**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI KHOA CÔNG NGHỆ THÔNG TIN**



BÁO CÁO ĐỒ ÁN

**XÂY DỰNG WEBSITE BÁN ĐỒNG HỒ TOPWATCH**

**GVHD: Ths. Mai Thanh Hồng**

**SVTH: Bùi Việt Anh**

**LỚP:** [**2019DHCNTT01**](https://classroom.google.com/u/0/c/MjI4MjA2NDcwNTE2)

**KHÓA: 14**

**HÀ NỘI – 2023**

# Lời cảm ơn

Ngày nay, công nghệ thông tin đang được ứng dụng rộng rãi trên mọi lĩnh vực của cuộc sống. Với tốc độ nhanh và chính xác, máy tính đã giúp con người tận dụng hiệu quả thời gian và chất lượng công việc. Bán hàng online hiện nay là một trong những lĩnh vực áp dụng công nghệ thông tin rộng rãi. Bằng việc áp dụng khả năng tính toán và lưu trữ của máy tính điện tử, việc quản lý và điều phối hoạt động bán hàng đã dễ dàng và chính xác hơn trước đây rất nhiều. Vốn là công việc đòi hỏi nhiều yêu cầu và ràng buộc về hệ thống để đảm bảo nhân viên và quản lý lên các mặt hàng giúp khách hang có thể tiếp cận sản phẩm một cách dễ dàng hơn, người bán cũng có thể dễ dàng quản lý các mặt hang và đơn hàng, sẽ không tốn chi phí thuê mặt bằng. Do vậy, em đã chọn đề tài ‘‘*Xây dựng website bán đồng hồ cho cửa hàng TopWatch*’’ nhằm diễn tả một phần nào đó việc bán hàng online của một cửa hàng ngày nay.

Đây là một phần mềm quản lý với các tính năng phục vụ việc quản lý và xử lý các mặt hàng, khách hàng và đơn hàng. Phần mềm cung cấp những chức năng như Quản lý khách hàng, Quản lý sản phẩm, Quản lý chi tiết sản phẩm, Quản lý đơn hàng,… cho cho phép nhân viên và quản lý dễ dàng theo dõi được diễn biến hoạt động kinh doanh cửa hàng, dễ dàng phát hiện các vấn đề phát sinh, nhanh chóng khắc phục sự cố.

Để hoàn thành đồ án này em cần phải tìm hiểu, nghiên cứu phân tích quy trình bán hàng, giao hàng và các website bán hàng online thông qua Internet. Bên cạnh đó, việc trang bị những kiến thức C# / .NET là vô cùng cần thiết và quan trọng cho bài tập lớn này.

* Lập trình hướng đối tượng

Cần nắm vững các kiến thức về lớp và đối tượng trong C# / .NET. Các kiến thức liên quan đến lớp như thuộc tính, phương thức của lớp, phương thức tạo, tính đóng gói, kế thừa, đa hình.

* SQL Server

Tất cả dữ liệu của chương trình về admin, phòng, nhân viên,... được lưu trong SQL Server để tiện lợi truy xuất và cập nhật.

* Ngoài ra, các kiến thức về phân tích thiết kế hệ thống, thiết kế cơ sở dữ liệu,

các kỹ thuật lập trình và các thao tác với cơ sở dữ liệu như tạo bảng, insert, update, delete dữ liệu cũng là các công cụ cần thiết để em hoàn thiện đồ án này.

# Tóm tắt

Với mức độ phức tạp và quy mô ứng dụng, cộng thêm vấn đề thời gian cho nên đề tài “ Xây dựng Website bán đồng hồ ” chỉ dừng ở mức tìm hiểu ngôn ngữ lập trình Web ASP.NET MVC & SQL, và áp dụng xây dựng ứng dụng thực nghiệm trang Web bán hàng trên mạng.

Website với mục đích cung cấp cho khách hàng các thông tin chính xác về các sản phẩm đồng hồ và linh kiện đang có trên thị trường của các hãng có uy tín như (Rolex, Iwc, Omega, Tag Heuer, Rotary...) và cách thức đặt mua hàng qua mạng. Các thông tin được cập nhật thường xuyên và nhanh chóng. Vì vậy, rút ngắn được khoảng cách giữa người mua và người bán, đưa thông tin về các sản phẩm mới nhanh chóng đến cho khách hàng.

Do còn một số hạn chế, nên website mới chỉ dừng lại ở chức năng đặt hàng và hình thức thanh toán trực tiếp. Trong tương lai, hệ thống sẽ phát triển theo hướng thương mại điện tử (với hình thức thanh toán trực tuyến qua thẻ tín dụng, tài khoản ngân hàng, thẻ mua hàng trực tuyến...).

# Mục lục

[Lời cảm ơn 1](#_Toc134098840)

[Tóm tắt 3](#_Toc134098841)

[Mục lục 4](#_Toc134098842)

[Danh mục hình vẽ 8](#_Toc134098843)

[Danh mục bảng 10](#_Toc134098844)

[Danh mục các từ viết tắt 11](#_Toc134098845)

[Chương 1 KHẢO SÁT, CƠ SỞ LÝ THUYẾT XÂY DỰNG BÀI TOÁN 12](#_Toc134098846)

[1.1 Đặt vấn đề 12](#_Toc134098847)

[1.1.1 Nghiệp vụ của hệ thống hiện tại 12](#_Toc134098848)

[1.1.2 Những nhược điểm của hệ thống 12](#_Toc134098849)

[1.2 Định hướng giải pháp 14](#_Toc134098850)

[1.3 Công nghệ sử dụng 15](#_Toc134098851)

[1.3.1 Ngôn ngữ ASP.NET là gì? 15](#_Toc134098852)

[1.3.2 Cấu trúc và các thành phần của ASP.NET 16](#_Toc134098853)

[1.3.2.1 Các đặc điểm cơ bản của ASP.Net framework bao gồm: 17](#_Toc134098854)

[1.3.2.2 Cấu trúc của .Net framework dựa trên các thành phần cơ bản sau: 17](#_Toc134098855)

[1.4 Thiết kế kiến trúc 18](#_Toc134098856)

[1.4.1 Lựa chọn kiến trúc phần mềm 18](#_Toc134098857)

[1.4.2 Thiết kế tổng quan 26](#_Toc134098858)

[Chương 2 PHÂN TÍCH THIẾT KẾ WEBSITE 27](#_Toc134098859)

[2.1 Khảo sát hiện trạng 27](#_Toc134098860)

[2.2 Tổng quan chức năng 27](#_Toc134098861)

[2.2.1 Biểu đồ use case tổng quan 28](#_Toc134098862)

[2.2.2 Biểu đồ phân rã use case quản sản phẩm 29](#_Toc134098863)

[2.2.3 Biểu đồ phân rã use case Quản lý thương hiệu 30](#_Toc134098864)

[2.2.4 Biểu đồ phân rã tổng quát User 31](#_Toc134098865)

[2.2.5 Quy trình nghiệp vụ 31](#_Toc134098866)

[2.3 Đặc tả chức năng 31](#_Toc134098867)

[2.3.1 Đặc tả use case đăng kí 31](#_Toc134098868)

[2.3.2 Đặc tả use case đặt hàng 32](#_Toc134098869)

[2.3.3 Đặc tả use case xem thông tin sản phẩm 33](#_Toc134098870)

[2.3.4 Đặc tả use case quản lý thương hiệu 34](#_Toc134098871)

[2.3.5 Đặc tả use case quản lý sản phẩm 35](#_Toc134098872)

[2.3.6 Đặc tả use case quản lý danh mục 36](#_Toc134098873)

[2.3.7 Đặc tả use case quản lý chương trình khuyến mại 37](#_Toc134098874)

[2.3.8 Đặc tả use case quản lý tài khoản 38](#_Toc134098875)

[2.3.9 Đặc tả use case thêm vào giỏ hàng 38](#_Toc134098876)

[2.3.10 Đặc tả use case tìm kiếm 39](#_Toc134098877)

[2.4 Yêu cầu phi chức năng 40](#_Toc134098878)

[2.5 Thiết kế chi tiết 40](#_Toc134098879)

[2.5.1 Thiết kế giao diện 40](#_Toc134098880)

[2.5.2 Thiết kế lớp 41](#_Toc134098881)

[2.5.2.1 Lớp User (khách hàng) 41](#_Toc134098882)

[2.5.2.2 Lớp sản phẩm 44](#_Toc134098883)

[2.5.2.3 Lớp Thương hiệu 47](#_Toc134098884)

[2.5.3 Thiết kế cơ sở dữ liệu 51](#_Toc134098885)

[Chương 3 CÀI ĐẶT VÀ KIỂM THỬ 52](#_Toc134098886)

[3.1 Xây dựng ứng dụng 52](#_Toc134098887)

[3.1.1 Thư viện và công cụ sử dụng 52](#_Toc134098888)

[3.1.2 Kết quả đạt được 53](#_Toc134098889)

[3.1.3 Minh hoạ các chức năng 54](#_Toc134098890)

[3.2 Kiểm thử phần mềm 60](#_Toc134098891)

[3.2.1 Kiểm thử Trang chủ 60](#_Toc134098892)

[3.2.2 Kiểm thử tìm kiếm 60](#_Toc134098893)

[3.2.3 Kiểm thử trang xem thông tin chi tiết sản phẩm 60](#_Toc134098894)

[3.2.4 Kiểm thử giỏ hàng 60](#_Toc134098895)

[3.2.5 Kiểm thử thanh toán 61](#_Toc134098896)

[3.2.6 Kiểm thử các sản phẩm trong các Category 61](#_Toc134098897)

[3.2.7 Kiểm thử các nội dung sau khi order 61](#_Toc134098898)

[3.2.8 Kiểm thử các nội dung khác 61](#_Toc134098899)

[Chương 4 KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM 62](#_Toc134098900)

[4.1 Những mục tiêu đề ra 62](#_Toc134098901)

[4.2 Nội dung đã thực hiện 62](#_Toc134098902)

[4.3 Bài học kiến thức 63](#_Toc134098903)

[4.4 Hướng phát triển. 63](#_Toc134098904)

[Tài liệu tham khảo 64](#_Toc134098905)

# Danh mục hình vẽ

[Hình 1 *Cấu trúc .Net* 17](#_Toc134099393)

[Hình 2 *Minh họa cấu trúc lập trình web* 18](#_Toc134099394)

[Hình 3*Mô hình MVC* 19](#_Toc134099395)

[Hình 4 *Mô hình ASP.Net MVC* 21](#_Toc134099396)

[Hình 5 *ASP.NET là gì?* 21](#_Toc134099397)

[Hình 6 *cơ chế hoạt động của ASP.NET MVC* 25](#_Toc134099398)

[Hình 7 *Thiết kế tổng quan UML* 26](#_Toc134099399)

[Hình *8 Biểu đồ use case tổng quát* 28](#_Toc134099400)

[Hình 9 *Biểu đồ phân rã Admin tổng quát* 28](#_Toc134099401)

[Hình *11 Biểu đồ phân rã Use Case quản lý sản phẩm* 29](#_Toc134099402)

[Hình 12 *Biểu đồ phân rã Use Case quản lý thương hiệu* 30](#_Toc134099403)

[Hình 13 *Biểu đồ phân rã tổng quát User* 31](#_Toc134099404)

[Hình 14 *Trang danh sách sản phẩm* 41](#_Toc134099405)

[Hình 15 *Đăng nhập* 41](#_Toc134099406)

[Hình 16 *Lớp User* 42](#_Toc134099407)

[Hình 17 *Mô hình tuần tự use case đăng kí* 43](#_Toc134099408)

[Hình 18  *Mô hình tuần tự use case đăng nhập* 43](#_Toc134099409)

[Hình 19 *Lớp sản phẩm* 44](#_Toc134099410)

[Hình 20 *Mô hình tuần tự use case thêm sản phẩm* 45](#_Toc134099411)

[Hình 21 *Mô hình tuần tự use case xóa sản phẩm* 46](#_Toc134099412)

[Hình 22 *Mô hình tuần tự sửa sản phẩm* 47](#_Toc134099413)

[Hình 23 *Lớp thương hiệu* 48](#_Toc134099414)

[Hình 24 *Mô hình tuần tự use case thêm thương hiệu* 48](#_Toc134099415)

[Hình 25 *Mô hình tuần tự xóa thương hiệu* 49](#_Toc134099416)

[Hình 26 *Mô hình tuần tự sửa thông tin thương hiệu* 50](#_Toc134099417)

[Hình 27 *Biểu đồ diagram* 51](#_Toc134099418)

[Hình 28 *Xem thông tin sản phẩm* 54](#_Toc134099419)

[Hình 29 *Đăng nhập* 54](#_Toc134099420)

[Hình 30 *Lịch sử mua hàng* 55](#_Toc134099421)

[Hình 31 *Giỏ hàng* 55](#_Toc134099422)

[Hình 32 *Xem sản phẩm* 56](#_Toc134099423)

[Hình 33 *Xem phụ kiện* 56](#_Toc134099424)

[Hình 34 *Xem khuyến mãi* 57](#_Toc134099425)

[Hình 35 *Chỉnh sửa thông tin cá nhân của người dùng* 57](#_Toc134099426)

[Hình 36 *Thanh toán đơn hàng* 58](#_Toc134099427)

[Hình 37 *Hiện thị thống kê* 58](#_Toc134099428)

[Hình 38 *Trả lời đánh giá sản phẩm* 59](#_Toc134099429)

[Hình 39 *Danh sách đơn hàng* 59](#_Toc134099430)

# Danh mục bảng

[Bảng 1 Danh sách thư viện và công cụ sử dụng 52](#_Toc134098817)

[Bảng 2 Thống kê thông tin về ứng dụng 53](#_Toc134098818)

# Danh mục các từ viết tắt

|  |  |
| --- | --- |
| MVC | Model - View – Controller  Quản lý xử lý dữ liệu – Hiển thị dữ liệu – Điều khiển |
| **CSS** | Cascading Style Sheets  Ngôn ngữ tìm và định dạng các thành phần của HTML |
| **SQL** | **Structured Query Language**  Ngôn ngữ truy vấn có cấu trúc |
| **HTML** | HyperText Markup Language  Ngôn ngữ đánh dấu siêu văn bản |

# KHẢO SÁT, CƠ SỞ LÝ THUYẾT XÂY DỰNG BÀI TOÁN

## Đặt vấn đề

Sau khi đi khảo sát các khách hàng, bạn bè những người hay có nhu cầu đi mua đồng hồ và các thiết bị phụ kiện của đồng hồ. Em đã tìm ra được vấn đề của việc phải đến cửa hàng trực tiếp để mua hàng:

### Nghiệp vụ của hệ thống hiện tại

* Lựa chọn sản phẩm: Khách hàng sẽ phải tự đi tìm sản phẩm mình muốn mua trong cửa hàng và chọn lựa sản phẩm phù hợp với nhu cầu của mình.
* Kiểm tra sản phẩm: Sau khi chọn sản phẩm, khách hàng cần kiểm tra kỹ sản phẩm để đảm bảo sản phẩm không bị lỗi hoặc hư hỏng.
* Thương lượng giá cả: Trong một số trường hợp, khách hàng có thể thương lượng giá cả với nhân viên cửa hàng để có giá tốt hơn.
* Thanh toán: Sau khi đã chọn sản phẩm và thỏa thuận giá cả, khách hàng sẽ phải thanh toán trực tiếp tại cửa hàng bằng tiền mặt hoặc thẻ thanh toán.
* Nhận sản phẩm: Khách hàng sẽ nhận sản phẩm tại cửa hàng sau khi đã thanh toán và được nhân viên cửa hàng hướng dẫn sử dụng sản phẩm và cách bảo quản.

