# BÁO CÁO TIẾN ĐỘ THỰC HIỆN ĐỒ ÁN

**Báo cáo từ ngày: 27/02 đến ngày 23/04**

Họ tên SV: Nguyễn Ngọc Hồng Hân

Lớp: 61.CNTT-2

Tên đề tài: Xây dựng website cửa hàng mỹ phẩm trực tuyến Minh Thư, sử dụng ASP.NET MVC và web API

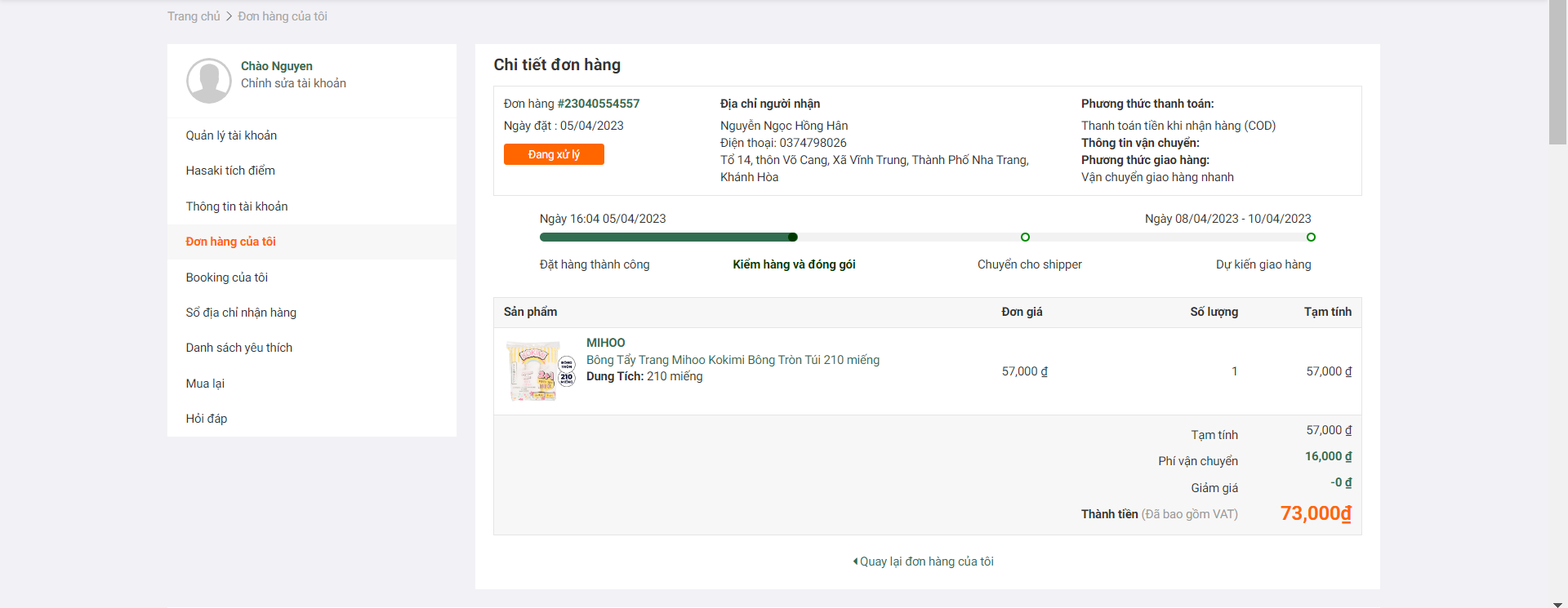
1. **Nội dung công việc đã và đang thực hiện**

