

BỘ CÔNG THƯƠNG
ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN TỐT NGHIỆP

Đề tài: Xây dựng website thương mại

Giáo viên hướng dẫn: Ths Trần Thanh Huân

Sinh viên thực hiện: Nguyễn Năng Dũng

Lớp: Khoa học máy tính 1 – K7

Hà Nội 5 – 2016

[illegible]

(Ký tên, ghi rõ họ tên)

Lời nói đầu

*Hiện nay trên thế giới thương mại điện tử đang phát triển rất mạnh mẽ. Kỹ thuật số giúp chúng ta tiết kiệm đáng kể các chi phí nhờ chi phí vận chuyển trung gian, chi phí giao dịch và đặc biệt là giúp tiết kiệm thời gian để con người đầu tư vào các hoạt động khác. Hơn nữa thương mại điện tử còn giúp con người có thể tìm kiếm tự động theo nhiều mục đích khác nhau, tự động cung cấp thông tin theo nhu cầu và sở thích của con người. Giờ đây, con người có thể ngồi tại nhà để mua sắm mọi thứ theo ý muốn và các website bán hàng trên mạng sẽ giúp ta làm được điều đó. Chính vì vậy trong đồ án này em chọn đề tài về: **“Xây dựng Website bán hàng thông minh”**.*

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

Trước hết em xin chân thành cảm ơn Thầy Trần Thanh huân , Giảng viên Khoa Công Nghệ Thông Tin, Trường Đại học Công nghiệp Hà Nội, người đã trực tiếp hướng dẫn, nhận xét, giúp đỡ em trong suốt quá trình thực hiện đồ án.

Xin chân thành cảm ơn ban giám hiệu nhà trường, các thầy cô trong Khoa Công Nghệ Thông Tin và các phòng ban nhà trường đã tạo điều kiện tốt nhất cho em cũng như các bạn khác trong suốt thời gian học tập và làm tốt nghiệp.

Cuối cùng em xin gửi lời cảm ơn đến gia đình, bạn bè, người thân đã giúp đỡ động viên em rất nhiều trong quá trình học tập và làm Đồ án thực tập Tốt Nghiệp. Do thời gian thực hiện có hạn kiến thức còn nhiều hạn chế nên Đồ án thực hiện chắc chắn không tránh khỏi những thiếu sót nhất định.

Em rất mong nhận được ý kiến đóng góp của thầy cô giáo và các bạn để em có thêm kinh nghiệm và tiếp tục hoàn thiện đồ án của mình.

Em xin chân thành cảm ơn!

Hà Nội, ngày 28 tháng 4 năm 2016

Sinh viên

Nguyễn Năng Dũng

Mục lục

Chương 1: Tổng quan.....	9
1.1 Giới thiệu về đề tài.....	9
1.2 Lý do chọn đề tài.....	9
1.3 Mục đích phát triển đề tài.....	9
1.4 Ý nghĩa thực tiễn.....	9
1.5 Đặt bài toán.....	9
1.5.1 Chức năng.....	10
1.5.2 Yêu cầu đặt ra.....	10
1.6 Các giai đoạn triển khai dự án.....	11
1.7 Vấn đề - Giải pháp.....	11
Chương 2: Tìm hiểu 1 số công nghệ xây dựng website.....	12
1.8 ASP.NET và MVC.....	12
1.8.1 ASP.NET.....	12
1.8.2 MVC.....	13
1.9 Javascript.....	16
1.10 Ajax.....	18
1.11 Bootstrap.....	19
1.12 Hệ cơ sở dữ liệu SQL server.....	19
1.13 Truy vấn Linq.....	19
1.13.1 Các thành phần của LINQ.....	20
1.13.2 LINQ vs ADO.NET.....	23
Chương 3: Phân tích thiết kế hệ thống.....	24
1.14 Tổng quan thiết kế.....	24
1.15 Một số yêu cầu.....	25
1.15.1 Actors and Use Cases Descriptions.....	25
1.16 Phân tích UC.....	29
1.16.1 UC01: Login.....	29
1.16.2 UC02: Manage product.....	30
1.16.3 UC03: Add new product.....	30

1.16.4 UC04: Edit product.....	32
1.16.5 UC05: Delete product.....	33
1.16.6 UC06: View product.....	34
1.16.7 UC7: Manage statistic.....	34
1.16.8 UC8: Statistic sales.....	35
1.16.9 UC9: Statistic selling product.....	36
1.16.10 UC10: Manage Manufacture.....	36
1.16.11 UC11: View Manufacture.....	37
1.16.12 UC12: Create Manufacture.....	38
1.16.13 UC13: Edit Manufacture.....	39
1.16.14 UC14: Delete Manufacture.....	40
1.16.15 UC15: View Order.....	41
1.16.16 UC16: Change Order.....	42
1.16.17 UC17: View feedback.....	43
1.16.18 UC18: View Cart.....	44
1.16.19 UC19: Add product in cart.....	44
1.16.20 UC23: Update account profile.....	45
1.16.21 UC22: Register account.....	46
1.16.22 UC20: Remove product in cart.....	47
1.16.23 UC21: Send feedback.....	48
Chương 4:Thiết kế cơ sở dữ liệu.....	49
1.17 Thiết kế các bảng chính.....	49
1.17.1 Bảng Product (Sản phẩm).....	49
1.17.2 Bảng Loại sản phẩm.....	51
1.17.3 Bảng Đơn đặt hàng.....	51
1.17.4 Bảng chi tiết đơn đặt hàng.....	52
1.17.5 Bảng Theo dõi thay đổi trạng thái đơn hàng.....	52
1.17.6 Bảng Thông tin tài khoản hệ thống.....	53
1.18 Thiết kế bảng phụ.....	53
1.18.1 Bảng nhà sản xuất.....	53
1.18.2 Bảng menu.....	54

1.18.3 Bảng Slide.....	54
1.18.4 Bảng phản hồi khách hàng.....	55
1.19 Thiết kế quan hệ, ràng buộc các bảng trong CSDL.....	55
Chương 5: Giao diện hệ thống.....	58
1.20 SC01: Login.....	58
1.21 SC2: Trang chủ.....	60
1.22 SC3: Chi tiết sản phẩm.....	61
1.23 SC4:Giỏ hàng.....	61
1.24 SC5:Đặt hàng.....	62
1.25 SC6: Tìm kiếm.....	64
1.26 SC7:Danh sách sản phẩm (Quản trị).....	65
1.27 SC8:Update produce.....	66
1.28 SC9:Thêm mới 1 sản phẩm.....	67
1.29 SC10: Báo cáo thống kê.....	68
Chương 6: Phát triển một số tính năng nâng cao cho website.....	69
1.30 Bài toán đặt ra.....	69
1.31 Phát triển hệ thống tìm kiếm gần đúng.....	70
1.32 Phát triển hệ thống chăm sóc khách hàng.....	73
1.33 Phát triển hệ thống gợi ý sản phẩm.....	73
1.34 Cổng thanh toán online trong website bán hàng.....	74
1.34.1 Cổng thanh toán Onepay.....	74
1.34.2 Cổng thanh toán Paypal.....	75
1.34.3 Cổng thanh toán Ngân Lượng.....	76
1.34.4 Ví điện tử Bảo Kim.....	77
1.34.5 Cổng thanh toán Payoo.....	77
1.34.6 Cổng thanh toán Vn Mart.....	78
Kết luận.....	79
Tài liệu tham khảo.....	80