### Những nhược điểm của hệ thống

* Thời gian: Khách hàng phải tốn thời gian đến cửa hàng, có thể bị mất công đi lại và chờ đợi khi đến cửa hàng vào giờ cao điểm hoặc khi có quá nhiều khách hàng đến đồng thời.
* Không đảm bảo về sản phẩm: Khách hàng không thể kiểm tra tất cả các sản phẩm mà cửa hàng đang có trong kho. Nếu không may mua phải sản phẩm lỗi hoặc không đúng như mình mong đợi, khách hàng phải quay trở lại cửa hàng để đổi hoặc trả sản phẩm, tốn thêm thời gian và công sức.
* Giá cả: Giá cả của sản phẩm tại cửa hàng có thể cao hơn so với giá bán online hoặc ở các cửa hàng khác. Khách hàng phải điều tra giá cả của nhiều cửa hàng để tìm được sản phẩm với giá tốt nhất.
* Không thuận tiện: Có những khách hàng có lịch trình bận rộn và không thể đến cửa hàng trong giờ làm việc hoặc ở xa cửa hàng, đòi hỏi phải tốn chi phí đi lại hoặc chờ đợi khi mua hàng trực tiếp tại cửa hàng.
* Hạn chế về địa điểm: Cửa hàng có giới hạn về không gian và vị trí, điều này giới hạn lượng khách hàng có thể đến mua hàng.
* Hạn chế về thời gian: Cửa hàng chỉ hoạt động trong giờ hành chính, không đáp ứng được nhu cầu mua sắm của khách hàng vào các giờ tối hoặc cuối tuần.
* Chi phí vận hành cửa hàng: Phải trang trải các chi phí như thuê mặt bằng, tiền điện nước, lương nhân viên, chi phí quảng cáo để thu hút khách hàng.
* Không thể tiếp cận được khách hàng ở xa: Khách hàng ở xa không thể đến trực tiếp cửa hàng để mua hàng, điều này giới hạn khả năng bán hàng của cửa hàng.
* Khó khăn trong việc quản lý kho hàng: Cửa hàng phải quản lý và bảo quản hàng hóa một cách chặt chẽ để tránh bị thất thoát, hư hỏng hoặc mất cắp, điều này tốn nhiều thời gian và chi phí.

Mục tiêu và phạm vi đề tài

Mục tiêu để xây dựng website bán đồng hồ giúp người xem nắm rõ những thông tin chính xác về sản phẩm, các thông tin chất lượng, mới nhất về các sản phẩm ra mắt, cũng như những thông tin các sản phẩm sắp xa mắt, ... mà mọi người săn đón.

Nhiệm vụ:

* Tải các công cụ xây dựng website.
* Phân tích thiết kế.
* Đánh giá khả năng của hệ thống website
* Nghiên cứu, phát triển tổng hợp các tiềm năng phù hợp với hệ thống website

Phạm vi đề tài:

* Nghiên cứu các website bán đồng hồ.
* Nghiên cứu quy trình đặt hàng online.
* Nghiên cứu quy trình ship hàng online.

## Định hướng giải pháp

Với việc thiết kế một website bán đồng hồ online, cửa hàng sẽ đem lại nhiều lợi ích cho cả cửa hàng và khách hàng như sau:

Tiết kiệm chi phí và thời gian: Với việc bán hàng online, cửa hàng không cần phải thuê mặt bằng, tuyển dụng nhân viên bán hàng, và trang bị hệ thống POS như khi bán hàng trực tiếp tại cửa hàng. Điều này giúp cửa hàng tiết kiệm được chi phí và thời gian đáng kể.

Mở rộng khách hàng tiềm năng: Với website bán hàng, cửa hàng có thể tiếp cận được đến nhiều khách hàng tiềm năng hơn, không chỉ trong khu vực lân cận mà còn trên toàn quốc. Điều này giúp cửa hàng tăng doanh số bán hàng và thu nhập.

Giúp khách hàng dễ dàng tìm kiếm và mua hàng: Với website bán hàng, khách hàng có thể dễ dàng tìm kiếm và mua hàng mọi lúc mọi nơi chỉ với một vài cú click chuột. Họ không cần phải tốn thời gian và tiền bạc để di chuyển đến cửa hàng.

Tạo sự tin tưởng và minh bạch: Khi cửa hàng sử dụng website bán hàng, khách hàng có thể tiện lợi kiểm tra thông tin về sản phẩm, đánh giá của khách hàng khác, giá cả và các ưu đãi khác một cách nhanh chóng và dễ dàng. Điều này giúp tăng sự tin tưởng và minh bạch trong quá trình mua hàng.

Quản lý hàng hoá và doanh thu dễ dàng: Với website bán hàng, cửa hàng có thể quản lý thông tin hàng hoá, số lượng tồn kho, doanh thu và đánh giá của khách hàng một cách dễ dàng và thuận tiện. Điều này giúp cửa hàng có thể đưa ra quyết định kinh doanh chính xác hơn.

Từ đó em nhận thấy việc thiết kế một website bán đồng hồ online sẽ giúp cửa hàng tăng doanh số bán hàng, tiết kiệm chi phí và thời gian, tăng khả năng tiếp cận khách hàng tiềm năng, tạo sự tin tưởng và minh bạch, đồng thời giúp cửa hàng quản lý hàng hoá và doanh thu dễ dàng.

## Công nghệ sử dụng

### Ngôn ngữ ASP.NET là gì?

ASP.NET là một framework phát triển ứng dụng web được Microsoft giới thiệu vào năm 2002. Nó là một phần của .NET Framework và được xây dựng trên ngôn ngữ lập trình C#, Visual Basic và F#. ASP.NET cho phép phát triển viên tạo các ứng dụng web có hiệu suất cao, an toàn và có thể mở rộng trên nền tảng Windows.

ASP.NET bao gồm các thành phần quan trọng để xây dựng ứng dụng web như Web Forms, MVC (Model-View-Controller), Web API, SignalR, và Razor. Mỗi thành phần này cung cấp các công cụ khác nhau để phát triển ứng dụng web trên nền tảng ASP.NET.

Ngôn ngữ chính được sử dụng trong ASP.NET là C# (C-Sharp) và Visual Basic (VB.NET). C# là ngôn ngữ lập trình đối tượng, được Microsoft phát triển để cải tiến tính bảo mật và hiệu suất của các ứng dụng trên nền tảng .NET. Trong khi đó, VB.NET là một ngôn ngữ lập trình đối tượng, được Microsoft phát triển dựa trên ngôn ngữ Visual Basic trước đây.

ASP.NET hỗ trợ sử dụng nhiều ngôn ngữ lập trình khác nhau, bao gồm C++, F# và IronPython. Bên cạnh đó, nó cũng cho phép phát triển viên sử dụng các công nghệ như HTML, CSS và JavaScript để tạo giao diện và chức năng cho các ứng dụng web.

Ngoài ra, ASP.NET còn hỗ trợ các công nghệ mã nguồn mở như Angular, React, và Vue.js để tạo ứng dụng web đa nền tảng. Với ASP.NET, phát triển viên có thể tạo các ứng dụng web đáp ứng nhu cầu của nhiều thiết bị khác nhau và đảm bảo tương thích trên nhiều nền tảng hệ điều hành.

Tóm lại, ASP.NET là một framework phát triển ứng dụng web mạnh mẽ, đa nền tảng và linh hoạt. Nó cung cấp nhiều công cụ và tính năng cho phát triển viên để tạo ra các ứng dụng web an toàn, hiệu suất cao và có khả năng mở rộng trên nhiều nền tảng.

### Cấu trúc và các thành phần của ASP.NET

ASP.NET là một nền tảng phát triển ứng dụng web của Microsoft. Nó được xây dựng trên framework .NET Framework và hỗ trợ nhiều ngôn ngữ lập trình như C#, VB.NET, F#, v.v.

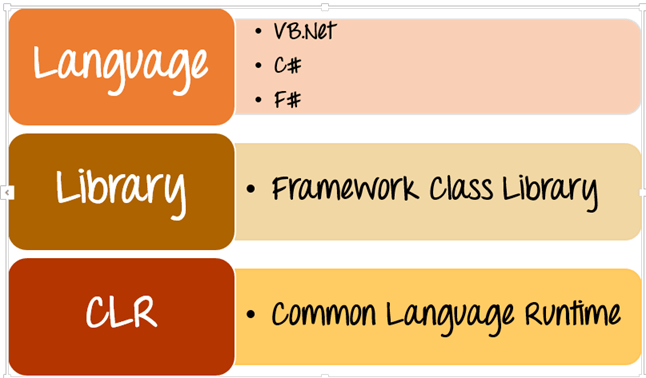
Cấu trúc của ASP.NET bao gồm 3 phần chính: Application Frameworks, Utility Frameworks, và Platform.

#### Các đặc điểm cơ bản của ASP.Net framework bao gồm:

* Code Behind Mode/ Trạng thái code rời: Đây là khái niệm về tách rời thiết kế và mã code. Bằng cách tách rời như vậy, việc duy trì ứng dụng ASP.Net trở nên dễ dàng hơn. Loại file thông dụng của ASP.Net là aspx. Giả sử chúng ta có một trang web có tên là MyPage.aspx, sẽ có một tệp khác có tên là MyPage.aspx.cs biểu thị cho phần mã code của trang. Bởi vậy, Visual Studio mới tạo ra các tập tin riêng biệt cho mỗi trang web, một cho phần thiết kế và một dành cho mã code.
* State Management/Quản lý trạng thái: ASP.Net có các phương tiện để kiểm soát quản lý trạng thái, trong khi HTTP được biết đến là một giao thức "không trạng thái". Lấy một ví dụ về ứng dụng giỏ hàng: khi một user đã chọn lựa xong những gì mình muốn mua và đưa ra đưa ra quyết định mua hàng trên trang web, người đó sẽ nhấn nút gửi. Ứng dụng cần ghi nhớ các mục mà người dùng đã chọn mua. Đây là hành động ghi nhớ trạng thái của một ứng dụng tại một thời điểm trong hiện tại. Vì HTTP là giao thức không trạng thái nên khi user truy cập các web bán hàng, HTTP sẽ không lưu trữ thông tin trên các giỏ hàng. Do đó, sẽ cần thêm một số thao tác coding bổ sung để đảm bảo các giỏ hàng sẽ được chuyển đến trang bán hàng. Việc triển khai như vậy có thể trở nên phức tạp tại một vài thời điểm. Nhưng ASP.Net có thể quản lý trạng thái thay cho bạn nhờ tính năng ghi nhớ giỏ hàng và chuyển các mục đã được chọn mua qua trang bán hàng.
* Caching – Bộ nhớ Cache: ASP.Net cũng có thể thực hiện chức năng của Caching, qua đó cải thiện hiệu suất làm việc cho ứng dụng. Với việc lưu bộ nhớ đệm cache, các trang thường xuyên được người dùng yêu cầu có thể được lưu trữ ở một vị trí tạm thời. Các trang này có thể được truy xuất nhanh hơn và người dùng có thể nhận được các phản hồi tốt hơn. Nhờ vậy, bộ nhớ đệm có thể giúp cải thiện đáng kể hiệu suất của ứng dụng. Về mặt tổng quan, ASP.Net là một ngôn ngữ phát triển được sử dụng để xây dựng các ứng dụng web-based và được cấu trúc để hoạt động tương thích với giao thức HTTP chuẩn.

#### Cấu trúc của .Net framework dựa trên các thành phần cơ bản sau:

* Language/Ngôn ngữ: Có rất nhiều ngôn ngữ lập trình khác nhau tồn tại trong .net framework. Các ngôn ngữ này là VB.net và C #, có thể được sử dụng để phát triển các ứng dụng web.
* Library/Thư viện: .NET Framework gồm có một bộ các lớp library chuẩn. Library phổ biến nhất được sử dụng cho các ứng dụng web trong .net là Web library. Web library bao gồm tất cả các thành phần cần thiết sử dụng trong phát triển các ứng dụng web-based.
* Common Language Runtime/Thời gian chạy ngôn ngữ lập trình thông thường hay CLR: Common Language Infrastructure - Cơ sở hạ tầng ngôn ngữ lập trình phổ thông hay CLI là một nền tảng dùng để chạy các chương trình .Net. Trong đó, CLR sẽ thực hiện các tác vụ chính bao gồm xử lý các trường hợp cá biệt và thu gom rác.



Hình 1 *Cấu trúc .Net*

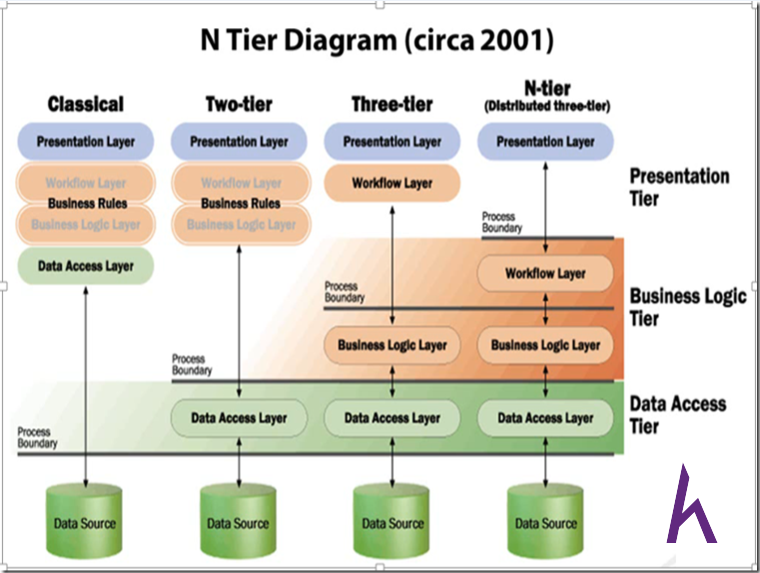
## Thiết kế kiến trúc

### Lựa chọn kiến trúc phần mềm

**Tổng quan mô hình MVC**

Cấu trúc cơ bản của mô hình lập trình web hiện nay

* Minh họa cấu trúc lập trình web:



Hình 2 *Minh họa cấu trúc lập trình web*

Trong lập trình web được chia ra thành 3 tầng như sau:

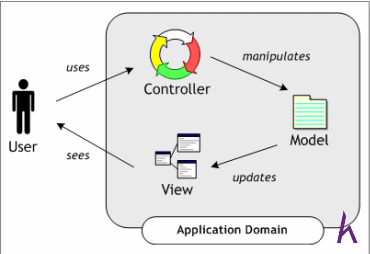
* Presentation Layer (tầng 3): tầng này giúp hiển thị giao diện
* Business Logic Layer (tầng 2): tầng này chứa các hàm thực thi, giúp xử lý sự kiện xảy ra trên tầng Presentation Layer.
* Data Access Layer (tầng 1): tầng này chứa hàm kết nối với các hệ quản trị cở sở dữ liệu như SQL Server, Mysql, Oracle …

Theo như hình mô tả, cấu trúc cổ điển là 3 tầng này trộn vào nhau hay còn gọi là cấu trúc One – Tier. Khi làm việc với cấu trúc One – Tier lập trình viên hay gặp khó khăn trong việc chỉnh sửa code hay chỉnh sửa giao diện vì các tầng này bi trộn vào nhau.