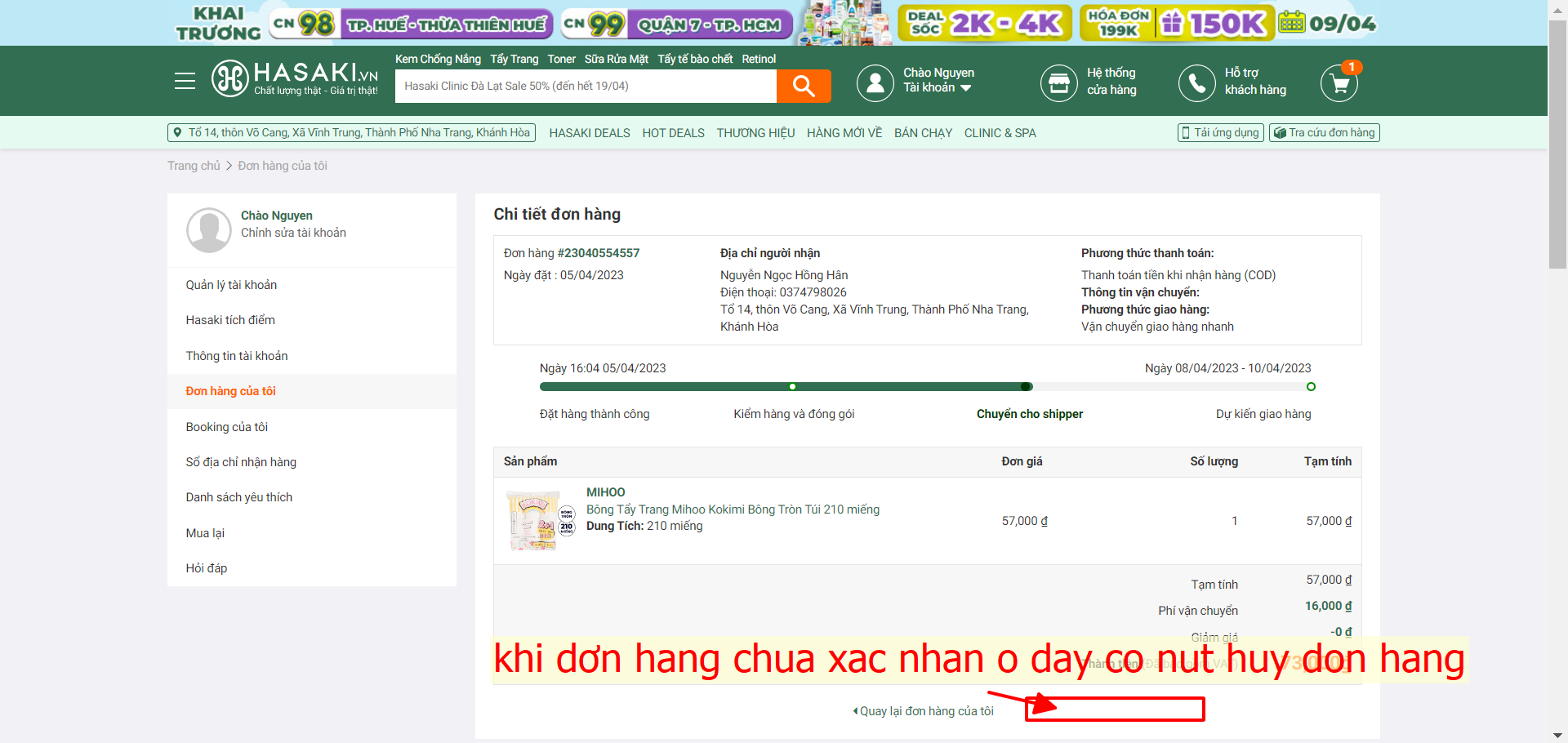
|  |  |  |
| --- | --- | --- |
| Thời gian | Nội dung công việc | Link Gitthub |
| 27/02 – 05/03 | Khảo sát thực trạng công việc tại cửa hàng, phân tích thực trạng, vấn đề cần giải quyết, mục tiêu phát triển của hệ thống | <https://github.com/NguyenNgocHongHan/MINHTHUShop/tree/dev> |
| 06/03 – 19/03 | Phân tích hệ thống, vẽ sơ đồ Use Case |
| 20/03 – 26/03 | Thiết kế CSDL và tìm hiểu về ôn lại các kiến thức về ASP.NET |
| 27/03 – 02/04 | Chỉnh sửa CSDL, tìm hiểu angular và tạo project |
| 03/04 – 09/04 | * Dựng phần Domain Entity cho dự án * Thiết kế Class cho phần Entity Framework Code First * Triển khai Migration, Repository và Unit Of Work cho Data Access Layer |
| 10/04 – 16/04 |
| 17/04 – 23/04 | * Triển khai phần Service và dựng thử test case trong UnitTest * Triển khai các API cho Product và ProductCategory bằng AutoMapper * Identity cho người dùng bằng project mẫu Identity |
| 24/04 – 30/04 | * Triển khai phần quản trị bằng Angular * Tìm kiếm dữ liệu trong bảng * Thêm, sửa, xóa dữ liệu |
| 01/05 – 07/05 | * Tạo form đăng nhập quản trị và điều hướng * Tạo danh sách sản phẩm và phân trang * Sắp xếp danh sách theo điều kiện * Tạo trang danh sách sản phẩm, giới thiệu cửa hàng |