Danh sách Hình vẽ

Hình 2-1: Các thành phần chính của mô hình MVC.....	14
Hình 2-2: Kiến trúc của linq trong .NET Framework 3.5.....	20
Hình 2-3: Kiến trúc Dataset.....	21
Hình 2-4: Kiến trúc của Entity Framework.....	22
Hình 3-5: Admin roles.....	25
Hình 3-6: UseCase Customer.....	25
Hình 4-7: Bảng Product.....	50
Hình 4-8: Bảng loại sản phẩm.....	51
Hình 4-9: Bảng đơn đặt hàng.....	51
Hình 4-10: Bảng chi tiết đơn đặt hàng.....	52
Hình 4-11: Theo dõi trạng thái đơn hàng.....	52
Hình 4-12: Bảng tài khoản.....	53
Hình 4-13: Bảng nhà sản xuất.....	54
Hình 4-14: Bảng menu.....	54
Hình 4-15: Bảng Slide.....	54
Hình 4-16: Bảng phản hồi khách hàng.....	55
Hình 4-17: Diagram CSDL.....	57
Hình 5-18: Trang login quản trị.....	58
Hình 5-19: Login khách hàng.....	59
Hình 5-20: Trang chủ.....	60
Hình 5-21: Trang chi tiết sản phẩm.....	61
Hình 5-22: Giỏ hàng.....	62
Hình 5-23: Giao diện đặt hàng.....	63
Hình 5-24: Tìm kiếm sản phẩm.....	64
Hình 5-25: Danh sách sản phẩm(Quản trị).....	65
Hình 5-26: Sửa thông tin sản phẩm.....	66
Hình 5-27: Thêm mới sản phẩm.....	67
Hình 5-28: Biểu đồ báo cáo.....	68

Danh sách các từ viết tắt

#	Chữ viết tắt	Gải thích
1	SRS	Software Requirement Specification
2	BR	Business Rule
3	SC	Screen
4	UC	Use Case
5	CRUD	Create/Read/Update/Delete a record in database
6	CSDL	Cơ sở dữ liệu
7	OOP	Object Oriented Programming
8	JS	JavaScript
9	HTML	
10	PHP	
11	SEO	

Chương 1: Tổng quan

1.1 Giới thiệu về đề tài.

Trong bối cảnh kinh tế thị trường hiện nay, việc mỗi công ty, cửa hàng hay thậm chí cá nhân có một hay nhiều website là một việc bình thường, xong việc khai thác dữ liệu từ website biến những tác động lên website trở thành những dữ liệu có ích lại là 1 vấn đề khá còn mới mẻ. Vấn đề còn bỏ ngỏ, nhận thấy cơ hội, khả năng phát triển, em nhận đề tài “Xây dựng website bán hàng thông minh” nhằm mục đích đưa các giải pháp, triển khai thực tiễn 1 số thuật toán nhằm phân tích dữ liệu cho người dùng.

1.2 Lý do chọn đề tài

Khi internet ra đời, không ai nghĩ rằng nó sẽ phát triển đến ngày hôm nay, nó len lỏi đến mọi ngõ ngách trong cuộc sống, trở thành 1 phần không thể thiếu. Chỉ cần ngồi nhà và cả thế giới trong tầm tay bạn. Cũng như vậy, khi internet đã trở nên phổ biến, việc mua bán qua mạng internet phát triển cũng phát triển theo đó. Xong bên cạnh đó, việc nắm bắt tìm hiểu, phân tích nhu cầu tiêu dùng là một phần vô cùng quan trọng, nó giúp người tiêu dùng dễ dàng chọn lựa sản phẩm phù hợp, giúp các nhà kinh doanh quyết định đón đầu xu thế của khách hàng.

Bên cạnh đó, việc phát triển website đi kèm những tính năng nâng cao giúp tối ưu hóa việc trải nghiệm website của người dùng cũng là cách giúp website phát triển 1 cách bền vững và có hiệu quả.

1.3 Mục đích phát triển đề tài

Đề tài được phát triển nhằm mục đích thương mại điện tử, quản lý việc mua bán thông qua mạng internet, nâng cao trải nghiệm sử dụng website của người dùng qua đó đưa thương hiệu đến rộng rãi với người dùng hơn.

1.4 Ý nghĩa thực tiễn

Phát triển các sản phẩm công nghệ phù hợp với nhu cầu của xã hội, qua đó rèn luyện, trau dồi, phát triển kiến thức của bản thân nhằm đáp ứng nhu cầu của các nhà tuyển dụng sau khi ra trường, tạo ra sản phẩm thực tế có giá trị sử dụng trong cuộc sống.

1.5 Đặt bài toán

Bài toán đặt ra: Xây dựng website bán hàng gồm bao gồm các chức năng của 1 website bán hàng đơn thuần.

Qua quá trình khảo sát các nguồn thông tin từ thầy giáo hướng dẫn, bạn bè, những người có kinh nghiệm trong phát triển website, trải nghiệm thực tế trên một số website bán hàng trực tuyến có thương hiệu của nước ngoài như <http://www.ebay.com>, <http://www.alibaba.com>, thương hiệu trong nước như <https://www.thegioididong.com>, <http://www.trananh.vn/>, em phân tích bài toán:

1.5.1 Chức năng

Đây là website bán hàng và giới thiệu sản phẩm của cửa hàng, công ty đến người tiêu dùng với các thông tin chi tiết mặt hàng, hình ảnh mô tả sản phẩm, giá cả, khuyến mãi, ưu đãi với người mua hàng. Website cần bao gồm các chức năng sau:

- Cho phép cập nhập hàng hóa vào CSDL
- Hiện thị danh sách các mặt hàng hóa
- Hiện thị các mặt hàng người dùng đã chọn sau quá trình tìm hiểu sản phẩm
- Hiện thị thông tin khách hàng
- Quản lý đơn đặt hàng
- Cập nhập hàng hóa, nhà sản xuất, ...
- Xử lý đơn hàng
- Thống kê các khách hàng mua hàng trong khoảng thời gian nhất định
- Ngoài ra còn một số chức năng nâng cao theo yêu cầu của đơn vị đầu tư website

Ngoài ra, còn có thêm chức năng phát triển 1 số tính năng nâng cao như gợi ý mua hàng, tìm kiếm gần đúng, ...

1.5.2 Yêu cầu đặt ra

Hệ thống có 2 phần:

a. Phần thứ nhất: Khách hàng

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

Giao diện người dùng thân thiện, có thể thay đổi phù hợp với những ngày lễ, những đợt giảm giá khuyến mãi, ... nhằm đánh mạnh vào nhu cầu, tâm lý mua sắm của người dùng. Việc người dùng sử dụng website của bạn thêm một giây là gia tăng thêm tỷ lệ mua hàng của khách hàng.