* Tiếp đó, cấu trúc Two – Tier ra đời, tuy nhiên vẫn chưa giải quyết vấn đề của cấu trúc One – Tier gặp phải đó là việc chỉnh sửa code hay giao diện gặp khó khăn vì tầng 2 và tầng 3 vẫn chưa tách biệt khỏi nhau.
* Vì vậy câu trúc Three – Tier đã ra đời và giải quyết tất cả các vần đề mà One – Tier, Two – Tier gặp phải. Lúc này 3 tầng đã tách biệt nhau ra do đó việc chỉnh sửa code hay giao diện sẽ trở nên dễ dàng hơn giúp cho ứng dụng tối ưu hóa nhất

Sau khi hiểu và nhớ lại cấu trúc của một mô hình lập trình web cơ bản thì em sẽ đi qua tìm hiểu về mô hình MVC

* Minh họa mô hình MVC:



Hình 3*Mô hình MVC*

MVC (Model – View - Controller) là một design partern đã tồn tại rất lâu trong ngành công nghệ phần mềm. Một ứng dụng viết theo mô hình MVC sẽ bao gồm 3 thành phần tách biệt nhau đó là Model, View, Controller. Giống như trong cấu trúc Three – Tier, mô hình MVC giúp tách biệt 3 tầng trong mô hình lập trình web, vì vậy giúp tối ưu ứng dụng, dễ dàng thêm mới và chỉnh sửa code hoặc giao diện

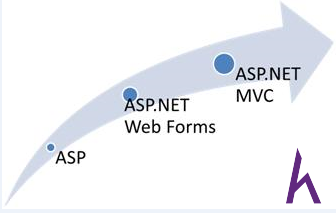
* Model: ở phần trước mình đã nhắc lại cho các bạn về 3 tầng trong mô hình
* Three – Tier thì trong đó gồm có 2 tầng Data Access Layer và tầng Business Logic Layer. Hai tầng này là hai tầng tương đương với tầng model trong mô hình MVC.
* View: là tầng giao diện, hiển thị dữ liệu được truy xuất từ tầng model. Tầng này tương đương với tầng Presentation Layer trong cấu trúc Three – Tier.
* Controller: đây là tầng giúp kết nối giữa tầng model và tầng view trong mô hình MVC, có nghĩa là nếu phía client yêu cầu hiển thị dữ liệu thì controller gọi giữ liệu từ model và trả về cho view vì view tương tác trực tiếp với client

Ví dụ: User yêu cầu hiển thị thông tin cá nhân của user

* User gửi một yêu cầu tới controller.
* Controller nhận yêu cầu, xử lý yêu cầu, nếu yêu cầu cần truy xuất dữ liệu thì controller sẽ gửi yêu đó xuống tầng model để truy xuất dữ liệu
* Tầng model sẽ lấy dữ liệu từ database sau đó truyền dữ liệu qua tầng view thông qua tầng controller để tầng view hiển thị dữ liệu cho User
* User sẽ thấy thông tin hiển thị ở giao diện và cụ thể ở đây là tầng view

Qua ví dụ minh họa thì em đã nắm bắt được cơ chế hoạt động của mô hình MVC rồi! Bây giờ em sẽ đi qua tổng quan về ASP.Net MVC

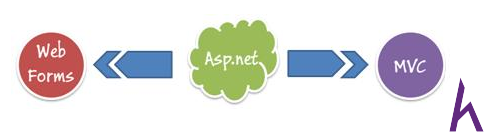
**Tổng quan về mô hình lập trình web ASP.NET MVC**



Hình 4 *Mô hình ASP.Net MVC*

Hình mô tả phía trên cho thấy trước khi **ASP.Net MVC** ra đời thì ASP.Net webform là 1 framework lập trình web khá quen thuộc đối với những lập trình viên của **ASP.Net**. Tuy nhiên, khi **ASP.Net MVC** xuất hiện thì dường như ASP.Net webform bị lãng quên.

**ASP.NET là gì ?**



Hình 5 *ASP.NET là gì?*

**ASP.NET** là một framework phát triển web được phát triển bởi Microsoft, cung cấp các công cụ và thư viện để phát triển các ứng dụng web động, chạy trên nền tảng .NET Framework hoặc .NET Core. ASP.NET cung cấp cho các nhà phát triển nhiều lựa chọn để xây dựng các ứng dụng web như trang web, ứng dụng web, dịch vụ web, và các hệ thống quản lý nội dung trực tuyến (CMS). Với ASP.NET, các nhà phát triển có thể sử dụng các ngôn ngữ lập trình phổ biến như C# hoặc Visual Basic để phát triển ứng dụng web và sử dụng các thư viện .NET để giảm thiểu thời gian và công sức trong quá trình phát triển.

**Ưu điểm khi lập trình với ASP.Net:**

* Hỗ trợ các ngôn ngữ lập trình đa dạng: ASP.NET hỗ trợ nhiều ngôn ngữ lập trình, bao gồm C#, VB.NET, F# và nhiều ngôn ngữ khác.
* Dễ dàng triển khai và quản lý: ASP.NET được thiết kế để dễ dàng triển khai và quản lý. Nó hỗ trợ các công cụ quản lý như IIS và Visual Studio.
* Bảo mật tốt: ASP.NET cung cấp nhiều tính năng bảo mật, bao gồm chứng thực, phân quyền và mã hóa dữ liệu.
* Tích hợp tốt với các công nghệ khác: ASP.NET có thể tích hợp với các công nghệ khác, bao gồm Web services, AJAX, LINQ, và Entity Framework.
* Hiệu suất cao: ASP.NET được thiết kế để tối ưu hóa hiệu suất. Nó cung cấp nhiều tính năng như mã hóa phía máy chủ, kiểm soát ViewState và việc sử dụng bộ nhớ đệm để tăng tốc độ.
* Hỗ trợ phát triển ứng dụng web đa nền tảng: ASP.NET Core là phiên bản mới nhất của ASP.NET, được thiết kế để hỗ trợ phát triển ứng dụng web đa nền tảng trên Windows, macOS và Linux.
* Cộng đồng hỗ trợ lớn: ASP.NET có cộng đồng phát triển lớn, với nhiều tài liệu, thư viện và công cụ hỗ trợ được phát triển và chia sẻ trên mạng.

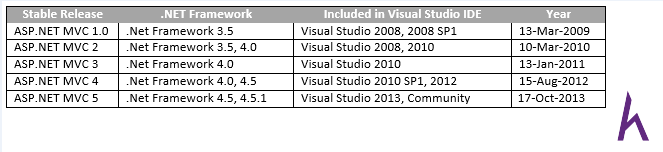
**Nhược điểm khi lập trình với ASP.Net:**

* Học ASP.NET khá khó khăn cho người mới bắt đầu: Vì ASP.NET sử dụng nhiều công nghệ và cần hiểu rõ cấu trúc, nên nó có thể khó khăn đối với người mới học.
* Chi phí cao: ASP.NET là một sản phẩm thương mại, do đó, chi phí để mua giấy phép và phát triển ứng dụng có thể rất đắt.
* Hạn chế đối với các ứng dụng di động: ASP.NET hướng tới việc phát triển các ứng dụng web, do đó, nó có hạn chế đối với các ứng dụng di động.
* Tính tương thích: Một số phiên bản của ASP.NET có thể không tương thích với các phiên bản mới nhất của trình duyệt web, vì vậy việc phải duy trì tính tương thích có thể gây khó khăn cho nhà phát triển.
* Khả năng mở rộng: ASP.NET có thể khó khăn để mở rộng với các giải pháp không phải Microsoft, do đó, việc phát triển các ứng dụng web phải được thực hiện với các công nghệ Microsoft liên quan.

Tuy nhiên, các nhược điểm này cũng đang được cộng đồng phát triển ASP.NET nỗ lực giải quyết và cải thiện liên tục trong các phiên bản mới nhằm tăng tính ổn định, đa dạng và hỗ trợ cho các nhà phát triển trong quá trình phát triển ứng dụng web.

**Cơ chế hoạt động trong lập trình web với ASP.Net**

**ASP.Net MVC**là 1 framework lập trình web mới của microsoft, công nghệ này ứng dụng mô hình MVC vào trong ASP.Net.

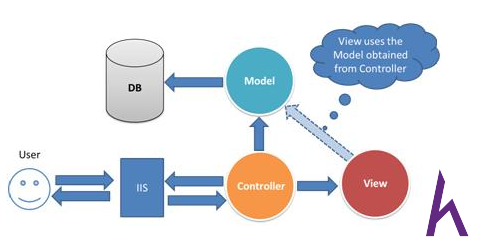


Qua hình mô tả phía trên cho các bạn thấy tuy ASP.Net MVC ra đời năm 2009 nhưng mà tính tới năm 2013 thì nó đã được update lên tới phiên bản **ASP.Net MVC 5**, đây cũng là 1 trong những ưu điểm của ông lớn Microsoft cập nhật công nghệ liên tục.

Và mới đây thì ASP.Net cũng đã cho ra đời 1 công nghệ mới đó là ASP.Net core 1.0

Với sự cập nhật thường xuyên như vậy đã giúp cho ASP.Net MVC ngày nay phát triển mạnh mẽ không thua kém gì các công nghệ lập trình web sử dụng Java và PHP.

**Cơ chế hoạt động của ASP.Net MVC**

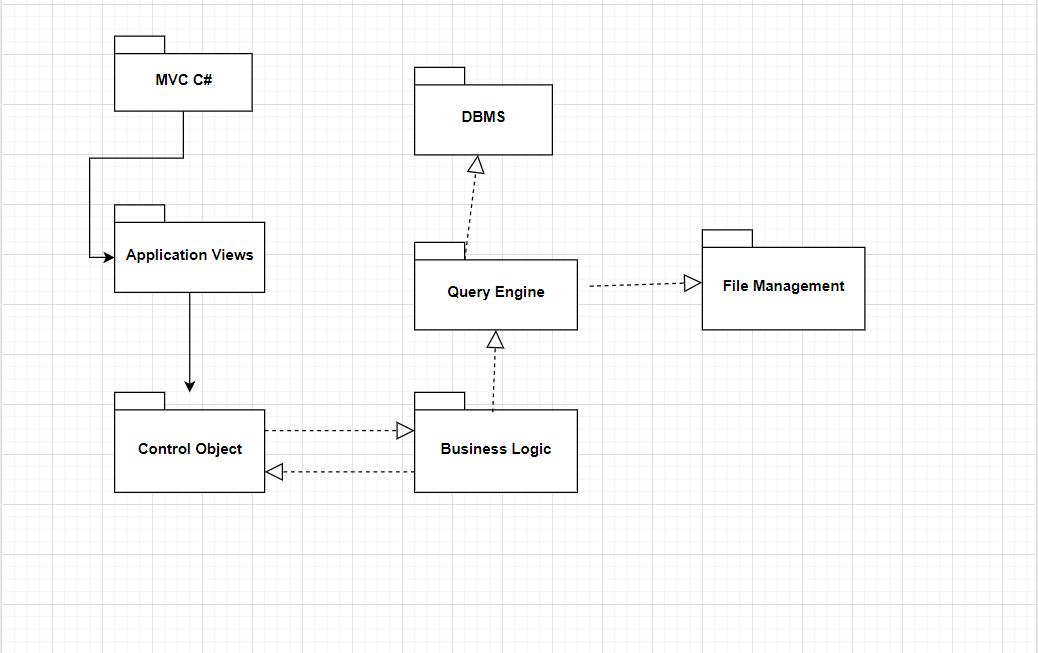


Hình 6 *cơ chế hoạt động của ASP.NET MVC*

Dựa vào hình mô phỏng phía trên này thì mình sẽ khái quá sơ qua cơ chế hoạt động của ASP.Net MVC

* User gửi 1 yêu cầu tới server bằng cách truyền vào 1 URL trong browser
* Yêu cầu đó được gửi tới controller đầu tiên, controller sẽ xử lý yêu cầu, nếu yêu cầu cần truy xuất dữ liệu thì controller sẽ chuyển qua tầng model
* Tại tầng model, dữ liệu được truy xuất từ database và sau đó truyền qua view thông qua controller
* Controller sẽ giúp dữ liệu được chuyển từ model qua view
* View là tầng cuối cùng giao tiếp với User, mọi dữ liệu sẽ được hiển thị cho User thông qua tầng View.

### Thiết kế tổng quan



Hình 7 *Thiết kế tổng quan UML*

Application Views sẽ thể hiện các nội dung muốn trình bày của website

Control Object là các điều khiển thuộc các đối tượng khi chúng ta tác động lên View và website sẽ thực hiện.

Business Logic sẽ thực hiện sau khi Control Object được kích hoạt và thực hiện các thao tác logic để trả lệnh và tạo ra các câu lệnh xuống Query Engine.

Query Engine sẽ lấy các câu lệnh trả xuống từ Business Logic và thực hiện trên DBMS hoặc các File Management.

# PHÂN TÍCH THIẾT KẾ WEBSITE

## Khảo sát hiện trạng

Khảo sát website:

* Hệ thống quảng cáo các website bán đồng hồ, điện thoại, thương mại điện tử.
* Bước tiếp đến cách tiền hành chọn hình ảnh và những thông tin cần thiết nhất của sản phẩm để đưa lên website.

Xây dựng và thiết kế cơ sở đữ liệu:

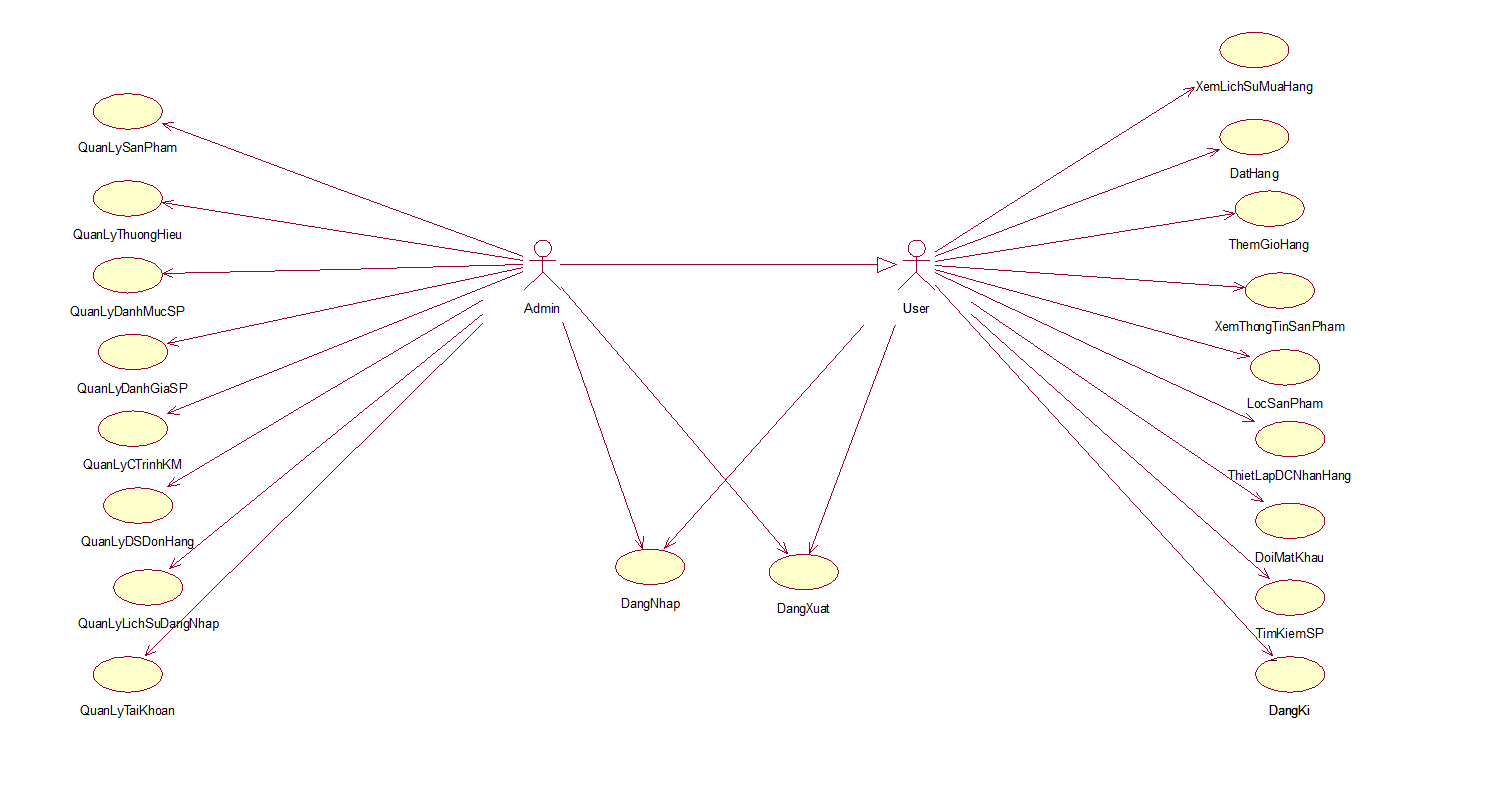
* Cơ sở đữ liệu phải đáp ứng được yêu câu thực tế giúp người nhân viên bán hàng có thể đễ hiểu nhất.
* Khả năng đáp ứng của CSDL này đáp ứng bao nhiêu phân trăm công việc.
* Đưa ra hướng tối ưu cho CSDL sau khi khảo sát thực tế.