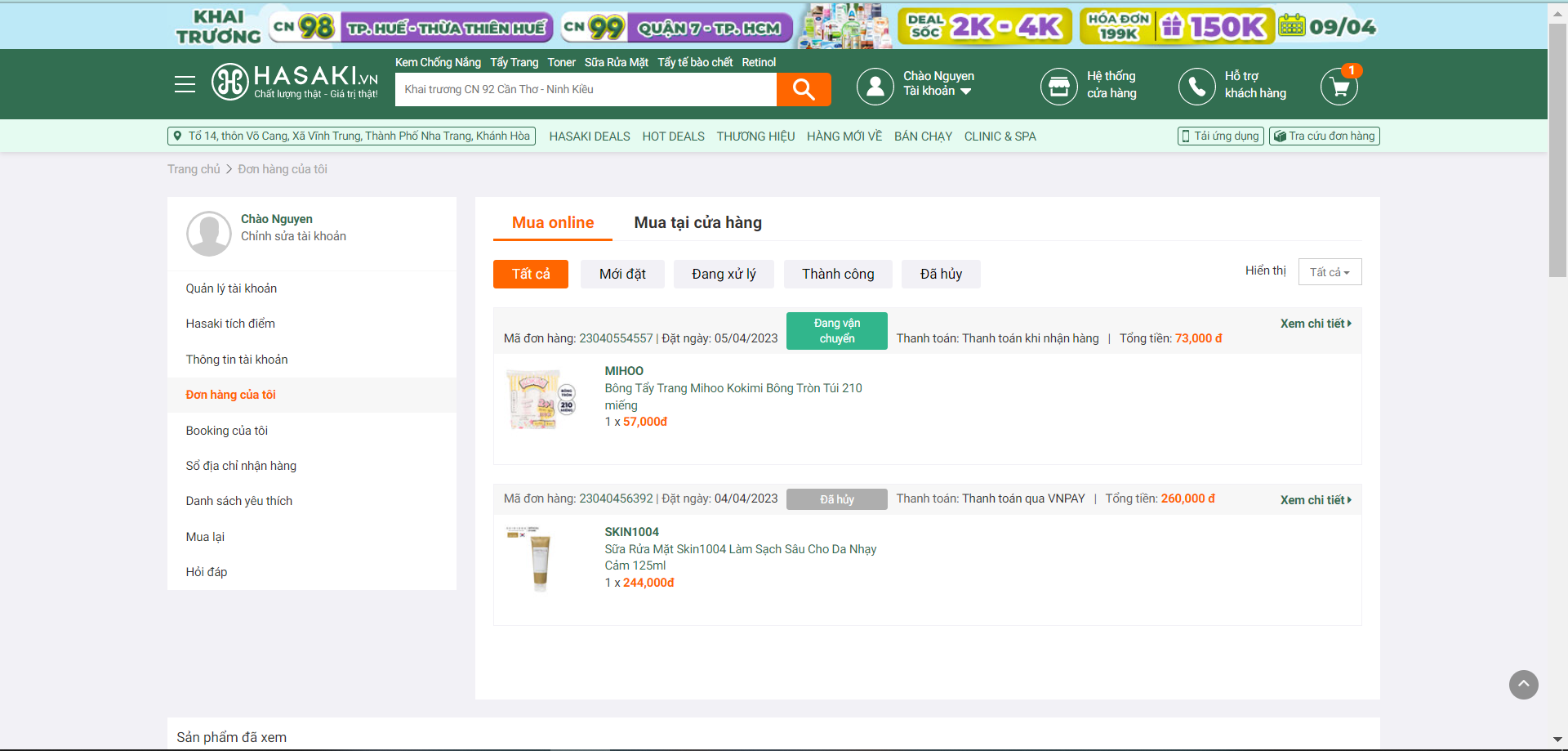
1. **Hình ảnh demo**

Báo cáo trên 50 trang

* ***Giao diện*** website hài hòa, thân thiện, giúp người dùng dễ dàng sử dụng; Trang chủ hiển thị danh sách sản phẩm bán chạy giúp khách hàng có thêm sự chọn lựa







1. **Khảo sát, xác định yêu cầu hệ thống**

* ***Khảo sát hiện trạng cửa hàng***

Đc: 835/1 đường 23/10. Nha Trang.

Shop Minh Thư là cửa hàng chuyên về mỹ phẩm chính hãng (Creave, SVR, La Roche Posay, Innisfree, Simple…) và giá thành hợp lý. Nếu khách hàng không có thời gian đến cửa hàng thì có thể liên lạc qua trang MXH Facebook để lựa chọn và đặt mua sản phẩm, 🡺 tạo nên sự bất tiện cho khách hàng khi chọn lựa sản phẩm nếu không có danh mục sản phẩm, và không có mục feckback của các khách hàng khác…

Cơ cấu nhân sự: chủ cửa hàng vừa trực tiếp nhập hàng từ các đại lý cửa hàng mỹ phẩm, bán hàng và kiểm kho dẫn đến nhiều sổ sách, giấy tờ nhập hàng gây nên bất tiện, thông tin khi cập nhật giá cả và hàng hóa thủ công, bất tiện trong việc kiểm soát hàng hóa tồn kho, gần date

* ***Mục tiêu phát triển hệ thống***

Phấn đấu trở thành chuỗi cửa hàng bán lẽ mỹ phẩm chuyên nghiệp, phục vụ được nhiều khách hàng hơn trong tương lai

Thiết kế 1 website có đầy đủ chức năng của 1 trang web thương mại, mang lại sự hài lòng cho khách hàng khi chọn lựa sản phẩm, ngoài ra còn giúp chủ cửa hàng dễ dàng thống kê doanh số cũng như số liệu sản phẩm, dễ dàng hơn trong việc quản lý cửa hàng

* ***Phương pháp khảo sát***

Quan sát trực tiếp, phỏng vấn cá nhân

* ***Nội dung khảo sát***

Tiếp cận với nghiệp vụ chuyên môn trong nhập hàng, bán hàng của shop Minh Thư, tìm hiểu cách thức hoạt động của cửa hàng, qua đó rút ra được những chức năng quan trọng cần có trong hệ thống; tìm ra những mặt hạn chế tồn tại trong cửa hàng để đề ra hướng giải quyết, nghiên cứu xây dựng hệ thống

Đối tượng khảo sát: Chủ cửa hàng, khách hàng, đối tượng tài liệu như sổ sách, hóa đơn,…

* ***Yêu cầu hệ thống:***

*- Chức năng của khách hàng*

+ Đăng ký, đăng nhập, đăng xuất.

+ Chỉnh sửa tài khoản.

+ Xem thông tin chi tiết sản phẩm.

+ Xem danh sách đơn hàng.

+ Xem chi tiết đơn hàng.

+ Xem lịch sử đơn hàng.

+ Xem các đánh giá của sản phẩm.

+ Xem các tin tức, bài viết từ cửa hàng.

+ Xem các sản phẩm hoặc bài viết khác thông qua các tag liên quan.

+ Xem các câu hỏi thường gặp.

+ Xem thông tin liên hệ cửa hàng.

+ Theo dõi trạng thái đơn hàng.

+ Gửi phản hồi, đóng góp ý kiến cho cửa hàng.

+ Gửi bình luận đánh giá các sản phẩm đã mua

+ Thêm vào giỏ hàng.

+ Đặt hàng.

+ Quản lý giỏ hàng.

+ Tìm kiếm sản phẩm.

+ Thanh toán sản phẩm.

+ Liên hệ chăm sóc khách hàng.

*- Chức năng của nhân viên*

+ Đăng nhập, đăng xuất.

+ Quản trị sản phẩm, nhà cung cấp, thương hiệu, bình luận sản phẩm, feedback, tin tức, các câu hỏi thường gặp.

+ Quản trị danh mục sản phẩm, danh mục tin tức, danh mục các câu hỏi thường gặp.

+ Quản trị các tag.

+ Quản trị đơn hàng, hình thức vận chuyển, hình thức thanh toán.

+ Quản trị banner, menu, thông tin giới thiệu cửa hàng.

+ Quản trị tài khoản khách hàng.

*- Chức năng của người quản trị*

+ Đăng nhập, đăng xuất.

+ Quản trị sản phẩm, nhà cung cấp, thương hiệu, bình luận sản phẩm, feedback, tin tức, các câu hỏi thường gặp.

+ Quản trị danh mục sản phẩm, danh mục tin tức, danh mục các câu hỏi thường gặp.

+ Quản trị các tag.

+ Quản trị đơn hàng, hình thức vận chuyển, hình thức thanh toán.

+ Quản trị banner, menu, thông tin giới thiệu cửa hàng.

+ Quản trị tài khoản khách hàng.