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

Sau khi khách hàng đã lựa chọn xong và muốn đặt hàng, cần có giao diện đặt hàng cho người dùng bao gồm các thông tin về hàng hóa đã chọn, số lượng, số tiền,

tổng tiền đơn hàng, các thông tin cá nhân của khách hàng để phục vụ việc liên lạc lại với khách hàng.

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

b. Phần thứ hai: Giao diện quản trị.

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

- Chức năng thêm, sửa, xóa các loại mặt hàng, nhà sản xuất, ...
- Tiếp nhận đơn hàng của khách hàng và sửa lý đơn hàng
- Thông kê doanh thu
- Một số chức năng nâng cao theo yêu cầu như tạo giao diện menu động, tùy chỉnh theo ý người quản trị website, ...

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

1.6 Các giai đoạn triển khai dự án.

- Giai đoạn 1:

Xây dựng 1 website đầy đủ các chức năng cơ bản, sử dụng cơ sở dữ liệu SQL và xây dựng website theo mô hình 3 lớp kết hợp MVC 5 (Giới thiệu tại phần tiếp theo)

- Giai đoạn 2:

Xây dựng 1 số tính năng mới để tăng trải nghiệm của người dùng với website, qua đó hỗ trợ người dùng 1 cách tối ưu.

Trong phần báo cáo thực tập tốt nghiệp, em phát triển giai đoạn 1. Tiến độ án tốt nghiệp sẽ phát triển giai đoạn 2.

1.7 Vấn đề - Giải pháp

- Khó khăn

- Kiến thức thực tế còn hạn chế
- Công nghệ biến đổi theo từng đơn vị thời gian nên cần có chiến lược phát triển, thời gian và công sức tìm hiểu.
- Triển khai bài toán theo hướng hoạt động cá nhân nên chưa có sự giao lưu ý tưởng.

- Giải pháp:

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

Chương 2: Tìm hiểu 1 số công nghệ xây dựng website

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

Chương này bao gồm lý thuyết về các công nghệ làm website được áp dụng.

1.8 ASP.NET và MVC

1.8.1 ASP.NET

ASP.NET là Active Server Pages .NET (.NET là .NET framework). Nói đơn giản và ngắn gọn thì ASP.NET là một công nghệ có tính cách mạng dùng để phát triển các ứng dụng về mạng hiện nay cũng như trong tương lai (ASP.NET is a revolutionary technology for developing web applications). ASP.NET là một phương pháp tổ chức hay khung tổ chức (framework) để thiết lập các ứng dụng hết sức hùng mạnh cho mạng dựa trên CLR (Common Language Runtime) chứ không phải là một ngôn ngữ lập trình.

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

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

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

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

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

ASP.NET hỗ trợ gần 25 ngôn ngữ lập trình mới với .NET và chạy trong môi trường biên dịch (compiled environment), còn ASP chỉ chấp nhận VBScript và JavaScript nên ASP chỉ là một scripted language trong môi trường thông dịch (in the interpreter environment). Không những vậy, ASP.NET còn kết hợp nhuần

nhuần với XML (Extensible Markup Language) để vận chuyển các thông tin qua mạng.

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

1.8.2 MVC

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

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

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

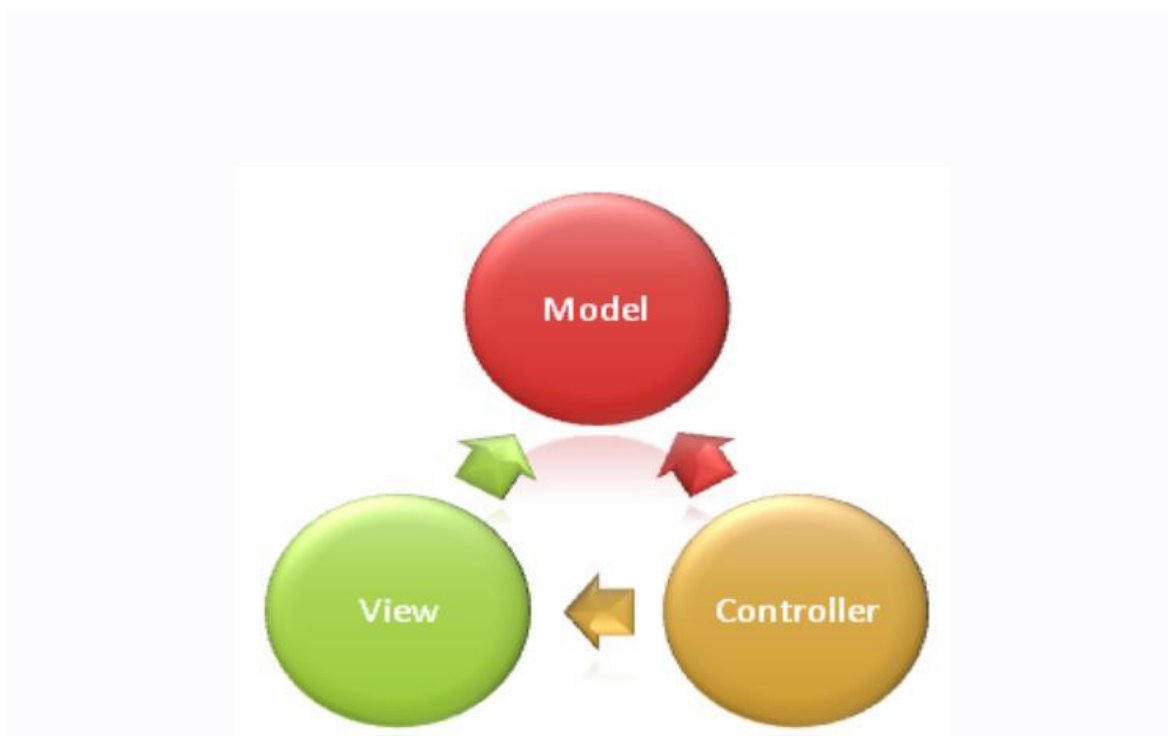
Xuất xứ

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

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

Kiến trúc mô hình MVC

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



Hình 2-1: Các thành phần chính của mô hình MVC

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

Đặc điểm của mô hình MVC

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

kể có thể làm việc đồng thời với nhau. MVC cho phép thay đổi trong 1 phần của ứng dụng mà không ảnh hưởng đến các phần khác

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

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

Tiềm hiểu về Controller

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

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

Tìm hiểu Model

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

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

Khi sử dụng models, ta cũng cần tuân theo nguyên tắc chính của chúng là không

xuất giá trị trực tiếp trong model. Mà tất cả những dữ liệu ấy, phải đưa vào mảng và trả về theo phương thức. Và tiếp tục ở view ta sẽ sử dụng nó để lấy dữ liệu ra.