## Tổng quan chức năng

Tổng quan hệ thống sẽ có những chức năng như sau:

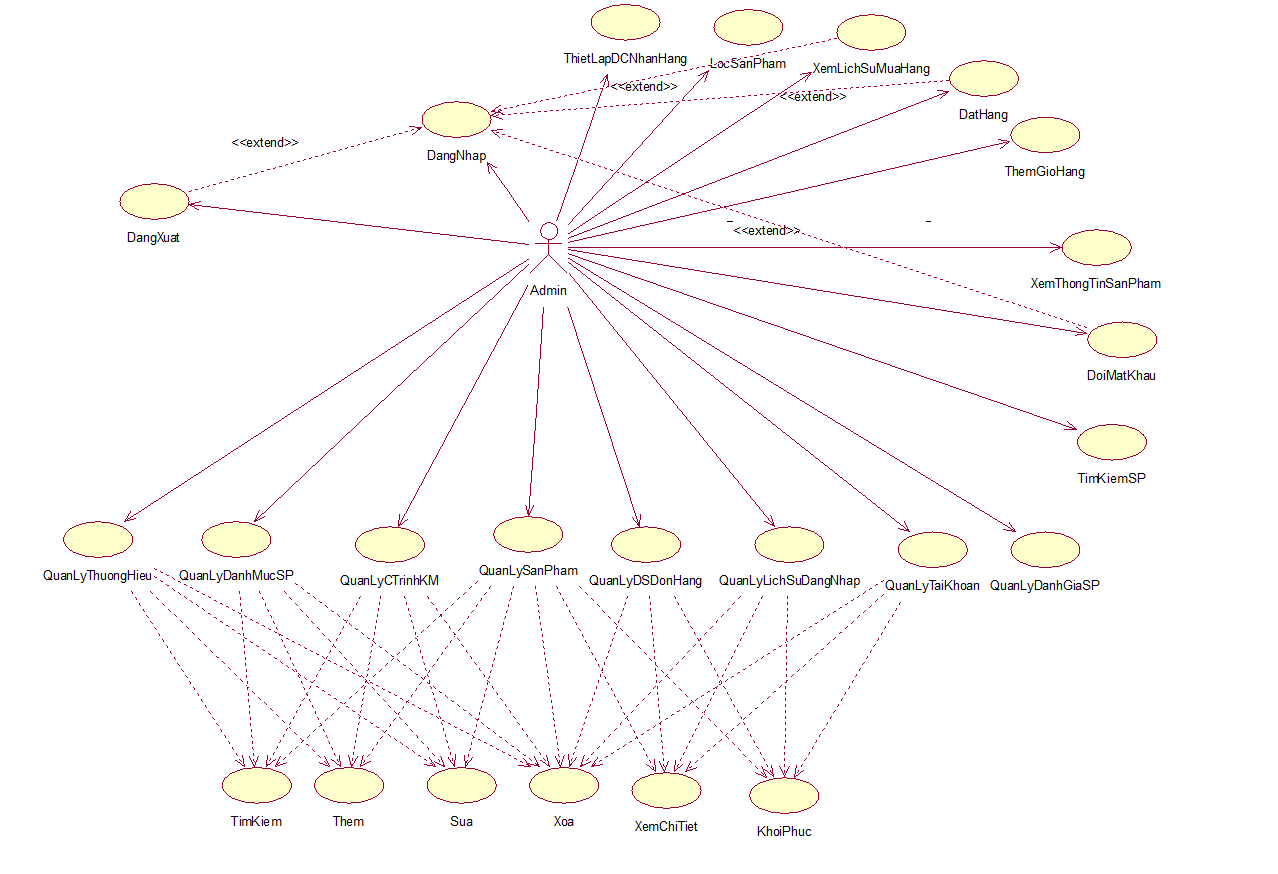
* Về quản trị viên: quản lý sản phẩm, quản lý thương hiệu, quản lý danh mục sản phẩm, quản lý đánh giá sản phẩm, quản lý chương trình giảm giá, quản lý danh sách đơn hàng, quản lý tài khoản, đổi mật khẩu, thiết lập địa chỉ nhận hàng, xem lịch sử mua hàng, thêm giỏ hàng, đặt hàng.
* Về khách hàng: đăng nhập, đăng xuất, xem thông sản phẩm, tìm kiếm, lọc sản phẩm, đổi mật khẩu, thiết lập địa chỉ nhận hàng, xem lịch sử mua hàng, thêm giỏ hàng, đặt hàng.

### Biểu đồ use case tổng quan



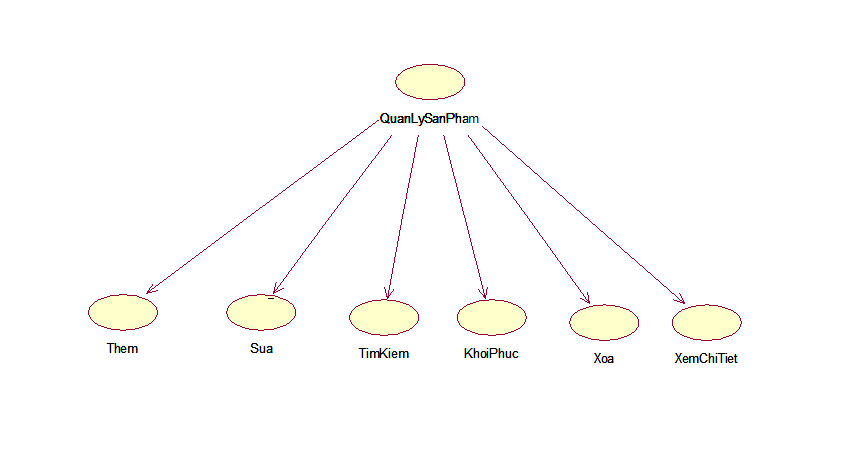
Hình *8 Biểu đồ use case tổng quát*

Biểu đồ use case phân rã Admin tổng quát



Hình 9 *Biểu đồ phân rã Admin tổng quát*

### Biểu đồ phân rã use case quản sản phẩm

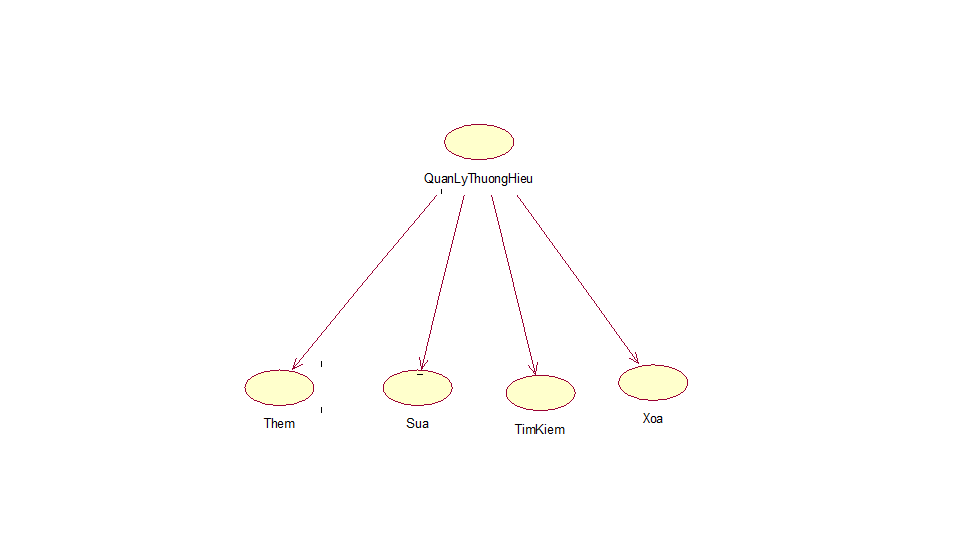


Hình *10 Biểu đồ phân rã Use Case quản lý sản phẩm*

Trong use case quản lý sản phẩm thì quản trị viên sẽ thực hiện các chức năng:

* Thêm mới
* Sửa thông tin
* Xóa
* Khôi phục
* Xem chi tiết

### Biểu đồ phân rã use case Quản lý thương hiệu

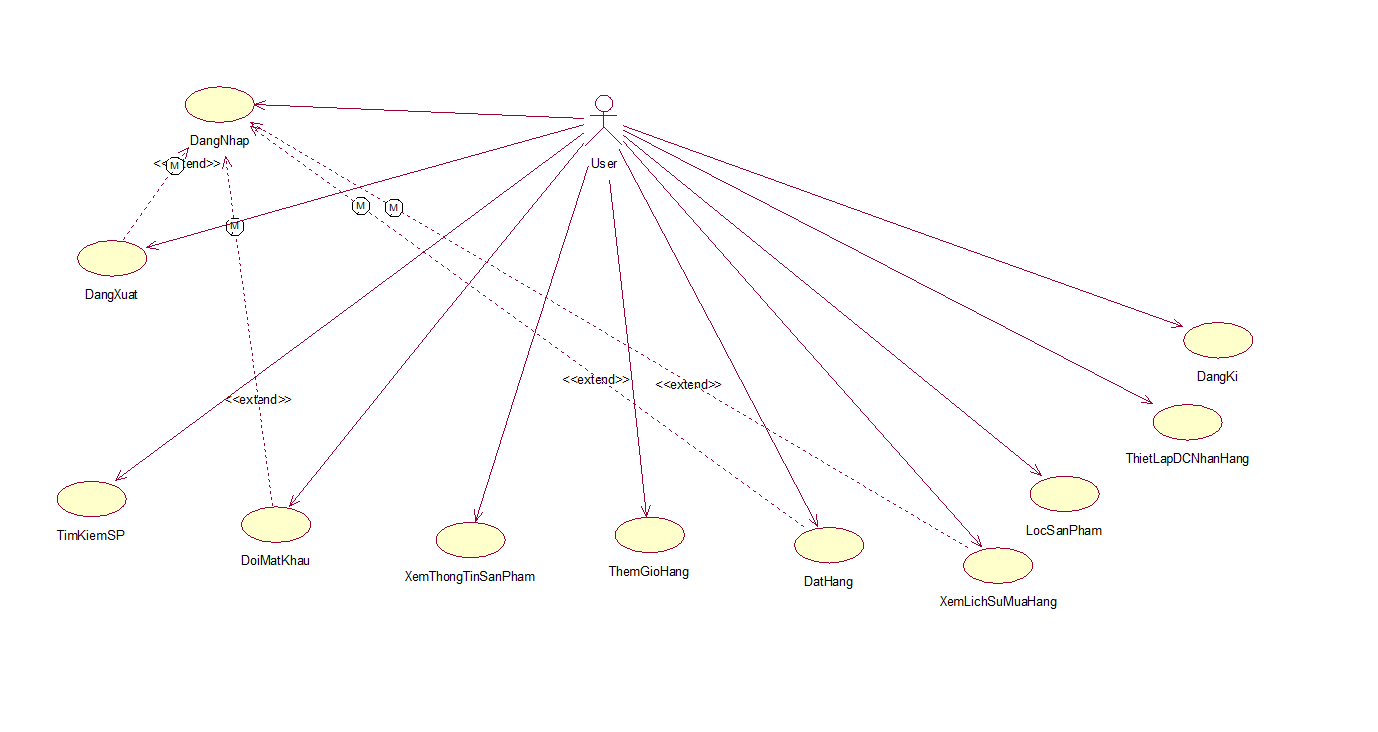


Hình 11 *Biểu đồ phân rã Use Case quản lý thương hiệu*

Trong use case quản lý thương hiệu thì quản trị viên sẽ thực hiện các chức năng:

* Thêm mới.
* Sửa thông tin.
* Xóa.
* Tìm kiếm.

### Biểu đồ phân rã tổng quát User



Hình 12 *Biểu đồ phân rã tổng quát User*

Biểu đồ phân rã tổng quát của user có các use case đăng kí, đăng nhập, đăng xuất, xem thông tin sản phẩm, đổi mật khẩu, tìm kiếm sản phẩm, lọc sản phẩm, thêm giỏ hàng, đặt hàng, xem lịch sử mua hàng, thiết lập địa chỉ nhận hàng.

### Quy trình nghiệp vụ

Quy trình nghiệp vụ của hệ thống:

Một khách hàng khi muốn đặt hàng trên hệ thống thì cần đăng kí tài khoản, sau đó đăng nhập vô hệ thống, chọn sản phẩm và đặt hàng.

## Đặc tả chức năng

### Đặc tả use case đăng kí

Tên use case: Đăng kí.

Mô tả: use case này cho phép khách hàng đăng kí tài khoản trên hệ thống.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi khách hàng nhấn vô nút đăng kí trên màn hình.
* Khách hàng nhập thông tin tài khoản, mật khẩu đầy đủ vào các ô textbox trên màn hình.
* Nhấn nút đăng kí hệ thống sẽ kiểm tra và đăng kí tài khoản, use case kết thúc.
* Luồng rẽ nhánh:
* Use case sẽ bị dừng khi tài khoản đăng kí bị trùng tên đăng nhập.

Các yêu cầu đặc biệt:

* Mật khẩu bao gồm chữ số, chữ cái, bắt đầu bằng chữ in hoa, có kí tự đặc biệt.

Tiền điều kiện: không.

Hậu điều kiện: không.

Điểm mở rộng: không.

### Đặc tả use case đặt hàng

Tên use case: đặt hàng.

Mô tả: use case này cho phép khách hàng đặt hàng online trên hệ thống mà không cần ra tận cửa hàng để mua.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi khách hàng nhấn vào nút thêm vào giỏ hàng trên ảnh sản phẩm hoặc trong thông tin chi tiết sản phẩm.
* Hệ thống lấy thông tin khách hàng và hiển thị lên màn hình, thông tin đơn hàng.
* Khách hàng xác nhận thông tin nhận hàng và nhấn nút đặt hàng ngay.
* Use case kết thúc.
* Luồng rẽ nhánh: không có

Tiền điều kiện:

* Khách hàng cần đăng nhập.

Hậu điều kiện: không.

Điểm mở rộng: không.

### Đặc tả use case xem thông tin sản phẩm

Tên use case: xem thông tin sản phẩm.

Mô tả: Use case này cho phép khách hàng có thể xem thông tin sản phẩm.

Luồng sự kiện:

* Luồng sự kiện chính:
  + - * Use case này bắt đầu khi người dùng kích chọn vào sản phẩm trên màn hình.
      * Hệ thống lấy thông tin sản phẩm trong CSDL và hiển thị lên màn hình.
      * Hệ thống truy vấn bảng product trong CSDL và hiển thị thông tin chi tiết lên màn hình.

Use case kết thúc.

* Luồng rẽ nhánh: không.

Tiền điều kiện:

* Không có

Hậu điều kiện: không.

Điểm mở rộng: không.

### Đặc tả use case quản lý thương hiệu

Tên use case: quản lý thương hiệu

Mô tả: use case này cho phép quản trị viên quản lý thông tin thương hiệu, thêm, sửa, xóa thông tin thương hiệu.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi quản trị viên nhấn chọn quản lý thương hiệu.
* Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút nút sửa, xóa thương hiệu.
* Quản trị viên thực hiện chức năng thêm, khi nhấm nút thêm hệ thống sẽ tải trang thêm thương hiệu để quản trị viên nhập các thông tin đầy đủ của thương hiệu. Sau khi nhập đầy đủ thông tin xong nhấn nút lưu lại và việc thêm thương hiệu đã thực hiện xong.
* Quản trị viên nhấn nút sửa của thương hiệu muốn sửa thông tin, hệ thống sẽ lấy thông tin của thương hiệu đã chọn và hiển thị lên màn hình. Quản trị viên sẽ nhập các thông tin cần thay đổi sau đó nhấn nút cập nhật và quá trình sửa thông tin thương hiệu đã hoàn tất.
* Quản trị viên nhấn nút xóa trên thương hiệu muốn xóa, sau khi nhấn nút xóa hệ thống sẽ lấy thông tin của thương hiệu đã chọn và xóa khỏi cơ sở dữ liệu.
* Luồng rẽ nhánh: không có

Tiền điều kiện: không.