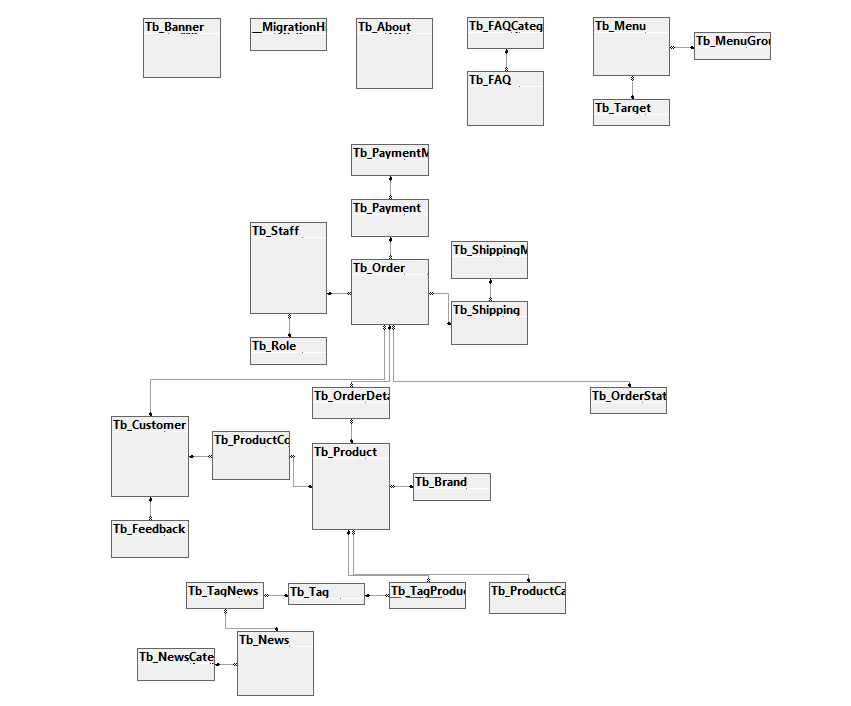
+ Quản trị tài khoản nhân viên.

+ Quản trị thông số cấu hình hệ thống.

+ Phân quyền người dùng.

+ Thống kê doanh thu.

1. **Cơ sở dữ liệu**



## Tb\_Brand: bảng hãng sản xuất, thương hiệu

BrandID: mã thương hiệu (khóa chính, tự động tăng)

Name: tên thương hiệu

BrandOrigin: tên nguồn gốc, xuất xứ thương hiệu

## Tb\_ProductCategory: bảng danh mục sản phẩm

CateID: mã danh mục (khóa chính, tự động tăng)

Name: tên danh mục sản phẩm

Sort: sắp xếp danh mục

ParentID: xây dựng danh mục phân cấp theo cấp cha, con (bằng NULL thì là cấp cha, bằng ID là con của ID đó)

## Tb\_Product: bảng danh mục sản phẩm

ProductID: mã sản phẩm (khóa chính, tự động tăng)

CateID: mã danh mục sản phẩm (khóa ngoại, liên kết với bảng Tb\_ProductCategory)

BrandID: mã hãng sản xuất (khóa ngoại, liên kết với bảng Tb\_Brand)

Name: tên sản phẩm

Image: ảnh đại diện của sản phẩm (ảnh lớn)

ListImage: ảnh nhỏ bên dưới

Price: giá sản phẩm

PromotionPrice: giá khuyến mãi

Description: Tóm tắt để đề cập đến những điểm nổi bật chính của sản phẩm (mô tả ngắn)

Detail: mô tả chi tiết sản phẩm

CreateDate: ngày tạo sản phẩm (mặc định là getdate())

MetaTitle: tiêu đề sản phẩm

MetaKeywords: từ khóa sản phẩm

MetaDescriptions: mô tả sản phẩm - mô tả nội dung xuất hiện bên dưới tên sản phẩm (được dùng cho trình duyệt và SEO)

Status: trạng thái hiển thị (Đang hiển thị/Ẩn, mặc định bằng 1 - true)

## Tb\_ProductComment: bảng quản lý bình luận sản phẩm

CommentID: mã bình luận tin tức (khóa chính, tự động tăng)

ProductID: mã sản phẩm được bình luận (khóa ngoại, liên kết với bảng Tb\_Product)

CustomerID: mã khách hàng (khóa ngoại, liên kết với bảng Tb\_Customer)

Vote: điểm đánh giá sản phẩmn

Comment: nhận xét về sản phẩm

CreateDate: ngày đăng bình luận (mặc định là getdate())

Status: trang thái bình luận (Đang hiển thị/Ẩn, mặc định bằng 1)

## Tb\_Tag: bảng quản lý danh mục nhóm các nội dung liên quan lại với nhau

TagID: mã tag (khóa chính, tự động tăng)

Name: tên tag (phân loại và tìm kiếm các sản phẩm hoặc tin tức liên quan)

## Tb\_TagProduct: bảng Tag của sản phẩm

TagProductID: mã liên kết giữa sản phẩm và tag trong hệ thống (khóa chính, tự động tăng)

TagID: mã tag (khóa ngoại, liên kết với bảng Tb\_Tag)

ProductID: mã sản phẩm (khóa ngoại, liên kết với bảng Tb\_Product)

## Tb\_TagNews: bảng Tag của tin tức

TagNewsID: mã liên kết giữa tin tức và tag trong hệ thống (khóa chính, tự động tăng)

TagID: mã tag (khóa ngoại, liên kết với bảng Tb\_Tag)

NewsID: mã tin tức (khóa ngoại, liên kết với bảng Tb\_News)

## Tb\_Customer: bảng thông tin khách hàng

CustomerID: mã khách hàng (khóa chính, tự động tăng)

Email: email khách hàng

Password: mật khẩu đăng nhập tài khoản khách hàng

Name: họ tên khách hàng

Phone: số điện thoại khách hàng

Address: địa chỉ khách hàng

Gender: giới tính khách hàng (Nam/Nữ)

DateOfBirth: ngày sinh khách hàng

Avatar: ảnh đại diện

CreateDate: ngày tạo tài khoản (mặc định là getdate())