Tìm hiểu về View

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

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

➤ Ưu Điểm:

- MVC làm cho ứng dụng trở nên trong sáng, giúp lập trình viên phân tách ứng dụng thành ba lớp một cách rõ ràng. Điều này sẽ rất giúp ích cho việc phát triển những ứng dụng xét về mặt lâu dài cho việc bảo trì và nâng cấp hệ thống.
- MVC hiện đang là mô hình lập trình tiên tiến bậc nhất hiện nay, điều mà các framework vẫn đang nỗ lực để hướng tới sự đơn giản và yếu tố lâu dài cho người sử dụng.

➤ Khuyết Điểm:

- Mặc dù, MVC tỏ ra lợi thế hơn nhiều so với cách lập trình thông thường. Nhưng MVC luôn phải nạp, load những thư viện đồ sộ để xử lý dữ liệu. Chính điều này làm cho mô hình trở nên chậm chạp hơn nhiều so với việc code tay thuần túy.
- MVC đòi hỏi người tiếp cận phải biết qua OOP, có kinh nghiệm tương đối cho việc thiết lập và xây dựng một ứng dụng hoàn chỉnh. Sẽ rất khó khăn nếu OOP của người sử dụng còn yếu.
- MVC tận dụng mảng là thành phần chính cho việc truy xuất dữ liệu. Nhất là với việc sử dụng active record để viết ứng dụng. Chúng luôn cần người viết phải nắm vững mô hình mảng đa chiều.

1.9 Javascript

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

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

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

Javascript là gì?

Về phương diện thực thi ngôn ngữ

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

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

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

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

Về phương diện phong cách ngôn ngữ lập trình

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

Nói tóm lại:

Javascript là một ngôn ngữ lập trình:

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

Javascript được sử dụng để làm gì?

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

***Ưu điểm của Javascript:**

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

****Nhược điểm của Javascript***

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

Nguyên tắc lập trình trong JavaScript

- Bất cứ ngôn ngữ lập trình nào cũng đều phải có nguyên tắc của nó, và Javascript cũng ko ngoại lệ. Để bắt đầu học lập trình 1 ngôn ngữ nào, trước tiên chúng ta cần phải biết quy tắc của nó.
- Một số quy tắc quan trọng trong JS mà chúng ta cần phải nhớ như sau:
 - Lệnh Javascript phân biệt chữ in hoa và chữ thường
 - Mọi câu lệnh Javascript đều kết thúc bằng dấu chấm phẩy “;”
 - Các điều kiện phải được khai báo trong cặp dấu ngoặc đơn ().
 - Khi sử dụng lệnh điều khiển, nếu sử dụng nhiều hơn 1 lệnh, bạn phải sử dụng cặp dấu ngoặc nhọn {}
 - Javascript sử dụng dấu chấm “.” để tham chiếu đến 1 phương thức hay thuộc tính của đối tượng

1.10 Ajax

Ajax là một kỹ thuật viết tắt của chữ *AJAX = Asynchronous JavaScript and XML*, đây là một công nghệ giúp chúng ta tạo ra những [Web động](#) mà hoàn toàn không reload lại trang nên rất mượt và đẹp. Đối với công nghệ web hiện nay thì ajax không thể thiếu, nó là một phần làm nên sự sinh động cho website.

Ajax được viết bằng ngôn ngữ Javascript nên nó chạy trên client, tức là mỗi máy (user) sẽ chạy độc lập hoàn toàn không ảnh hưởng lẫn nhau. Hiện nay có nhiều thư viện javascript như **jQuery** hay **Angular** đều hỗ trợ kỹ thuật này nhằm giúp chúng ta thao tác dễ dàng hơn.

1.11 Bootstrap

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

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

Các lợi ích khi sử dụng bootstrap

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

1.12 Hệ cơ sở dữ liệu SQL server

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

1.13 Truy vấn Linq

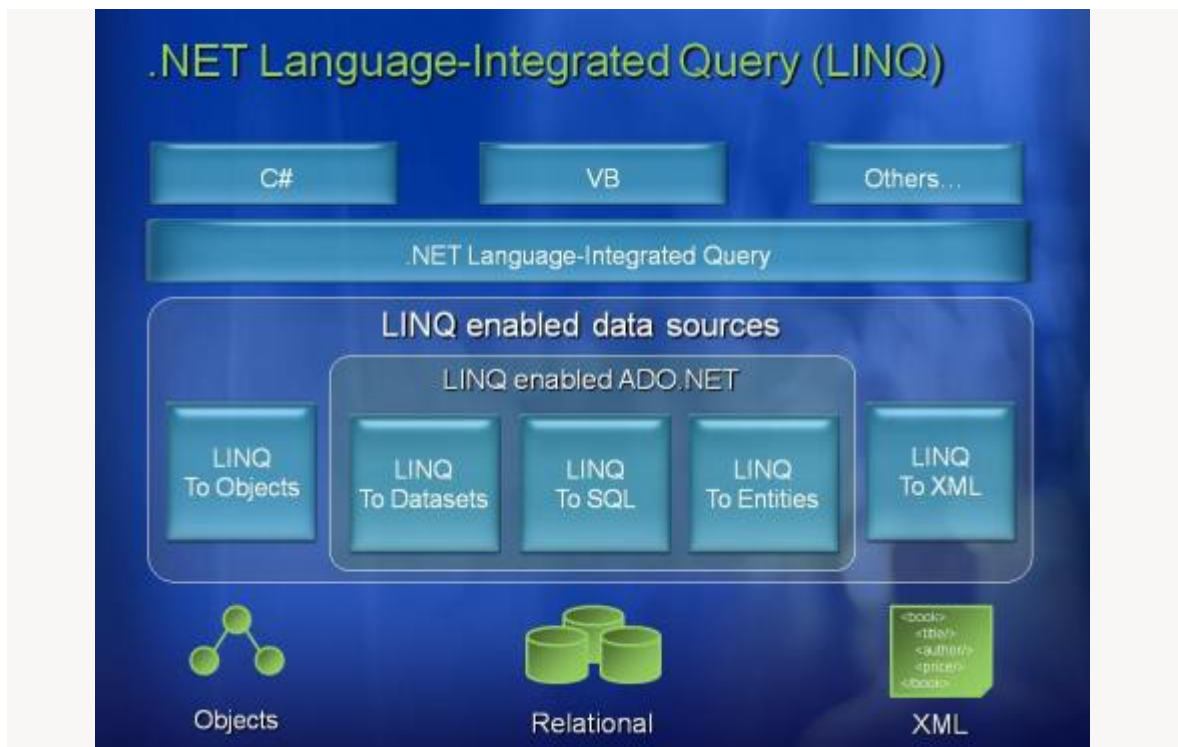
LINQ là gì ?

Để giảm gánh nặng thao tác trên nhiều ngôn ngữ khác nhau và cải thiện năng suất lập trình, Microsoft đã phát triển giải pháp tích hợp dữ liệu cho .NET

Framework có tên gọi là LINQ (Language Integrated Query), đây là thư viện mở rộng cho các ngôn ngữ lập trình C# và Visual Basic.NET (có thể mở rộng cho các ngôn ngữ khác) cung cấp khả năng truy vấn trực tiếp dữ liệu Object, cơ sở dữ liệu và XML.