Hậu điều kiện: không.

Điểm mở rộng; không.

### Đặc tả use case quản lý sản phẩm

Tên use case: quản lý sản phẩn

Mô tả: use case này cho phép quản trị viên quản lý thông tin sản phẩm, thêm, sửa, xóa thông tin sản phẩm.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi quản trị viên nhấn chọn quản lý sản phẩm.
* Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút nút sửa, xóa sản phẩm.
* Quản trị viên thực hiện chức năng thêm, khi nhấm nút thêm hệ thống sẽ tải trang thêm sản phẩn để quản trị viên nhập các thông tin đầy đủ của sản phẩm. Sau khi nhập đầy đủ thông tin xong nhấn nút lưu lại và việc thêm sản phẩm đã thực hiện xong.
* Quản trị viên nhấn nút sửa của sản phẩm muốn sửa thông tin, hệ thống sẽ lấy thông tin của sản phẩm đã chọn và hiển thị lên màn hình. Quản trị viên sẽ nhập các thông tin cần thay đổi sau đó nhấn nút cập nhật và quá trình sửa thông tin sản phẩm đã hoàn tất.
* Quản trị viên nhấn nút xóa trên sản phẩm muốn xóa, sau khi nhấn nút xóa hệ thống sẽ lấy thông tin của sản phẩm đã chọn và xóa khỏi cơ sở dữ liệu.
* Quản trị viên nhấn chọn thùng rác để hiển thị những sản phẩm bị xóa. Sau khi danh sách hiển thị lên thì quản trị viên chọn xóa nếu muốn xóa bỏ hoàn toàn sản phẩm hoặc nhấn khôi phục để đưa sản phẩm về lại danh sách bán hàng.
* Luồng rẽ nhánh: không có

Tiền điều kiện: không.

Hậu điều kiện: không.

Điểm mở rộng; không.

### Đặc tả use case quản lý danh mục

Tên use case: quản lý danh mục

Mô tả: use case này cho phép quản trị viên quản lý thông tin danh mục, thêm, sửa, xóa thông tin danh mục.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi quản trị viên nhấn chọn quản lý danh mục.
* Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút danh sách danh mục, nút sửa, xóa danh mục.
* Quản trị viên thực hiện chức năng thêm, khi nhấm nút thêm hệ thống sẽ tải trang thêm danh mục để quản trị viên nhập các thông tin đầy đủ của danh mục. Sau khi nhập đầy đủ thông tin xong nhấn nút lưu lại và việc thêm danh mục đã thực hiện xong.
* Quản trị viên nhấn nút sửa của danh mục muốn sửa thông tin, hệ thống sẽ lấy thông tin của danh mục đã chọn và hiển thị lên màn hình. Quản trị viên sẽ nhập các thông tin cần thay đổi sau đó nhấn nút cập nhật và quá trình sửa thông tin danh mục đã hoàn tất.
* Quản trị viên nhấn nút xóa trên danh mục muốn xóa, sau khi nhấn nút xóa hệ thống sẽ lấy thông tin của danh mục đã chọn và xóa khỏi cơ sở dữ liệu.
* Luồng rẽ nhánh: không có.

Tiền điều kiện: không.

Hậu điều kiện: không.

Điểm mở rộng; không.

### Đặc tả use case quản lý chương trình khuyến mại

Tên use case: quản lý chương trình khuyến mại

Mô tả: use case này cho phép quản trị viên quản lý thông tin chương trình khuyến mại, thêm, sửa, xóa thông tin chương trình khuyến mại.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi quản trị viên nhấn chọn quản lý chương trình khuyến mại.
* Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút danh sách ghế chương trình khuyến mại, nút sửa, xóa chương trình khuyến mại.
* Quản trị viên thực hiện chức năng thêm, khi nhấm nút thêm hệ thống sẽ tải trang thêm chương trình khuyến mại để quản trị viên nhập các thông tin đầy đủ của chương trình khuyến mại. Sau khi nhập đầy đủ thông tin xong nhấn nút lưu lại và việc thêm chương trình khuyến mại đã thực hiện xong.
* Quản trị viên nhấn nút sửa của chương trình khuyến mại muốn sửa thông tin, hệ thống sẽ lấy thông tin của chương trình khuyến mại đã chọn và hiển thị lên màn hình. Quản trị viên sẽ nhập các thông tin cần thay đổi sau đó nhấn nút cập nhật và quá trình sửa thông tin chương trình khuyến mại đã hoàn tất.
* Quản trị viên nhấn nút xóa trên chương trình khuyến mại muốn xóa, sau khi nhấn nút xóa hệ thống sẽ lấy thông tin của chương trình khuyến mại đã chọn và xóa khỏi cơ sở dữ liệu.
* Luồng rẽ nhánh: không có.

Tiền điều kiện: không.

Hậu điều kiện: không.

Điểm mở rộng; không.

### Đặc tả use case quản lý tài khoản

Tên use case: quản lý chương trình khuyến mại

Mô tả: use case này cho phép quản trị viên quản lý thông tin tài khoản của hệ thống.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi quản trị viên nhấn chọn tài khoản trong quản lý tài khoản.
* Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút danh sách tài khoản.
* Quản trị viên nhấn nút sửa quyền cho tài khoản, khi nhấn chọn vô select phân quyền sẽ chuyển quyền cho tài khoản 1 trong 3 quyền: quản trị viên, nhân viên, khách hàng.
* Quản trị viên nhấn nút xóa trong mục hành động, và xóa được tài khoản đã chọn.
* Luồng rẽ nhánh: không có.

Tiền điều kiện: không.

Hậu điều kiện: không.

Điểm mở rộng; không.

### Đặc tả use case thêm vào giỏ hàng

Tên use case: thêm vào giỏ hàng.

Mô tả: use case này cho phép người dùng thêm sản phẩm vào trong giỏ hàng.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi người dùng nhấn chọn vào nút thêm vào giỏ hàng của sản phẩm.
* Hệ thống sẽ lấy dữ liệu của sản phẩm và đưa vào giỏ hàng.
* Hệ thống thông báo thêm vào giỏ hàng thành công và use case kết thúc.
* Luồng rẽ nhánh: không có.

Tiền điều kiện: không.

Hậu điều kiện: không.

Điểm mở rộng; không.

### Đặc tả use case tìm kiếm

Tên use case: tìm kiếm.

Mô tả vắn tắt: Use case này cho phép người đọc tìm kiếm trong website.

Luồng sự kiện:

* Luồng sự kiện chính:
* Use case này bắt đầu khi người dùng đánh từ khóa tìm kiếm vào trong ô tìm kiếm sau đó nhấn tìm kiếm
* Hệ thống sẽ lấy dữ liệu của sản phẩm có thông tin gần đùng hoặc đúng với từ khóa ra.
* Hệ thống hiển thị những sản phẩm vừa được lấy ở trên ra màn hình.
* Use case kết thúc.
* Luồng rẽ nhánh: không có.

Tiền điều kiện: không.

Hậu điều kiện: không.

Điểm mở rộng; không.

## Yêu cầu phi chức năng

Chức năng thanh toán: Cần có nhiều cổng thanh toán như: momo, vnpay, zalo pay vv…

Chức năng chăm sóc khách hàng: tạo chat box hoặc liên kết tài khoản facebook hoặc zalo để tư vấn cho khách hàng.

## Thiết kế chi tiết

### Thiết kế giao diện

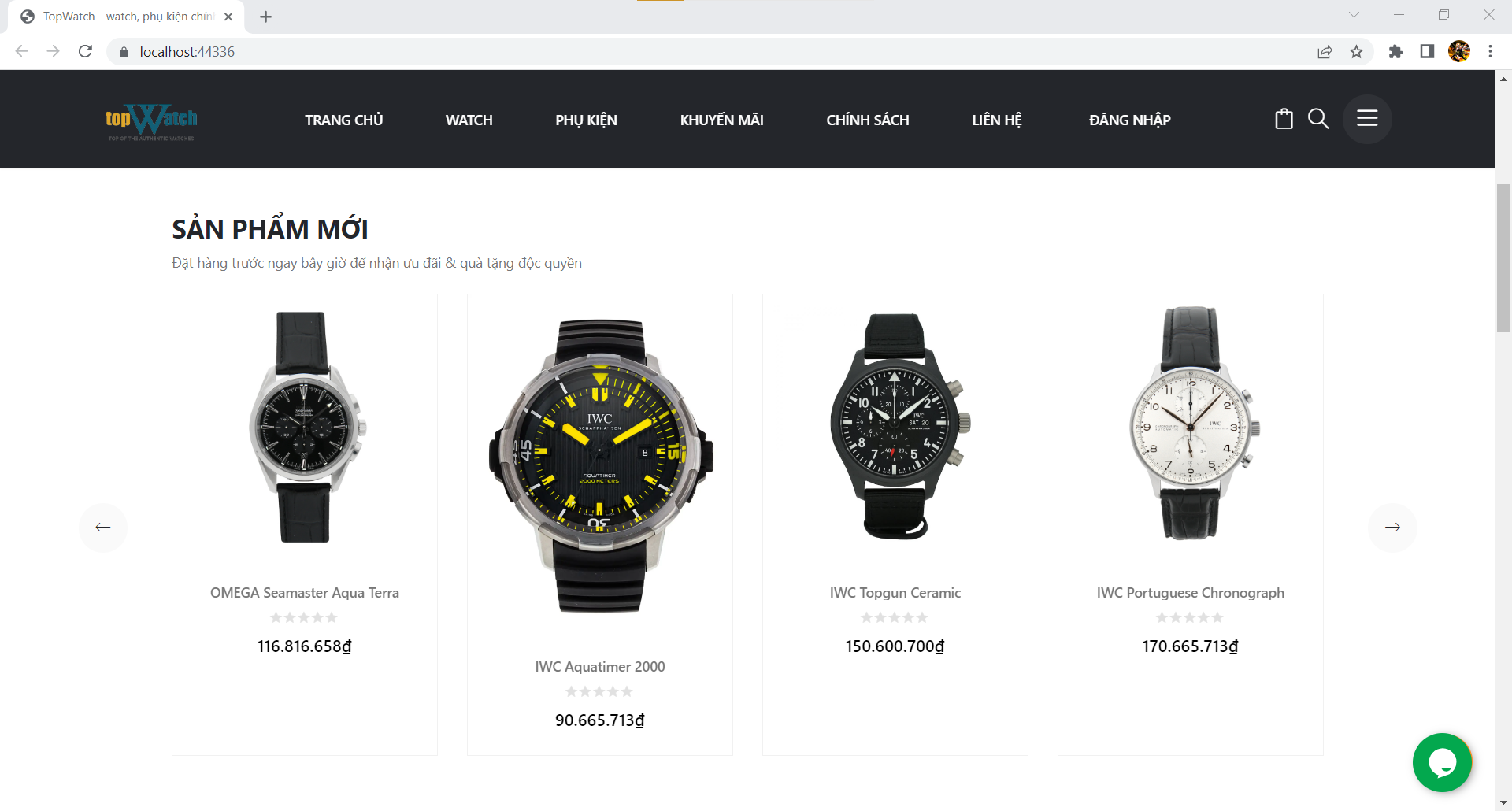
Đặc tả màn hình:

* Độ phân giải: Full HD
* Kích thước màn hình: tùy theo thiết bị của khách hàng sử dụng (đã sử dụng responsive)

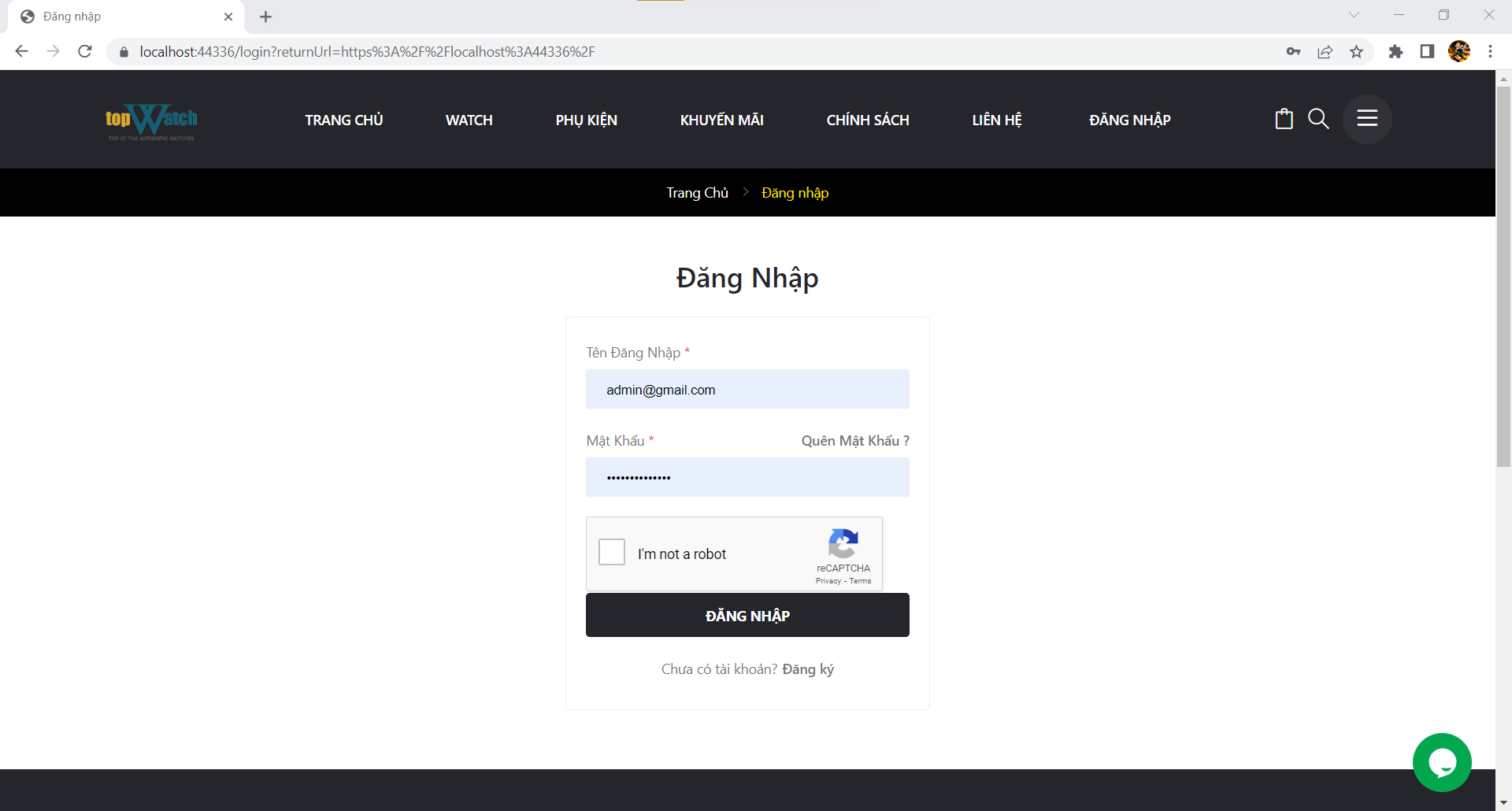
Giao diện màn hình bên khách hàng sử dụng có tông màu chủ đạo là trắng đỏ.

Font chữ: 'Dosis', sans-serif.