IsLoggedIn: trạng thái đăng nhập tài khoản của khách hàng (Đang đăng nhập/Đã đăng xuất)

LastLogin: thời gian truy cập cuối cùng của tài khoản (UPDATE Tb\_Customer SET LastLogin = getdate() WHERE IsLoggedIn = 1)

Status: trạng thái tài khoản (Được phép hoạt động/Không được phép hoạt động, mặc định bằng true)

## Tb\_Staff: bảng thông tin nhân viên

StaffID: mã nhân viên (khóa chính, tự động tăng)

RoleID: quyền truy cập (admin/ nhân viên, khóa ngoại, liên kết với bảng Tb\_Role)

UserName: tên đăng nhập

Password: mật khẩu đăng nhập tài khoản nhân viên

FullName: họ tên nhân viên

Email: email nhân viên

Phone: số điện thoại nhân viên

Gender: giới tính nhân viên (Nam/Nữ)

Address: địa chỉ nhân viên

DateOfBirth: ngày sinh nhân viên

Avatar: ảnh đại diện

CreateDate: ngày tạo tài khoản nhân viên (mặc định là getdate())

IsLoggedIn: trạng thái đăng nhập của nhân viên (Đang đăng nhập/Đã đăng xuất)

LastLogin: thời gian truy cập cuối cùng của tài khoản (UPDATE Tb\_Staff SET LastLogin = getdate() WHERE IsLoggedIn = 1)

Status: trạng thái tài khoản (Được phép hoạt động/Không được phép hoạt động, mặc định bằng 1)

## Tb\_Role: bảng phân quyền nhân viên

RoleID: mã phân quyền nhân viên (khóa chính, tự động tăng)

Name: tên chức vụ phân quyền (admin/ nhân viên bán hàng)

Description: mô tả thêm về chức vụ được phân quyền

## Tb\_Feedback: bảng đóng góp ý kiến của khách hàng

FeedbackID: mã feedback (khóa chính, tự động tăng)

CustomerID: mã khách hàng (khóa ngoại, liên kết với bảng Tb\_Customer)

Message: nội dung phản hồi/ đóng góp ý kiến

CreateDate: ngày gửi phản hồi (mặc định là getdate())

IsRead: trạng thái đọc phản hồi (Đã đọc/Chưa đọc, mặc định bằng fasle)

## Tb\_OrderStatus: bảng trạng thái đơn hàng

OrderStatusID: mã trạng thái đơn hàng (khóa chính, tự động tăng)

Name: tên trạng thái đơn hàng ("Đang chờ xử lý", "Đã xác nhận", "Kiểm hàng và đóng gói", "Đang vận chuyển", "Đã thanh toán", "Đã hủy bỏ", v.v.)

Description: mô tả chi tiết về trạng thái đơn hàng

## Tb\_Order: bảng đơn đặt hàng

OrderID: mã đơn đặt hàng (khóa chính, tự động tăng)

CustomerID: mã khách đặt hàng (khóa ngoại, liên kết với bảng Tb\_Customer)

StaffID: mã nhân viên phê duyệt đơn hàng (khóa ngoại, liên kết với bảng Tb\_Staff)

*// = User.Identity.Name;*

OrderStatusID: mã trạng thái đơn hàng (khóa ngoại, liên kết với bảng Tb\_OrderStatus)

ShippingMethodID: mã phương thức vận chuyển (khóa ngoại, liên kết với bảng Tb\_ShippingMethod)

PaymentMethodID: mã phương thức thanh toán (khóa ngoại, liên kết với bảng Tb\_PaymentMethod)

Total: tổng số tiền trong đơn hàng (gồm tổng tiền hàng + tiền phí vận chuyển)

CreateDate: ngày đặt hàng (mặc định là getdate())

Note: ghi chú của khách hàng về đơn hàng

IsCancel: khách hàng hủy đơn hàng (Hủy đơn hàng/Không hủy đơn hàng (Đơn hàng chỉ được hủy khi trạng thái đơn hàng "Đang chờ xử lý"), mặc định bằng false)

## Tb\_OrderDetail: bảng chi tiết đơn đặt hàng

OrderDetailID: mã chi tiết đơn đặt hàng (khóa chính, tự động tăng)

OrderID: mã đơn đặt hàng (khóa ngoại, liên kết với bảng Tb\_Order)

ProductID: mã sản phẩm (khóa ngoại, liên kết với bảng Tb\_Product)

Quantity: số lượng sản phẩm trong đơn hàng

## Tb\_ShippingMethod: bảng phương thức vận chuyển

ShippingMethodID: mã phương thức vận chuyển (khóa chính, tự động tăng)

Name: tên phương thức vận chuyển (giao hàng nhanh, giao hàng tiết kiệm, v.v)

Description: mô tả phương thức vận chuyển

Cost: chi phí vận chuyển

Status: trạng thái của các phương thức vận chuyển (Đang hoạt động/Ngừng hoạt động, mặc định bằng 1)

## Tb\_Shipping: bảng vận chuyển

ShippingID: mã vận chuyển (khóa chính, tự động tăng)

OrderID: mã đơn hàng (khóa ngoại, liên kết với bảng Tb\_Order)

ShippingDate: ngày vận chuyển

StartDate: dự kiến ngày đầu tiên nhận hàng (sau 3 ngày kể từ ngày vận chuyển)

EndDate: dự kiến ngày cuối cùng nhận hàng (sau 5 ngày kể từ ngày vận chuyển)

IsShipping: trạng thái vận chuyển đơn hàng (Đơn hàng đã được chuyển cho Shipper/Đơn hàng chưa được chuyển cho Shipper, mặc định bằng false)

## Tb\_PaymentMethod: bảng phương thức thanh toán

PaymentMethodID: mã phương thức thanh toán (khóa chính, tự động tăng)

Name: tên phương thức thanh toán (thẻ tín dụng, PayPal, thanh toán khi nhận hàng (COD), chuyển khoản ngân hàng, ví điện tử,…)