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

1.13.1 Các thành phần của LINQ



Hình 2-2:Kiến trúc của linq trong .NET Framework 3.5

1.13.1.1 LINQ to Objects

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

1.13.1.2 LINQ to SQL

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

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

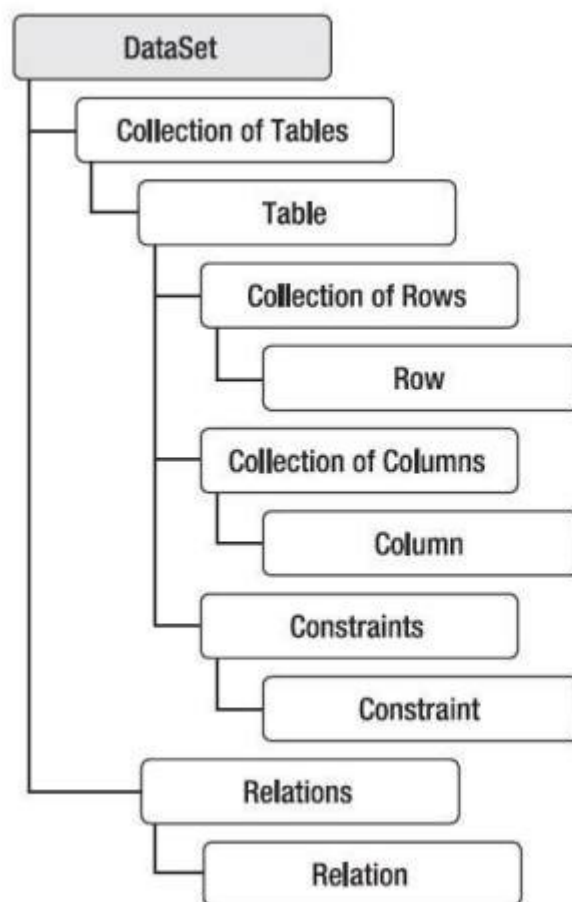
1.13.1.3 LINQ to XML

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

1.13.1.4 LINQ to Datasets

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

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



Hình 2-3:Kiến trúc Dataset

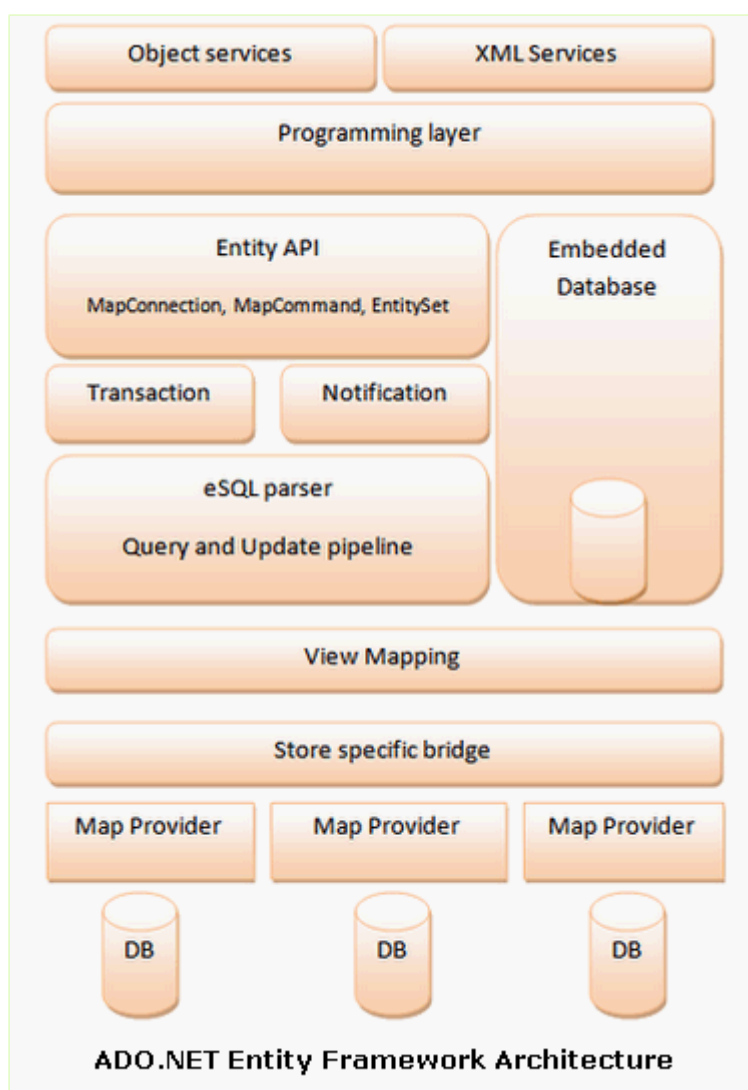
LINQ to DataSet cho phép người lập trình sử dụng DataSets như một nguồn dữ liệu bình thường bằng các cú pháp truy vấn căn bản của LINQ

1.13.1.5 LINQ to Entities

Khái Niệm về Entity Framework

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

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



Hình 2-4:Kiến trúc của Entity Framework

Kiến trúc của Entity Framework

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

Các Operator chuẩn trong LINQ

1.Select / SelectMany	11.GroupBy
2.Where	12.Distinct
3.Sum / Min / Max / Average / Aggregate	13.Union / Intersect / Except
4.Join / GroupJoin	14.EqualAll
5.Take / TakeWhile	15.First / FirstOrDefault / Last / LastOrDefault
6.Skip / SkipWhile	16.Single
7.OfType	17.ElementAt
8.Concat	18.Any / All / Contains
9.OrderBy / ThenBy	19.Count
10.Reverse	

1.13.2 LINQ vs ADO.NET

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

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

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

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

Chương 3: Phân tích thiết kế hệ thống

Chương này sẽ đi sâu vào phân tích thiết kế hệ thống, bao gồm các UseCase, phân tích các UseCase.

Phần khảo sát đã được mô tả trong Chương 1 phần 5: Đặt bài toán

1.14 Tổng quan thiết kế

Chủ cửa hàng, công ty là người có quyền cao nhất: có thể xem thông tin về tình trạng, hoạt động cũng như mọi thay đổi trên website.

Nếu chủ cửa hàng cũng là người quản trị website thì mọi thông tin trên website web là do chủ cửa hàng quản lý, nếu thuê nhân viên quản trị thì chủ cửa hàng cung cấp tài khoản và quyền hạn cho nhân viên đó truy cập vào hệ thống.

Nhân viên là người giả quyết các nhiệm vụ được giao như các nhận đơn hàng, tiếp thu ý kiến phản hồi của người sử dụng....