Các ô chọn là ô select – option



Hình 13 *Trang danh sách sản phẩm*



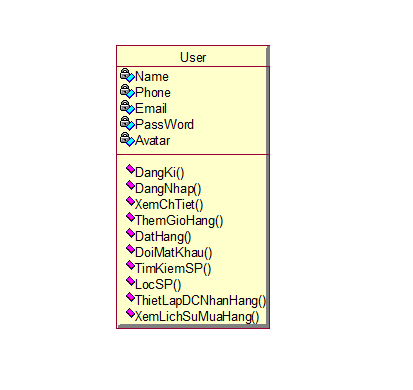
Hình 14 *Đăng nhập*

### Thiết kế lớp

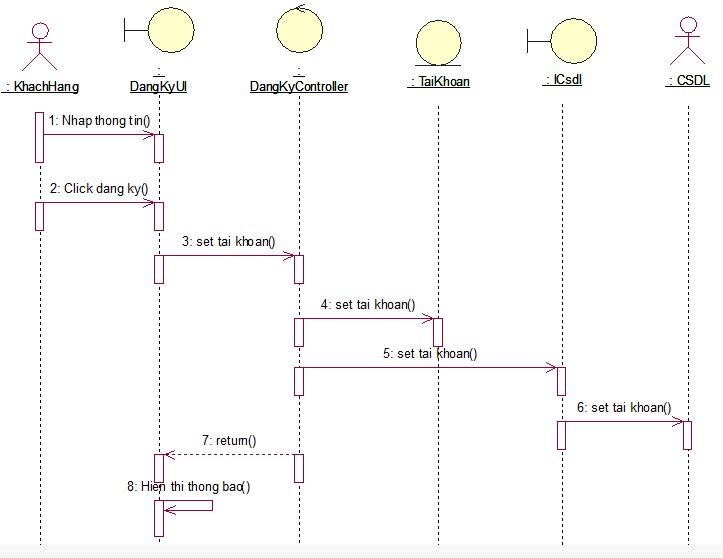
#### Lớp User (khách hàng)

Lớp khách hàng bao gồm các thuộc tính: tên người dùng, email, số điện thoại, mật khẩu, avatar.

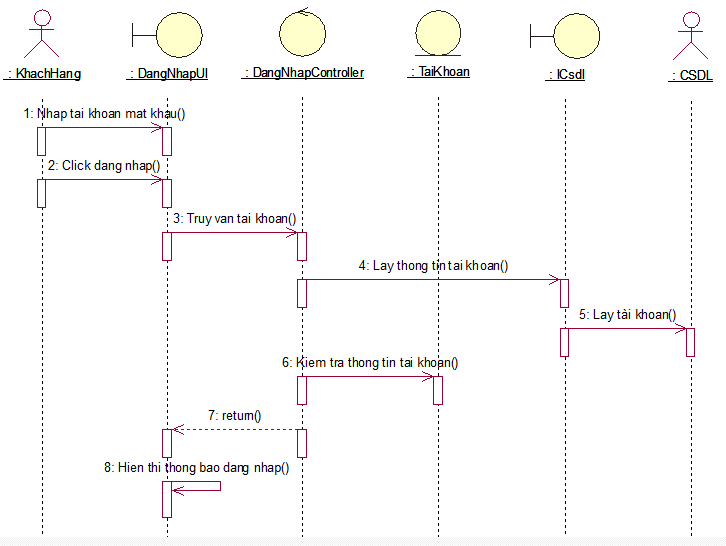
Lớp khách hàng bao gồm các hoạt động: đăng kí, đăng nhập, đăng xuất, xem thông tin sản phẩm, đổi mật khẩu, tìm kiếm sản phẩm, lọc sản phẩm, thêm giỏ hàng, đặt hàng, xem lịch sử mua hàng, thiết lập địa chỉ nhận hàng.



Hình 15 *Lớp User*



Hình 16 *Mô hình tuần tự use case đăng kí*

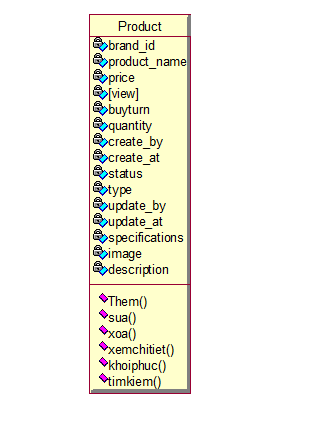


Hình 17  *Mô hình tuần tự use case đăng nhập*

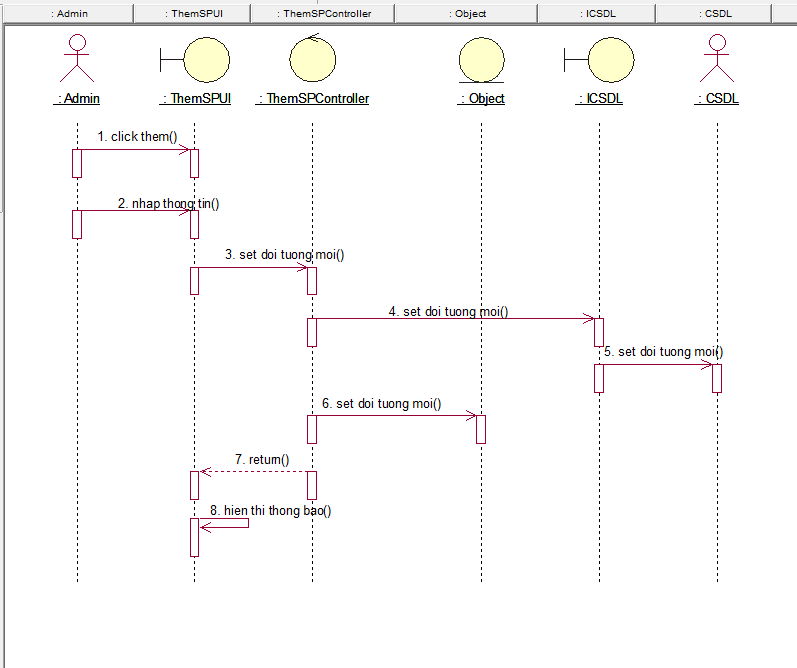
#### Lớp sản phẩm

Lớp sản phẩm có các thuộc tính: brand\_id, product\_name, price, [view], buyturn, quantity, status, create\_by, create\_at, Type, update\_by, update\_at, specifications**,** image, description.

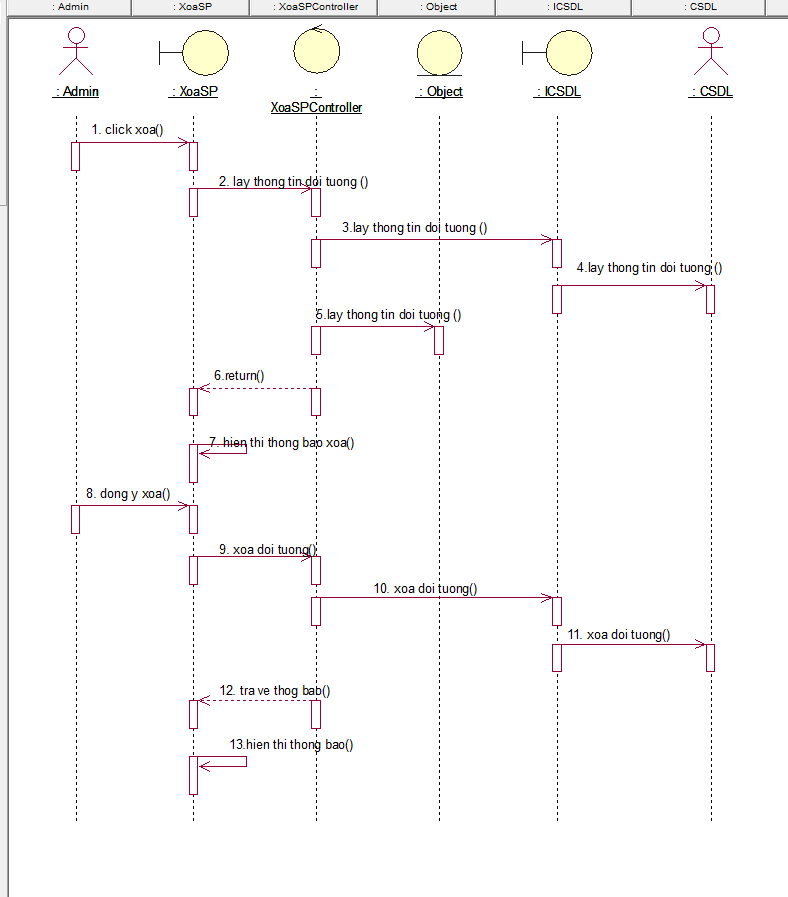
Lớp sản phẩm thực hiện các hoạt động: thêm, sửa, xóa, xem chi tiết, khôi phục.



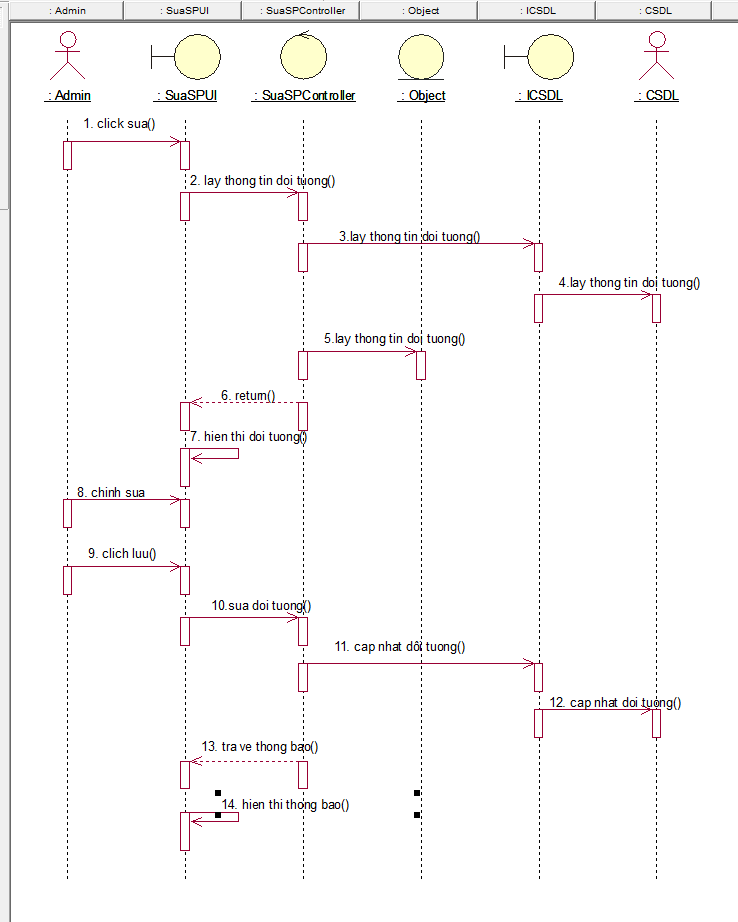
Hình 18 *Lớp sản phẩm*



Hình 19 *Mô hình tuần tự use case thêm sản phẩm*



Hình 20 *Mô hình tuần tự use case xóa sản phẩm*

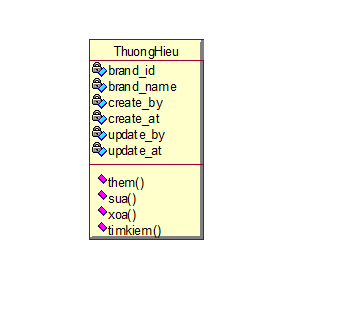


Hình 21 *Mô hình tuần tự sửa sản phẩm*

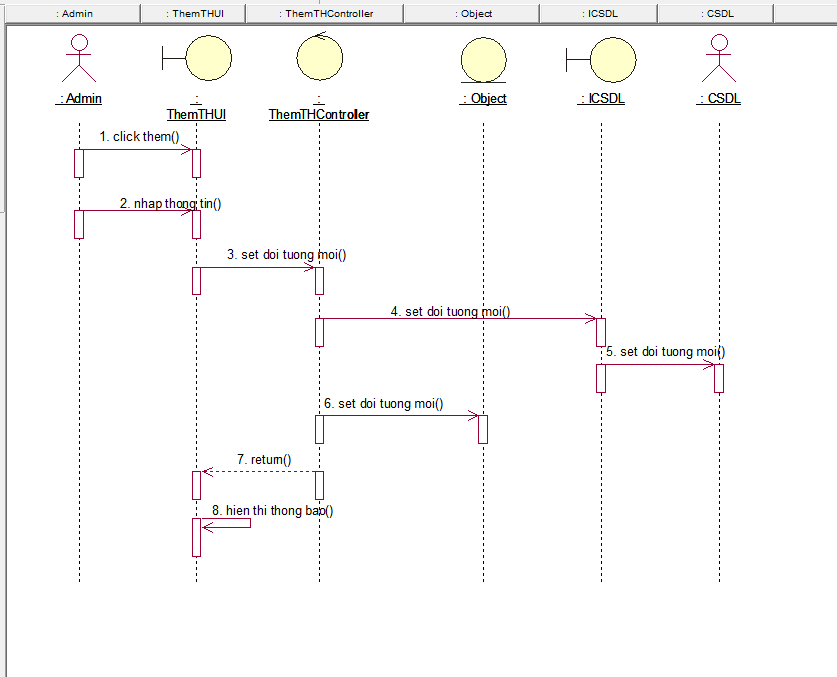
#### Lớp Thương hiệu

Lớp thương hiệu có các thuộc tính: brand\_id, brand\_name, create\_by, create\_at, update\_by, update\_at.

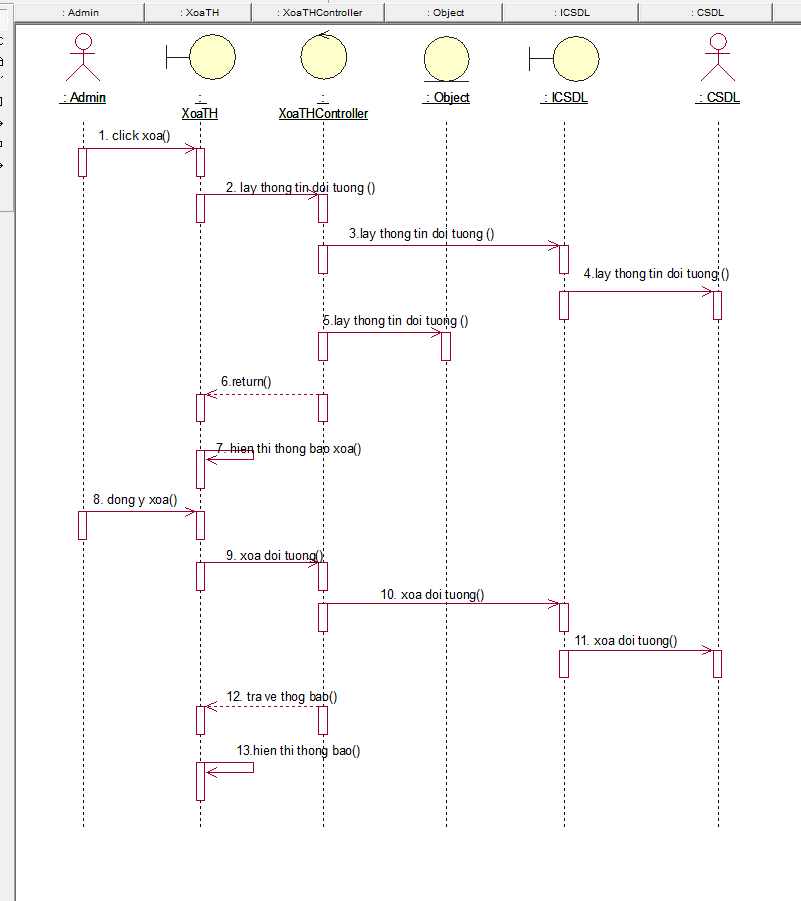
Lớp thương hiệu có các hoạt động: thêm, sửa, xóa, tìm kiếm