Description: mô tả phương thức thanh toán

Status: trạng thái của các phương thức thanh toán (Đang hoạt động/Ngừng hoạt động, mặc định bằng 1)

## Tb\_Payment: bảng lưu trữ thông tin về các khoản thanh toán

PaymentID: mã khoản thanh toán (khóa chính, tự động tăng)

OrderID: mã hóa đơn (khóa ngoại, liên kết với bảng Tb\_Order)

PaymentDate: ngày thanh toán

PaymentAmount: số tiền thực tế khách hàng đã thanh toán cho đơn hàng tương ứng, bao gồm cả các khoản phí và chiết khấu (nếu có) (khác với số tiền Total trong bảng Tb\_Order)

IsPaid: trạng thái thanh toán đơn hàng (Đã thanh toán/Chưa thanh toán, mặc định bằng false)

## Tb\_NewsCategory: bảng danh mục các tin tức

NewsCateID: mã danh mục tin tức (khóa chính, tự động tăng)

Name: tên danh mục tin tức

Sort: sắp xếp danh mục

ParentID: xây dựng danh mục theo cấp cha, con (bằng NULL thì là cấp cha, bằng ID là con của ID đó)

## Tb\_News: bảng tin tức

NewsID: mã tin tức (khóa chính, tự động tăng)

NewsCateID: mã danh mục tin tức (khóa ngoại, liên kết với bảng Tb\_NewsCategory)

Name: tên bài viết

Image: ảnh tin tức

Description: nội dung tin tức

CreateDate: ngày đăng tin tức (mặc định là getdate())

MetaTitle: tiêu đề tin tức

MetaKeywords: từ khóa tin tức

MetaDecriptions: mô tả nội dung xuất hiện bên dưới tên tin tức (được dùng cho trình duyệt và SEO)

Status: trạng thái của tin tức (Đang hiển thị/Ẩn, mặc định bằng true)

## Tb\_FAQCategory: bảng danh mục các câu hỏi thường gặp

FAQCateID: mã danh mục các câu hỏi thường gặp (khóa chính, tự động tăng)

Name: tên danh mục các câu hỏi thường gặp

Sort: sắp xếp danh mục

ParentID: xây dựng danh mục theo cấp cha, con (bằng null thì là cấp cha, bằng id là con của id đó)

## Tb\_FAQ: bảng quản lý thông tin về câu hỏi thường gặp của khách hàng

FAQID: mã của câu hỏi thường gặp (khóa chính, tự động tăng)

FAQCateID: mã danh mục của câu hỏi thường gặp (khóa ngoại, liên kết với bảng FAQCategory)

Question: câu hỏi thường gặp

Answer: câu trả lời cho câu hỏi thường gặp

MetaTitle: tiêu đề câu hỏi thường gặp sẽ được hiển thị trên Trang danh mục tin tức

MetaKeywords: khai báo từ khóa muốn truyền tải (tương tự hashtag)

MetaDecriptions: mô tả nội dung xuất hiện dưới danh mục (tương tự như đoạn mô tả xuất hiện dưới phần title URL mô tả nội dung URL đó, được dùng cho trình duyệt và SEO)

Status: trạng thái của câu hỏi thường gặp (Đang hiển thị/Ẩn, mặc định bằng true)

## Tb\_Config: bảng lưu các thông số cấu hình của hệ thống

ConfigID: mã cấu hình (khóa chính, tự động tăng)

Name: tên cấu hình (số lượng sản phẩm trên một trang, địa chỉ email của quản trị viên, cấu hình kết nối cơ sở dữ liệu, v.v)

ValueNum: giá trị của cấu hình dạng số

ValueString: giá trị cài đặt cấu hình dạng chuỗi

Description: mô tả chi tiết về cấu hình

Status: trạng thái cấu hình (Đang hoạt động/Ngừng hoạt động, mặc định bằng 1)

## Tb\_Banner: bảng quản lý các banner quảng cáo trên trang chủ

BannerID: mã banner (khóa chính, tự động tăng)

Name: tên của banner

Description: mô tả về banner

Image: hình ảnh banner

Link: đường dẫn khi người dùng click vào banner

Sort: sắp xếp thứ tự hiển thị các banner

Position: vị trí hiển thị của banner trên trang web

CreateDate: ngày tạo banner (mặc định là getdate())

Status: trạng thái của banner (Đang hoạt động/Ngừng hoạt động, mặc định bằng true)

## Tb\_Target: bảng quản lý target (cách mở liên kết khi người dùng nhấp vào mục menu)

TargetID: mã target (khóa chính, tự động tăng)

Name: tên target ("\_blank", "\_self", "\_parent", "\_top")

Description: mô tả target

* "\_blank": Liên kết sẽ được mở trong một cửa sổ mới.
* "\_self": Liên kết sẽ được mở trong cùng một khung trình duyệt.
* "\_parent": Liên kết sẽ được mở trong khung cha của khung hiện tại.
* "\_top": Liên kết sẽ được mở trong cửa sổ trình duyệt mới và sẽ thay thế toàn bộ nội dung trang web hiện tại.

## MenuGroup: bảng quản lý nhóm menu

MenuGroupID: mã nhóm menu (khóa chính, tự động tăng)

Name: tên nhóm menu

Sort: sắp xếp vị trí của các nhóm menu

## Tb\_Menu: bảng quản lý menu

MenuID: mã menu (khóa chính, tự động tăng)

MenuGroupID: mã nhóm menu (khóa ngoại, liên kết với bảng Tb\_MenuGroup)

TargetID: mã target (khóa ngoại, liên kết với bảng Tb\_Target)

Name: tên các thành phần trong menu

Icon: biểu tượng của mục menu

Description: mô tả thành phần menu

Link: đường dẫn liên kết tới trang web

Sort: sắp xếp vị trí của các thành phần trong menu

Status: trạng thái hiển thị (Đang hiển thị/Ẩn, mặc định bằng true)

## Tb\_About: bảng giới thiệu thông tin về cửa hàng

AboutID: mã giới thiệu cửa hàng (khóa chính, tự động tăng)