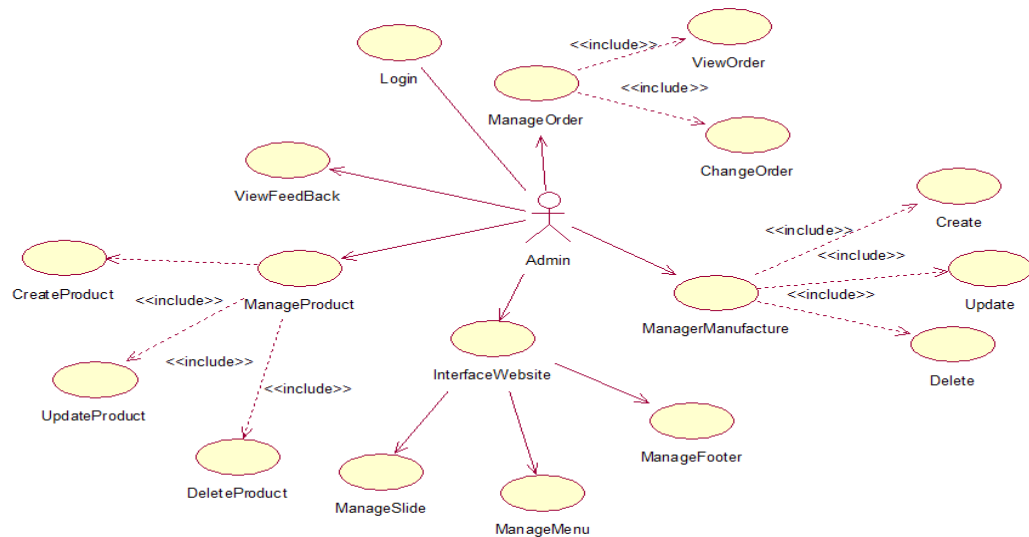
Quyền cao nhất là của chủ cửa hàng, có thể xem đầy đủ mọi thông tin trên website, còn nhân viên và khách hàng chỉ được xem những thông tin được cho phép

Một số trang bán hàng cao cấp có tích hợp nhiều phương thức thanh toán, xong do điều kiện thực tế, phần này khó áp dụng với website có quy mô nhỏ.

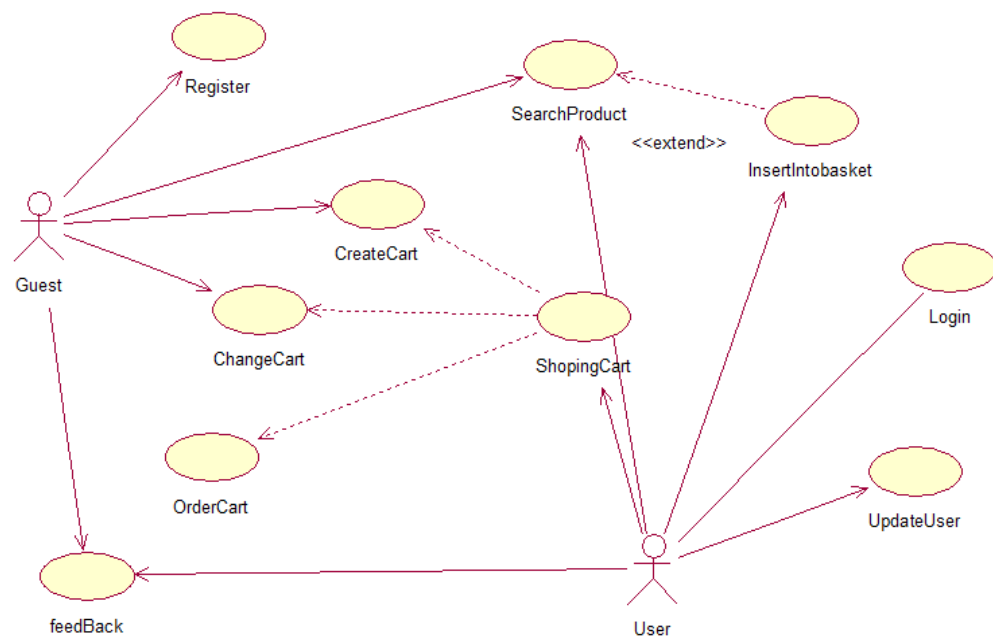
1.15 Một số yêu cầu

1.15.1 Actors and Use Cases Descriptions

1.15.1.1 Diagram



Hình 3-5: Admin roles



Hình 3-6: UseCase Customer

1.15.1.2 Actors Description

#	Actor Name	Definition & Interests
1	Admin	
2	Customer	

1.15.1.3 Use Cases Description

#	Code	Name	Brief Description
1	UC01	Login	Cho phép actor đăng nhập vào hệ thống
2	UC02	Manage product	Cho phép actor quản lý thông tin sản phẩm
3	UC03	Add new product	Cho phép actor thêm sản phẩm mới
4	UC04	Edit product	Cho phép actor sửa thông tin sản phẩm đã
5	UC05	Delete product	Cho phép actor xóa thông tin sản phẩm
6	UC06	View product	Cho phép actor xem thông tin sản phẩm
7	UC07	Manage statistic	Cho phép actor quản lý thông kê
8	UC08	Statistic sales	Cho phép actor thông kê doanh thu
9	UC09	Statistic selling product	Cho phép actor thống kê các sản phẩm bán chạy
10	UC10	Manage Manufacture	Cho phép actor quản lý thông tin cá nhân của mình
11	UC11	View Manufacture	Cho phép actor xem thông tin Manufacture
12	UC12	Add Manufacture	Cho phép actor thêm mới 1 Manufacture
13	UC13	Edit Manufacture	Cho phép actor sửa thông tin Manufacture
14	UC14	Delete Manufacture	Cho phép actor xóa thông tin Manufacture
15	UC15	View order	Cho phép actor quản lý thông tin đặt hàng của khách
16	UC16	ChaneOrder	Cho phép actor chỉnh sửa trạng thái của order
17	UC17	View FeedBack	Cho phép actor xem các feedback của khách hàng

18	UC18	View cart	Cho phép actor xem giỏ hàng
19	UC19	Add product in cart	Cho phép actor thêm sản phẩm vào giỏ hàng
20	UC20	Remove product in cart	Cho phép actor xóa sản phẩm khỏi giỏ hàng
21	UC21	Send complain	Cho phép actor gửi khiếu nại về trang web
22	UC22	Register account	Cho phép actor đăng kí tài khoản của website
23	UC23	Update account profile	Cho phép actor sửa thông tin tài khoản của website

Table 3: Use Case List

1.15.1.4 Use Case & Actor mapping

Actor	Admin	Customer
Use Case		
UC1: Login	x	x
UC2: Manage product	x	
UC3: Add new product	x	
UC4: Edit product	x	
UC5: Delete product	x	
UC6: View product	x	
UC7: Manage statistic	x	
UC8: Statistic sales	x	
UC9: Statistic selling product	x	
UC10: Manage Manufacture	x	
Actor	Admin	Customer
Use Case		

UC11: View Manufacture	x	
UC12: Add Manufacture	x	
UC13: Edit Manufacture	x	
UC14: Delete Manufacture	x	
UC15: View order	x	
UC16: ChaneOrder	x	
UC17: View FeedBack	x	
UC18: View cart		x
UC19: Add product in cart		x
UC20: Remove product in cart		x
UC21: Send complain		x
UC22: Register account		x
UC23: Update account profile		x

(*): Anonymous user chỉ có thể thêm giỏ hàng chỉnh sửa số lượng sản phẩm trong giỏ hàng.