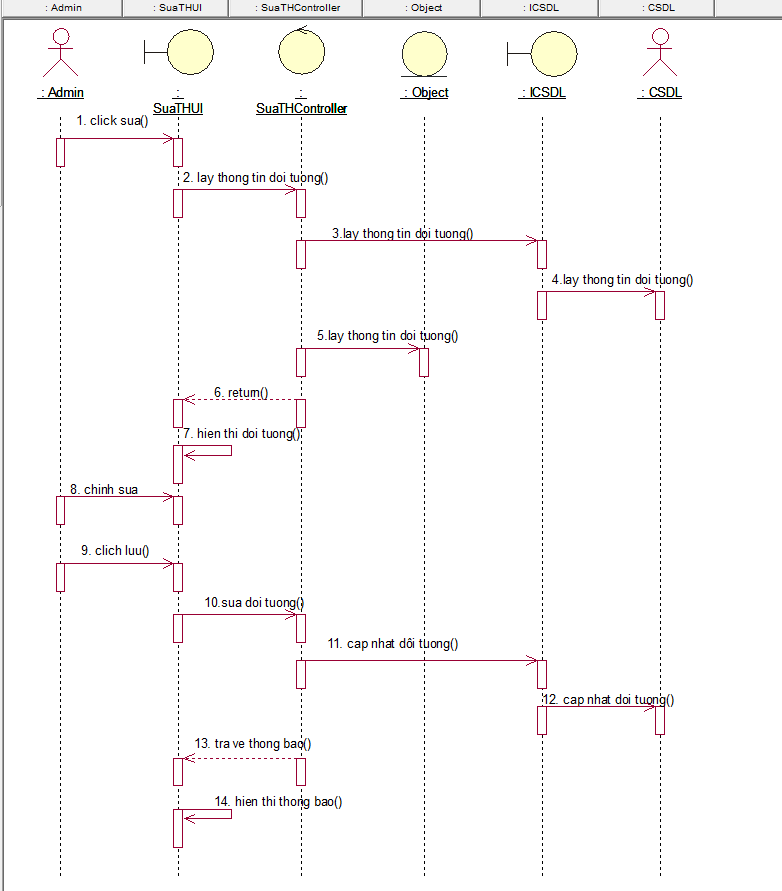
Hình 22 *Lớp thương hiệu*



Hình 23 *Mô hình tuần tự use case thêm thương hiệu*

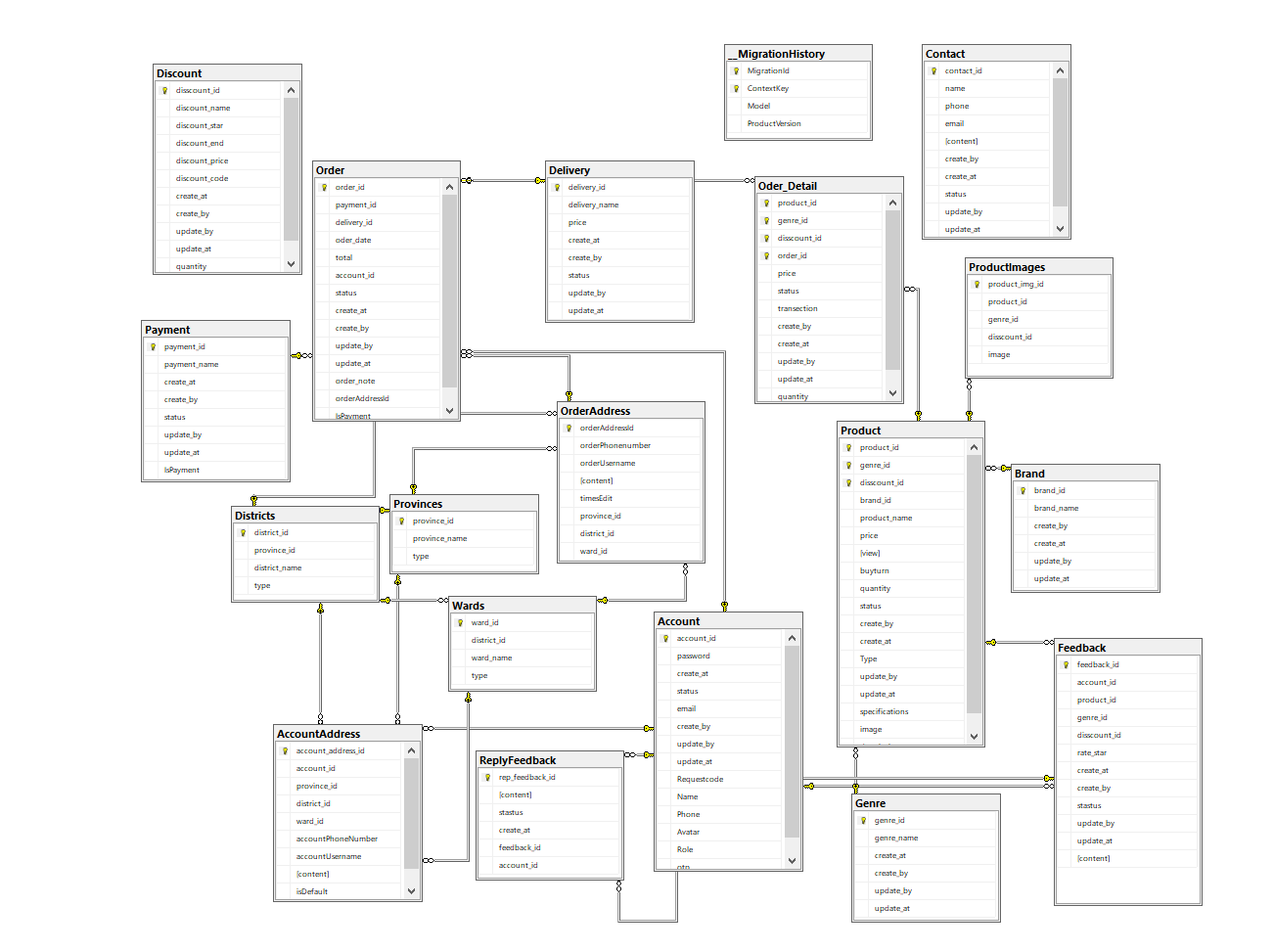


Hình 24 *Mô hình tuần tự xóa thương hiệu*



Hình 25 *Mô hình tuần tự sửa thông tin thương hiệu*

### Thiết kế cơ sở dữ liệu



Hình 26 *Biểu đồ diagram*

# CÀI ĐẶT VÀ KIỂM THỬ

## Xây dựng ứng dụng

### Thư viện và công cụ sử dụng

Bảng 1 Danh sách thư viện và công cụ sử dụng

|  |  |  |
| --- | --- | --- |
| **Mục đích** | **Công cụ** | **Địa chỉ URL** |
| MVC C# | Visual Studio 2022 | https://visualstudio.microsoft.com/vs |
| SQLSever 2019 | SQLServer  Management Studio | https://www.microsoft.com/en-us/sql-server/sql-server-downloads |
| BootStrap | BootStrap | https://getbootstrap.com/docs/5.1/getting-started/introduction/ |
| JQuery | JQuery | https://jquery.com/ |
| Icon V5 | Icon V5 | https://getbootstrap.com/docs/5.0/extend/icons/ |
| IIS | IIS | https://www.iis.net/ |
| LinQ | LinQ | https://learn.microsoft.com/en-us/dotnet/api/system.data.linq?view=netframework-4.8.1 |
| Entity Framework  SqlSever | Entity  Framework  SqlSever | https://www.learnentityframeworkcore.com/walkthroughs/existing-database |

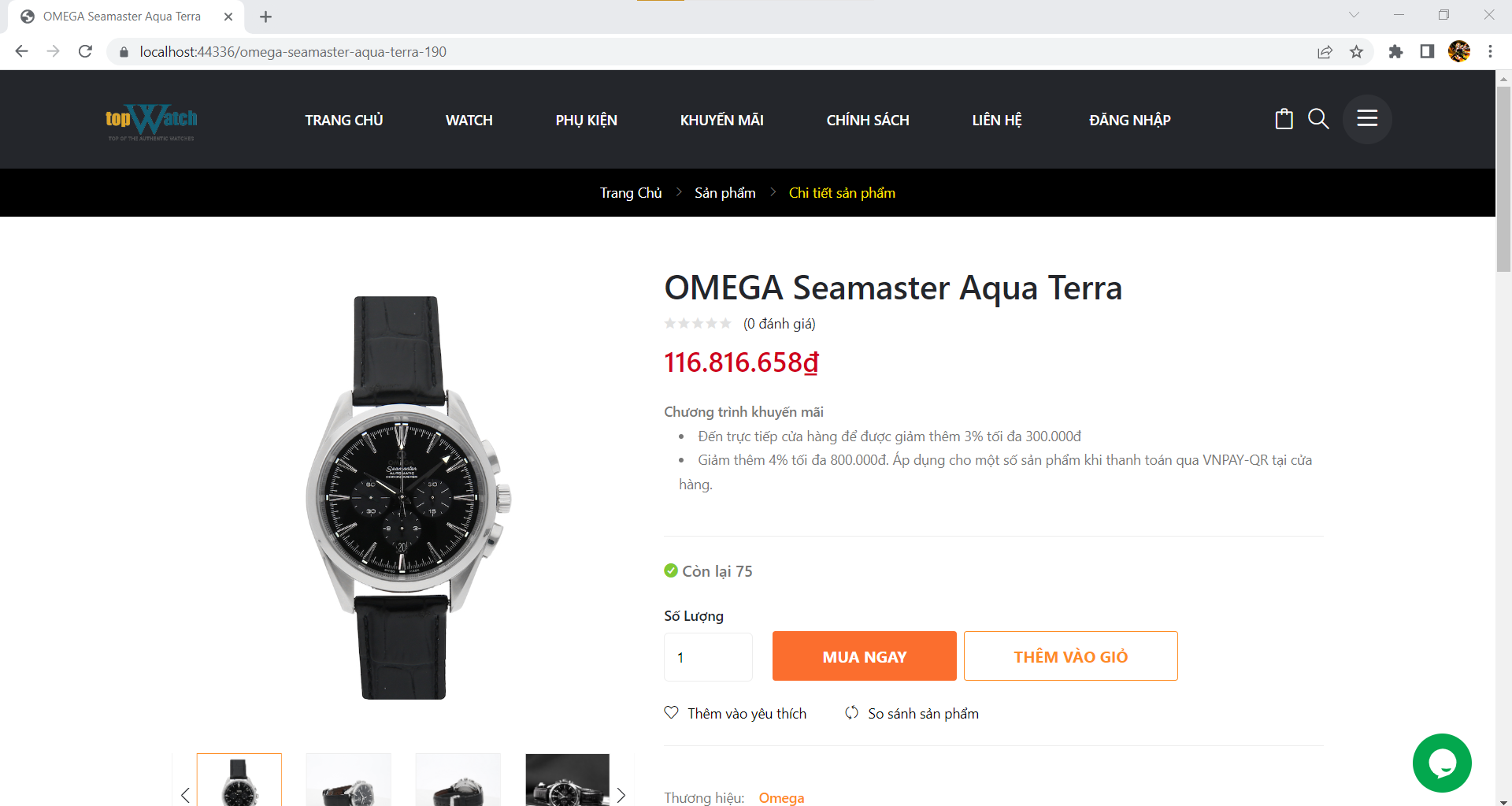
### Kết quả đạt được

Thống kê thông tin về ứng dụng:

Bảng 2 Thống kê thông tin về ứng dụng

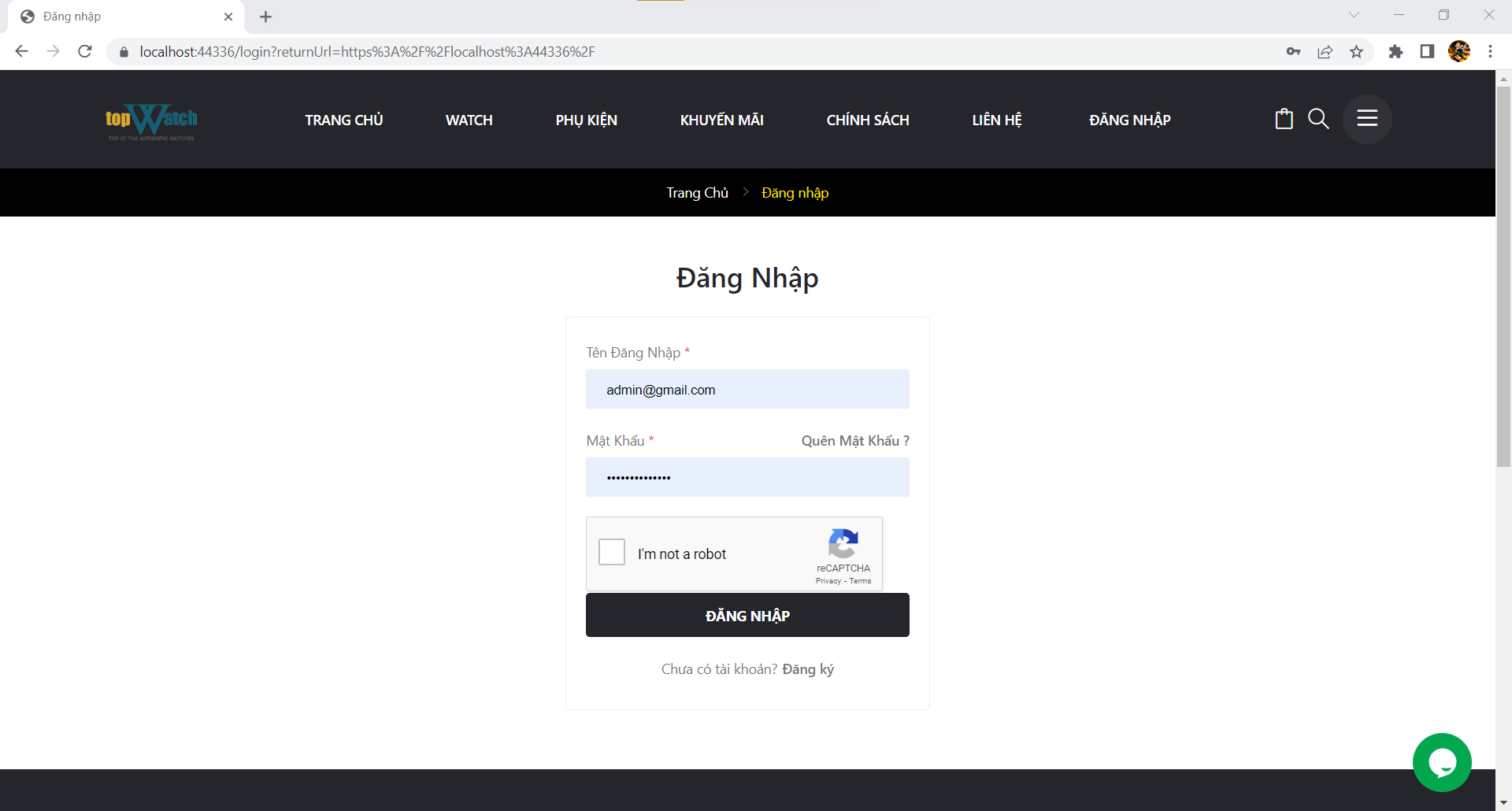
|  |  |
| --- | --- |
| Số dòng code | 4112 |
| Số lớp | 3 |
| Số gói | 7 |
| Dung lượng bộ mã nguồn | 392 MB |