Name: tên cửa hàng

Phone: số điện thoại cửa hàng

Email: email liên hệ cửa hàng

Fanpage: tài khoản mạng xã hội của cửa hàng

Address: địa chỉ cửa hàng

Description: mô tả nội dung giới thiệu về cửa hàng

MetaTitle: tiêu đề trang giới thiệu

MetaKeywords: từ khóa về trang giới thiệu cửa hàng (tương tự hashtag)

MetaDecriptions: mô tả trang giới thiệu (tương tự như đoạn mô tả xuất hiện dưới phần title URL mô tả nội dung URL đó, được dùng cho trình duyệt và SEO)

Status: trạng thái hiển thị thông tin giới thiệu, liên lạc (Đang hiển thị/Ẩn, mặc định bằng true)

## Tb\_Page: bảng quản lý các trang (page)

PageID: mã trang page của website (khóa chính, tự động tăng)

Name: tên trang web

Description: mô tả trang web

URL: địa chỉ URL của trang web

CreateDate: ngày tạo trang web

MetaTitle: tiêu đề trang web

MetaKeywords: từ khóa về trang web (tương tự hashtag)

MetaDecriptions: mô tả trang web (tương tự như đoạn mô tả xuất hiện dưới phần title URL mô tả nội dung URL đó, được dùng cho trình duyệt và SEO)

## Tb\_Error: bảng các lỗi của website

ErrorID: mã lỗi (khóa chính, tự động tăng)

Message: nội dung lỗi

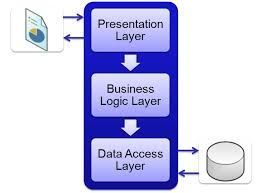
StackTrace: chứa danh sách các lệnh gọi phương thức (method) dẫn đến ngoại lệ (exception) / lỗi (error), cùng với tên file và số dòng,…

CreateDate: ngày tạo lỗi (mặc định getdate())

1. **Tổng quan**

## Cấu trúc dự án

* *Xây dựng dự án theo sự kết hợp của mô hình 3 lớp và MVC*: là một cách tiếp cận phổ biến để phát triển các ứng dụng phức tạp và web, giúp cho việc phân chia và quản lý mã nguồn trở nên rõ ràng hơn, đồng thời giúp cho việc phát triển và bảo trì ứng dụng trở nên dễ dàng hơn. Việc sử dụng cả hai mô hình giúp cho việc phát triển và bảo trì ứng dụng trở nên dễ dàng hơn, đồng thời giúp cho việc phân chia và quản lý mã nguồn trở nên rõ ràng hơn.
  + Mô hình 3 lớp là một kiến trúc phần mềm được sử dụng để phát triển các ứng dụng phức tạp. Trong mô hình này, ứng dụng được chia thành 3 lớp chính: giao diện người dùng (Presentation Layer), logic nghiệp vụ (Business Layer) và cơ sở dữ liệu (Data Layer). Mỗi lớp đảm nhận một chức năng cụ thể và có thể được phát triển và bảo trì độc lập.



* + MVC (Model-View-Controller) là một mô hình thiết kế phần mềm được sử dụng để phát triển các ứng dụng web. Trong mô hình này, ứng dụng được chia thành 3 phần chính: Model (mô hình dữ liệu), View (giao diện người dùng) và Controller (bộ điều khiển). Mỗi phần đảm nhận một chức năng cụ thể và có thể được phát triển và bảo trì độc lập.
* Mục đích là có thể sử dụng lại code ở phần server side cho cả trang web dành cho khách hàng và trang web dành cho nhân viên.

## Back-end: RESTFULL WEB API

* Back-end là toàn bộ code server side bao gồm Entity Framework, Web API, phần xử lý Linq Entity, phần xử lý code C# để đẩy ra ngoài giao diện.
* ASP.NET MVC 5
* Web API 2
* Visual Studio 2019 (tải Codemaid, Bundler & Minifier)
  + CodeMaid là một VS Extensions mã nguồn mở giúp dọn dẹp và đơn giản hóa code
* SignalR
* ASP.NET Identity 2
* OAuth
* SQL Sever
* SQL Server Management Studio 18

## Front-end: ANGULAR 2 CLI

* Front-end là AngularJS, client side, JavaScripts, Boostrap, Html
* Angular 2 (chính)
* Angular CLI (giao diện dòng lệnh, tương tác với Angular)
* TypeScript
* Webpack
* Boostrap 3 (Html)
* Visual Studio Code

## Công cụ sử dụng

* SQL Server Management Studio 18
* Visual Studio 2019 - Community Edition
* NodeJs
* Bower (Bower là công cụ quản lí các nguồn tài nguyên cho lập trình viên fornt-end. Tài nguyên như là: HTML, CSS, JS, font chữ và hình ảnh)

## Công nghệ sử dụng

* Truy xuất dữ liệu: Entity Framework Code First
* Resfull API: Web API
* Website dành cho khách hàng: ASP.NET MVC thuần (Razor - với công nghệ này thì chúng ta sẽ dùng dấu @ để chèn mã C# vào file View; ViewBags; Models)
* Phần quản trị: AngularJS
* Phần chứng thực: ASP.NET Identity (hỗ trợ tốt cho việc Login với Facebook, Google+ và phân quyền bảo mật)

## Link tham khảo

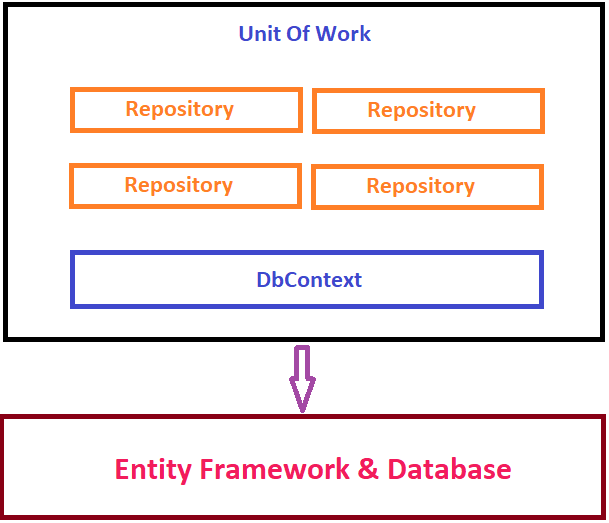
<https://itzone.com.vn/vi/article/thiet-ke-restful-api-goi-api-bang-httpclient-trong-asp-net/>