1.16 Phân tích UC

1.16.1 UC01: Login

Use Case Description

Use case name: Login	ID: 1	Importance level: Trung bình
Primary actor: Actor	Use case type: Chi tiết	
Stakeholders and interests: Người dùng đăng nhập vào hệ thống.		
Brief descprition: Khi bắt đầu phiên làm việc, người dùng cần đăng nhập vào hệ thống để thực hiện các thao tác nghiệp vụ.		
Trigger: Có yêu cầu đăng nhập vào hệ thống.		
Type: external		
Relationships: + Association: Actor + Include: Đăng xuất		
Normal flow of event: <div><div>1. Chọn chức năng đăng nhập</div><div><div>• Sub1: Nhập tên đăng nhập</div><div>• Sub2: Nhập mật khẩu</div></div><div>2. Chọn nút đăng nhập</div><div>3. Hệ thống tiến hành đăng nhập.</div><div>4. Hệ thống hiển thị các chức năng mà người dùng được phép thao tác.</div><div>5. Kết thúc</div></div>		
Subflows: <div><div>• Sub1: Nhập tên đăng nhập</div><div><div>1. Nhập tên người dùng</div><div>2. Kiểm tra tên người dùng</div><div>3. Hiện thông báo nếu có lỗi và quay lại Sub1 – 1, ngược lại đến Sub1 - 4.</div><div>4. Tiếp tục thực hiện bước 3.</div></div><div>• Sub2: Nhập mật khẩu</div><div><div>1. Nhập mật khẩu người dùng</div><div>2. Kiểm tra mật khẩu</div><div>3. Hiện thông báo nếu mật khẩu không đúng và quay lại Sub2 – 1, ngược lại đến Sub2 - 4.</div></div></div>		

4. Tiếp tục thực hiện bước 3.
Exceptional flows: Nếu nhập sai tên đăng nhập và mật khẩu thì thông báo cho người dùng.

1.16.2 UC02: Manage product

Use Case Description

Name	Manage product	Code	UC02
Description	Cho phép actor xem toàn bộ sản phẩm		
Actor	Admin	Trigger	Actor click button [Quản trị sản phẩm]
Pre-condition	Actor đã đăng nhập vào hệ thống quản trị		
Post condition	Chọn chức năng trong quản lý sản phẩm		

Activities

Actor		System	
Main Flow: View manage product			
1	Từ màn hình bất kỳ, actor lựa chọn left menu Quản trị sản phẩm		
		2	Load trang Quản lý thông tin sản phẩm

Business Rules

Rule No.	Rule	Description
1	Hiển thị Gridview	GridView hiển thị đúng dữ liệu, dữ liệu đảm bảo đã encode trước khi hiển thị, đảm bảo không vỡ trang khi dữ liệu quá dài.
2	Ẩn Gridview	Ẩn Gridview khi không có dữ liệu.

System Message

N/A

1.16.3 UC03: Add new product

Use Case Description

Name	Add new product	Code	UC03
Description	Cho phép actor thêm sản phẩm mới		
Actor	Admin	Trigger	Actor click button [Create New]
Pre-condition	Actor đã đăng nhập vào hệ thống quản trị		
Post condition	Thêm sản phẩm thành công		

Activities

Actor		System	
Main Flow: Add new product successfully			
1	Từ màn hình bất kỳ, actor lựa chọn left menu Quản lý sản phẩm		
		2	Load trang Quản lý thông tin sản phẩm
3	Click button [Thêm mới]		
		4	Load form thêm mới thông tin tin sản phẩm
5	Nhập thông tin sản phẩm và click button [Thêm mới]		
		6	Kiểm tra dữ liệu có thỏa mãn yêu cầu.
7	Actor nhập lại trường thông tin báo lỗi(nếu có)		
		8	Lưu xuống CSDL

Business Rules

Rule No.	Rule	Description
1	Dữ liệu về sản phẩm	Các trường dữ liệu phải thỏa mãn đúng yêu

	thêm phải thỏa mãn yêu cầu.	câu về loại trường hoặc các điều kiện bên ngoài thêm.
--	-----------------------------	---

System Message

MS01	Hệ thống thông báo nếu dữ liệu nhập vào không thỏa mãn tại các trường thuộc tính
MS02	Hệ thống thông báo “Thêm thành công” khi bản ghi được lưu thành công

1.16.4 UC04: Edit product

Use Case Description

Name	Edit product	Code	UC04
Description	Cho phép actor sửa thông tin sản phẩm		
Actor	Admin	Trigger	Actor click button [Sửa]
Pre-condition	Actor đã đăng nhập vào hệ thống.		
Post condition	Sửa sản phẩm thành công		

Activities

Actor		System	
Main Flow: Add new product successfully			
1	Từ màn hình bất kỳ, actor lựa chọn left menu Quản lý sản phẩm		
		2	Load trang Quản lý thông tin sản phẩm
3	Click button [Edit] trên bản ghi cần sửa		
		4	Load form thông tin sản phẩm
5	Sửa thông tin sản phẩm và click button [Save]		
		6	Kiểm tra dữ liệu có thỏa mãn yêu cầu. Thông báo nếu có trường không thỏa mãn
7	Actor nhập lại thông tin chưa đúng(nếu có)		

		8	Cập nhập thông tin mới xuống CSDL
--	--	---	-----------------------------------

Business Rules

Rule No.	Rule	Description
1	Dữ liệu về sản phẩm phải thỏa mãn yêu cầu.	Các trường dữ liệu phải thỏa mãn đúng yêu cầu về loại trường hoặc các điều kiện bên ngoài.

System Message

MS01	Hệ thống thông báo nếu dữ liệu nhập vào không thỏa mãn tại các trường thuộc tính
MS02	Hệ thống thông báo “Sửa thành công” nếu dữ liệu sửa đúng quy cách

1.16.5 UC05: Delete product

Use Case Description

Name	Delete product	Code	UC05
Description	Cho phép actor xóa sản phẩm		
Actor	Admin	Trigger	Actor click button [Xóa sản phẩm]
Pre-condition	Actor đã đăng nhập vào hệ thống		
Post condition	Xóa sản phẩm thành công		

Activities

Actor		System	
Main Flow: Delete product successfully			
1	Từ màn hình view Thông tin sản phẩm click button [Delete]		
		2	Xóa thông tin sản phẩm và báo thành công

Business Rules

Rule No.	Rule	Description

System Message

MS01	Thông báo “Xoá thành công” sau khi xoá bản ghi
------	--

1.16.6 UC06: View product

Use Case Description

Name	View product	Code	UC06
Description	Cho xem thông tin sản phẩm		
Actor	Admin ,Customer,Guest	Trigger	Trên trang chủ của website
Pre-condition			
Post condition			

