | DATETIME |  |  |  |
| Status | BINARY |  |  |  |
| Content | NVARCHAR |  |  |  |

* + - 1. **Bảng Product**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ProductId | INT | 11 | PRIMARY KEY |  |
| Name | NVARCHAR |  |  |  |
| Description | NVARCHAR |  |  |  |
| ImportPrice | FLOAT |  |  |  |
| Price | FLOAT |  |  |  |
| Status | BINARY |  |  |  |
| MainPhoto1 | NVARCHAR |  |  |  |
| MainPhoto2 | NVARCHAR |  |  |  |
| MainPhoto3 | NVARCHAR |  |  |  |
| SecondaryPhoto1 | NVARCHAR |  |  |  |
| SecondaryPhoto2 | NVARCHAR |  |  |  |
| SecondaryPhoto3 | NVARCHAR |  |  |  |
| CreatedDate | DATETIME |  |  |  |
| Color | NVARCHAR |  |  |  |
| Amount | FLOAT |  |  |  |
| Point | FLOAT |  |  |  |
| ProductGroupId | INT | 11 | FOREIGN KEY |  |
| GuaranteeProductId | INT | 11 | FOREIGN KEY |  |
| PromotionId | INT | 11 | FOREIGN KEY |  |
| ManufacturerId | INT | 11 | FOREIGN KEY |  |

* + - 1. **Bảng ProductGroup**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ProductGroupId | INT | 11 | PRIMARY KEY |  |
| Name | NVARCHAR |  |  |  |

* + - 1. **Bảng ProductWarrantyCard**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ProductWarrantyCardId | INT | 11 | PRIMARY KEY |  |
| Time | INT | 11 |  |  |
| ProductId | INT | 11 | FOREIGN KEY |  |
| UserId | INT | 11 | FOREIGN KEY |  |

* + - 1. **Bảng Promotion**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| PromotionId | INT | 11 | PRIMARY KEY |  |
| Name | NVARCHAR |  |  |  |
| SaleOff | INT | 2 |  |  |

* + - 1. **Bảng Role**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| RoleId | INT | 11 | PRIMARY KEY |  |
| RoleName | NVARCHAR |  |  |  |

* + - 1. **Bảng Service**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ServicerId | INT | 11 | PRIMARY KEY |  |
| ServiceName | NVARCHAR |  |  |  |
| Description | NVARCHAR |  |  |  |
| Price | FLOAT |  |  |  |
| WarrantyPeriod | INT |  |  |  |

* + - 1. **Bảng ServiceWarrantyCard**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| ServiceWarrantyCardId | INT | 11 | PRIMARY KEY |  |
| ServiceId | INT | 11 | FOREIGN KEY |  |
| UserId | INT | 11 | FOREIGN KEY |  |
| Price | FLOAT | 12 |  |  |
| Employee | NVARCHAR |  |  |  |
| RepairDate | DATETIME |  |  |  |

* + - 1. **Bảng User**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường dữ liệu | Kiểu dữ liệu | Độ dài | Ràng buộc | Mô tả |
| UserId | INT | 11 | PRIMARY KEY |  |
| RoleId | INT | 11 | FOREIGN KEY |  |
| Username | NVARCHAR |  |  |  |
| Password | NVARCHAR |  |  |  |
| FirstName | NVARCHAR |  |  |  |
| LastName | NVARCHAR |  |  |  |
| Address | NVARCHAR |  |  |  |
| Email | NVARCHAR |  |  |  |
| Phone | NVARCHAR |  |  |  |
| Birthday | DATETIME |  |  |  |
| CreateDate | DATETIME |  |  |  |
| ModifiedDate | DATETIME |  |  |  |
| Avatar | NVARCHAR |  |  |  |
| Status | BINARY |  |  |  |
| IsActive | BINARY |  |  |  |

**Chương 3. Cài đặt và triển khai website quản lý nhà hàng**

* 1. **Giao diện của khách hàng**
  2. **Giao diện quản trị**
  3. **Một số giao diện khác**