<https://tuhocict.com/entity-framework-code-first/>

* Code import từ form của ProductAPIController: <https://viblo.asia/p/aspnet-truyen-du-lieu-tu-form-den-web-api-ZK1ov1w1M5b9>

# Thực hiện dự án

## Các thành phần của solution (tạo 1 Blank Solution)

* *MINHTHUShop.Common* (Class Library): Chứa các lớp tiện ích dùng chung cho dự án
* *MINHTHUShop.Model* (Class Library): Chứa các lớp Domain Entities (các thực thể) của dự án
  + Cài đặt các package
    - Install package EntityFramework
    - Install package Microsoft.AspNet.Identity.Core
    - Install package Microsoft.AspNet.Identity.EntityFramework
  + Bao gồm các thư mục:
    - *Abstract:* chứa các class có các thuộc tính giống nhau để các lớp trong thư mục Models kế thừa
      * Class *ISEO* và *SEO*: chứa những trường thuộc tính về SEO
    - *Models*: chứa các lớp thực thể của dự án (dựa trên CSDL ở [phần 4)](#_Tb_Brand:_bảng_hãng)
* *MINHTHUShop.Data* (Class Library): Chứa các tầng truy cập dữ liệu sử dụng Entity Framework Code First
  + Cài đặt các package
    - Install package EntityFramework
    - Install package Microsoft.AspNet.Identity.EntityFramework
  + References
    - MINHTHUShop.Common
    - MINHTHUShop.Model
  + Bao gồm các thư mục:
    - 
    - *Infrastructure*:
      * Lớp IRepository, IUnitOfWork, IdbFactory: định nghĩa ra các giao tiếp cho các class
      * *RepositoryBase, IRepository*: là 1 lớp trừu tượng nằm giữ business logic và data access bằng cách tập trung vào các hoạt động CRUD của các thực thể (entity)
      * *UnitOfWork, IUnitOfWork*: được sử dụng để tất cả các hoạt động thực hiện trong 1 transaction, đảm bảo tính toàn vẹn của dữ liệu
      * *Disposable*: chứa phương thức tự động hủy để giải phóng bộ nhớ (tham khảo ở: <https://learn.microsoft.com/en-us/dotnet/standard/garbage-collection/implementing-dispose>)
      * *DbFactory, IDbFactory*: khởi tạo các đối tượng thông qua DbContext để truy cập cơ sở dữ liệu
    - *Migrations*: chứa lịch sử thay đổi của migration và connect đến database (Link tham khảo: <https://www.entityframeworktutorial.net/code-first/code-based-migration-in-code-first.aspx>)
    - Repositories: chứa các lớp repository của các thực thể
    - Class *MINHTHUShopDbContext*: chứa tập thực thể (DbSet<TEntity>) cho tất cả thực thể nối với những bảng của CSDL.
* MINHTHUShop.Service (Class Library): Chứa các lớp Service xử lý Business logic
  + References
    - MINHTHUShop.Common
    - MINHTHUShop.Data
    - MINHTHUShop.Model
* MINHTHUShop.UnitTest (Unit Test Project): Chứa các class Test sử dụng cho việc Unit Test
  + Cài đặt các package
    - Install package Moq version 4.5.30
    - Install package Entity Framework
  + References
    - MINHTHUShop.Common
    - MINHTHUShop.Data
    - MINHTHUShop.Model
    - MINHTHUShop.Service
    - MINHTHUShop.Web
  + Chứa các thư mục
    - *RepositoryTest*: chứa các test case của Repository để thao tác trực tiếp trên database
    - *ServiceTes*t: chứa các test case của Service, dùng phương thức mock để tạo ra đối tượng ảo để test mà không cần thay đổi dữ liệu trên database
* MINHTHUShop.Web (ASP.NET Web Application): Project chính dùng để hiển thị giao diện và tương tác người dùng
  + Cài đặt các package
    - Install package Autofac version 3.5.2
    - Install package Autofac.WebApi2 version 4.0.0
    - Install package Autofac.Mvc5 version 3.3.4
    - Install package AutoMapper version 5.1.1
    - Install package Microsoft.Owin.Host.SystemWeb
    - Install package EntityFramework
    - Install package Microsoft.AspNet.Identity.EntityFramework
    - Install package Microsoft.AspNet.Identity.Owin
  + References
    - MINHTHUShop.Common
    - MINHTHUShop.Data
    - MINHTHUShop.Model
    - MINHTHUShop.Service
  + Chứa các thư mục
    - *API*: chứa các class API Controller để gọi RESTfull API cho Admin
    - *App\_Start:* thư mục chưa các file cấu hình khởi động và biên dịch của project
    - *Content:* chứa các thư viện định dạng cho trang web
    - *Controllers:* chứa các class controller để xử lý yêu cầu và phản hồi cho người dùng
    - *fonts:* chứa các tập tin font files của trang web
    - *Infrastructure:* chứa các class thao tác dùng chung cho trang web
      * *Core*
      * *Extensions*: chứa class để lưu các entity dữ liệu trong db từ entity view model được truyền vào
    - *Mappings:* chứa file ánh xạ để chuyển đổi dữ liệu từ một entity trong database sang một entity khác
    - *Models:* khác với MINHTHUShop.Model là domain model, thư mục Models gồm các class ViewModel thể hiện các thuộc tính cần thiết để hiển thị cho người dùng
    - *Scripts:* chứa các tập tin JavaScript hay VBScript
    - *Views:* chứa các file Html