Activities

Actor		System	
Main Flow: View productsuccessfully			
1	Vào trang chủ hệ thống(trang Home)		
		2	Hệ thống hiển thị thông tin theo quy cách nhất định để gây ấn tượng cho người dùng, giúp người dùng dễ dàng chọn sản phẩm

Business Rules

Rule No.	Rule	Description

System Message

N/A

1.16.7 UC7: Manage statistic

Use Case Description

Name	Manage static	Code	UC11
Description	Cho phép actor quản lý thông kê theo mục chọn		
Actor	Admin	Trigger	Actor view màn hình quản lý thống kê

Pre-condition	Actor đã đăng nhập vào hệ thống
Post condition	Hiển thị chức năng quản lý thống kê trên màn hình

Activities

Actor		System	
Main Flow: View productsuccessfully			
1	Từ màn hình bất kỳ, actor lựa chọn left menu Statistic		
		2	Hiển thị 1 số biểu đồ thể hiện 1 số thống kê chính cho người dùng

Business Rules

Rule No.	Rule	Description

System Message

N/A

1.16.8 UC8: Statistic sales

Use Case Description

Name	Statistic sales	Code	UC12
Description	Cho phép actor thống kê doanh thu bán sản phẩm		
Actor	Admin	Trigger	Actor click button [Thống kê doanh thu]
Pre-condition	Actor đã đăng nhập vào hệ thống		
Post condition	Hiển thị thống kê thành công.		

Activities

Actor		System	
Main Flow: View statistic successfully			
1	Actor chọn button [Thống kê		

	doanh thu] trên menu trái của trang quản trị.		
		2	Hệ thống hiển thị thống kê doanh thu.

Business Rules

Rule No.	Rule	Description

System Message

N/A

1.16.9 UC9: Statistic selling product

Use Case Description

Name	Statistic selling product	Code	UC12
Description	Cho phép actor thống kê sản phẩm bán chạy		
Actor	Admin	Trigger	Actor click button [Thống kê sản phẩm bán chạy]
Pre-condition	Actor đã đăng nhập vào hệ thống		
Post condition	Hiển thị thông tin thống kê thành công.		

Activities

Actor		System	
Main Flow: View statistic successfully			

Business Rules

Rule No.	Rule	Description

System Message

N/A

1.16.10 UC10: Manage Manufacture

Use Case Description

Name	Manage Manufacture	Code	UC10
Description	Cho phép actor quản lý thông tin nhà sản xuất		

Actor	Admin	Trigger	Actor click button [Quản trị nhà sản xuất]
Pre-condition	Actor đã đăng nhập vào hệ thống quản trị		
Post condition	Hiển thị thông tin các nhà sản xuất		

Activities

Actor		System	
Main Flow: View statistic successfully			
1	Trên trang bất kì, trên menu trái actor chọn [Quản trị nhà sản xuất]		
		2	Hệ thống hiển thị danh sách các bản ghi về nhà sản xuất

Business Rules

Rule No.	Rule	Description
1	Hiển thị Gridview	GridView hiển thị đúng dữ liệu, dữ liệu đảm bảo đã encode trước khi hiển thị, đảm bảo không vỡ trang khi dữ liệu quá dài.
2	Ẩn Gridview	Ẩn Gridview khi không có dữ liệu.

System Message

--	--

1.16.11 UC11: View Manufacture

Use Case Description

Name	Statistic selling product	Code	UC12
Description	Cho phép actor hiển thị chi tiết nhà sản xuất		
Actor	Admin	Trigger	Actor click button [Detail]
Pre-condition	Actor đã đăng nhập vào hệ thống		
Post	Hiển thị thông tin các nhà sản xuất		

condition	
------------------	--

Activities

Actor		System	
Main Flow: View statistic successfully			
1	Trong màn hình hiển thị danh sách tất cả các nhà sản xuất,actor bấm vào [Details] trên dòng nhà sản xuất muốn xem		
		2	Hệ thống hiển thị các thông tin chi tiết đầy đủ của nhà sản xuất người dùng chọn lên màn hình

Tải bản FULL (80 trang): <https://bit.ly/3eMJiBK>
Dự phòng: <fb.com/TaiHo123doc.net>

Business Rules

Rule No.	Rule	Description

System Message

1.16.12 UC12: Create Manufacture

Use Case Description

Name	Statistic selling product	Code	UC12
Description	Cho phép actor tạo mới 1 nhà sản xuất		
Actor	Admin	Trigger	Actor click button [Create New]
Pre-condition	Actor đã đăng nhập vào hệ thống		
Post condition			

Activities

Actor	System
Main Flow: View successfully	

1	Trên màn hình hiển thị danh sách nhà sản xuất, actor bấm chọn [Create New]		
		2	Hiển thị form cho phép người dùng nhập mới thông tin 1 nhà sản xuất
3	Người dùng nhập thông tin theo form		
		4	Hệ thống kiểm tra dữ liệu
4	Người dùng sửa thông tin những trường dữ liệu nhập chưa đúng(nếu có)		
		5	Lưu vào hệ thống csdl

Tải bản FULL (80 trang): <https://bit.ly/3eMJiBK>
Dự phòng: fb.com/TaiHo123doc.net

Business Rules

Rule No.	Rule	Description
1	Các trường thông tin yêu cầu cần phải nhập	Các trường thông tin được yêu cầu bắt buộc nhập actor cần phải nhập, nếu không hệ thống sẽ báo lỗi
2	Các trường thông tin đặc thì cần đúng yêu cầu đặc thù của trường dữ liệu	Các trường dữ liệu như kiểu số thì không được nhập chữ cái ...

System Message

Error	Có trường thông tin nhập không đúng
Success	Tạo mới thành công

1.16.13 UC13: Edit Manufacture

Use Case Description

Name	Edit manufacture	Code	UC13
Description	Cho phép actor sửa thông tin nhà sản xuất		
Actor	Admin	Trigger	Actor click button [Edit]

Pre-condition	Actor đã đăng nhập vào hệ thống
Post condition	Hiển thị thông tin thống kê thành công.

Activities

Actor		System	
Main Flow: View statistic successfully			
1	Trên màn hình danh sách các nhà sản xuất, actor chọn [Edit] trên trường dữ liệu cần thay đổi		
		2	Hiển thị form thông tin nhà sản xuất cần sửa với các thông tin cũ của nhà sản xuất
3	Actor thay đổi các thông tin cần update		
		3	Hệ thống kiểm tra dữ liệu
4	Actor sửa thông tin những trường dữ liệu nhập chưa đúng(nếu có)		
		5	Lưu vào hệ thống csdl

Business Rules

Rule No.	Rule	Description
1	Các trường thông tin yêu cầu cần phải nhập	Các trường thông tin được yêu cầu bắt buộc nhập actor cần phải nhập, nếu không hệ thống sẽ báo lỗi
2	Các trường thông tin đặc thì cần đúng yêu cầu đặc thù của trường dữ liệu	Các trường dữ liệu như kiểu số thì không được nhập chữ cái ...

System Message

1.16.14 UC14: Delete Manufacture

Use Case Description