### Minh hoạ các chức năng



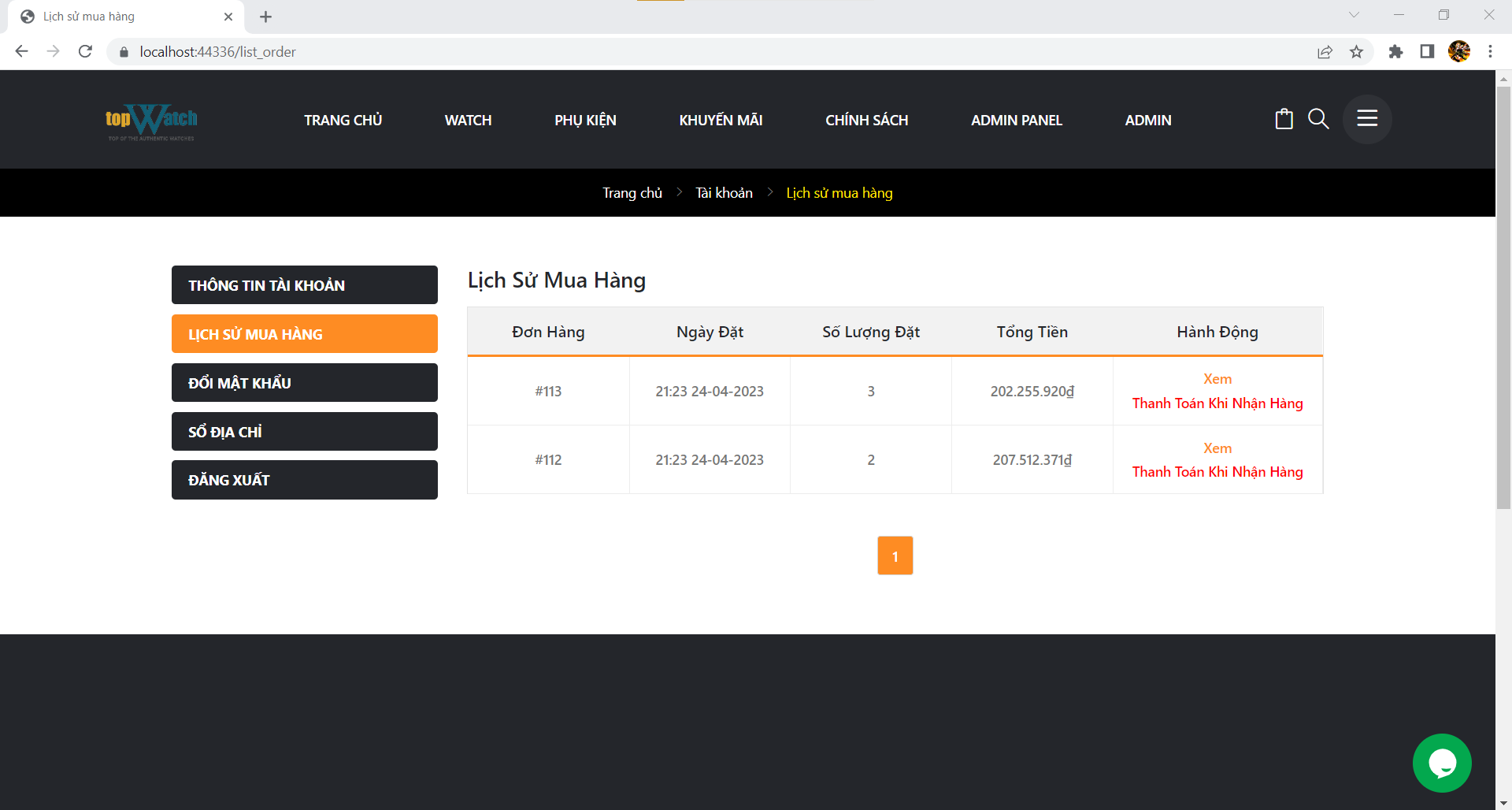
Hình 27 *Xem thông tin sản phẩm*

Chức năng xem thông tin sản phẩm. Chức năng này cho phép khách hàng xem thông tin về thông số của sản phẩm, từ nhà cung cấp thông tin đồng hồ, vật liệu đến giá tiền.



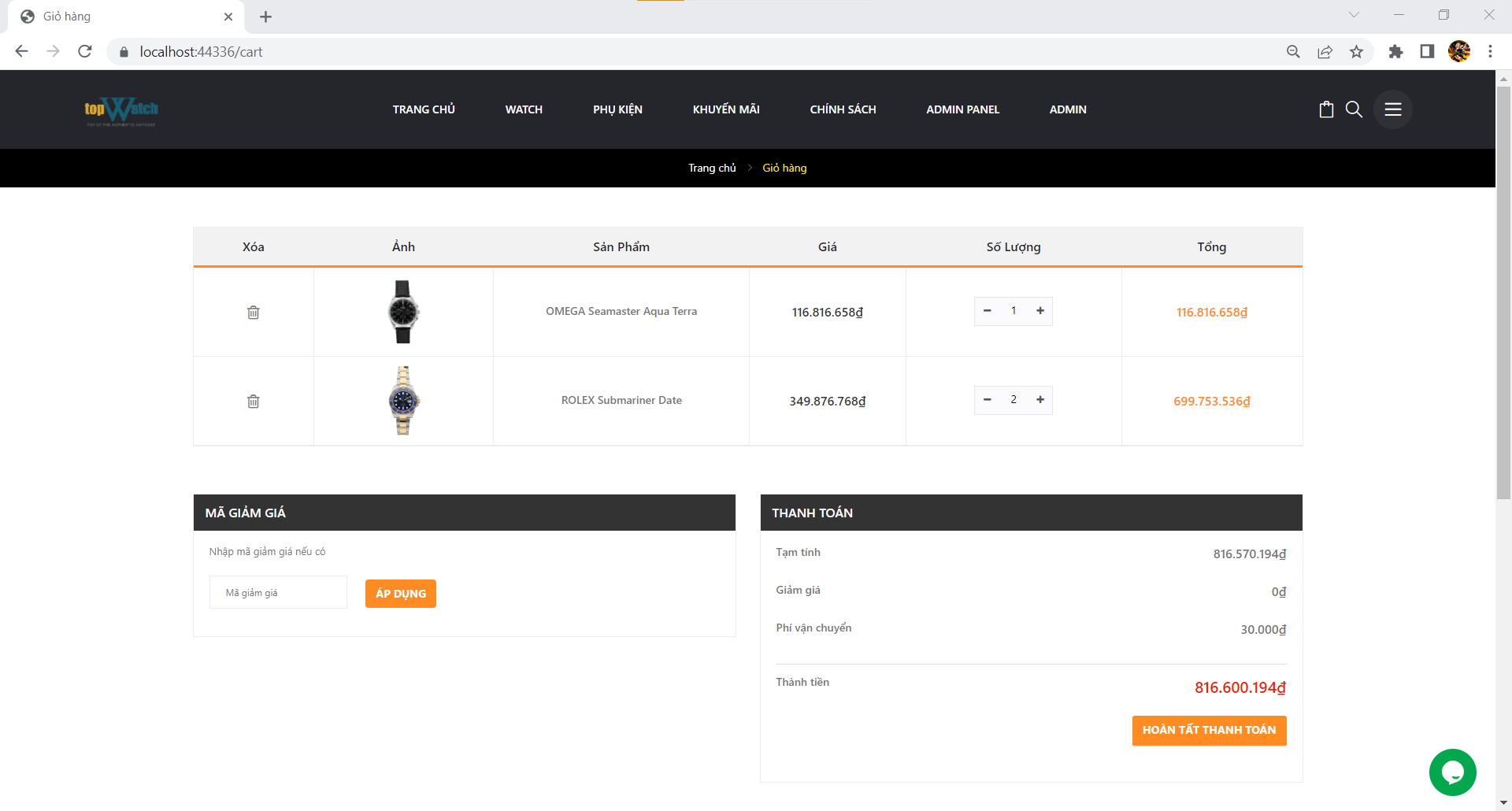
Hình 28 *Đăng nhập*

Màn hình đăng nhập. Tại đây khách hàng sử dụng tài khoản đã đăng kí để đăng nhập vào hệ thống và đặt mua sản phẩm.



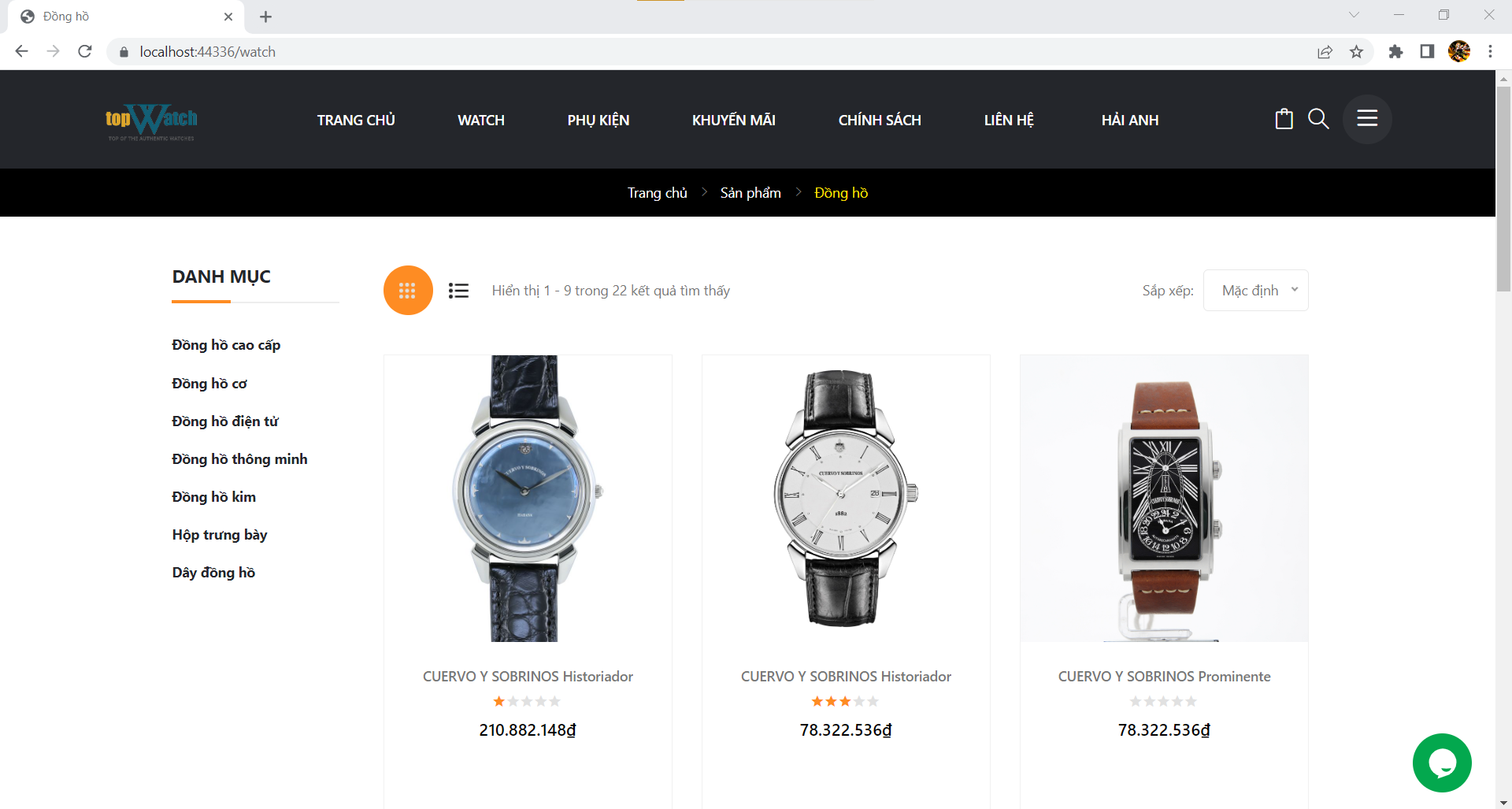
Hình 29 *Lịch sử mua hàng*

Màn hình lịch sử mua hàng cho phép người dùng kiểm tra những đơn hàng mình đã đặt mua, có thể xem chi tiết đơn hàng để xác định tình trạng đơn hàng.



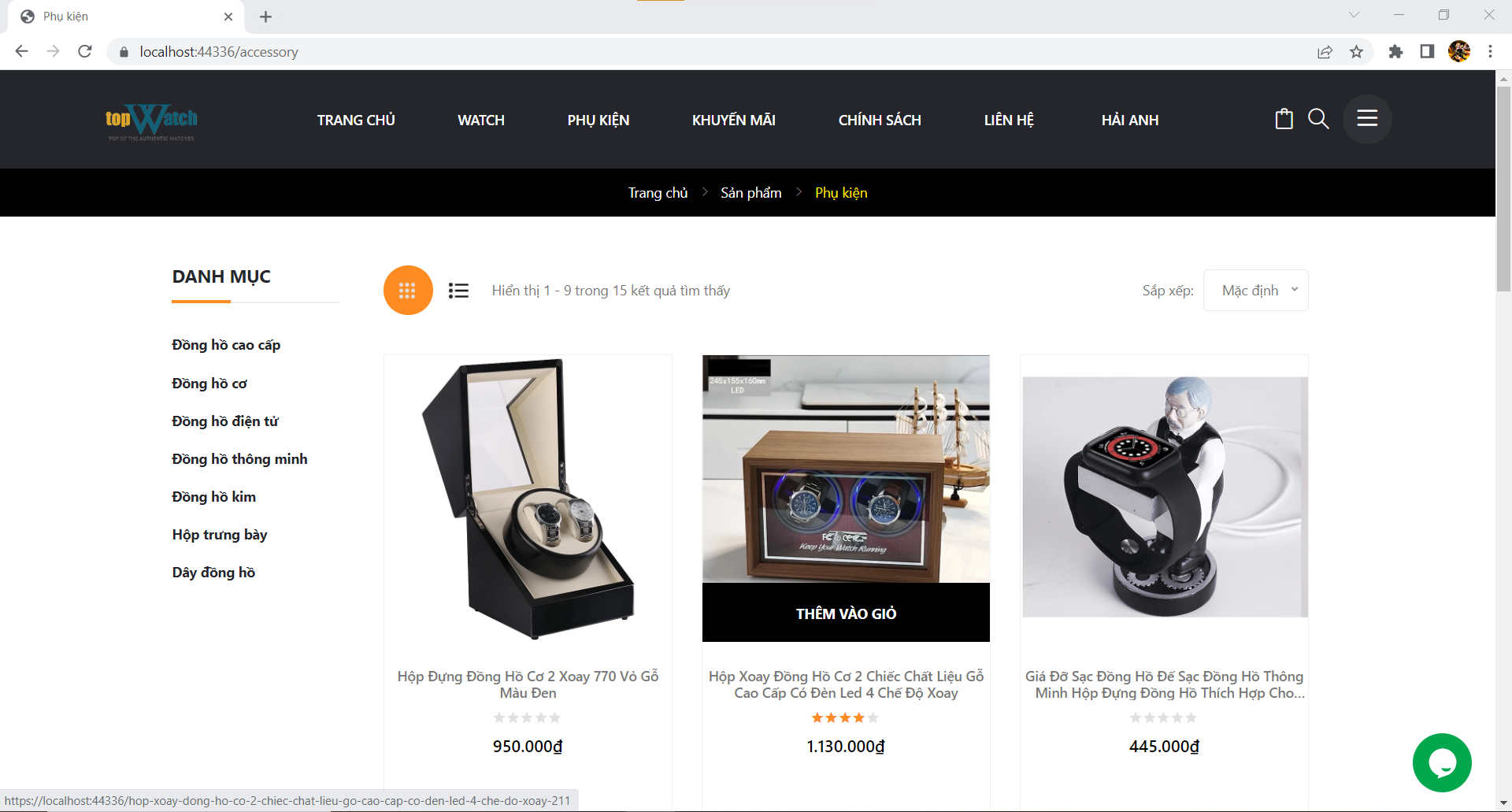
Hình 30 *Giỏ hàng*

Hiển thị những sản phẩm trong giỏ hàng, khi nhấn hoàn tất thanh toán sẽ đưa khách hàng sang trang thanh toán và hoàn tất quá trình đặt hàng, khác hàng có thể xóa bỏ mặt hàng không muốn bằng các bấm hình thùng rác trước mặt hàng.



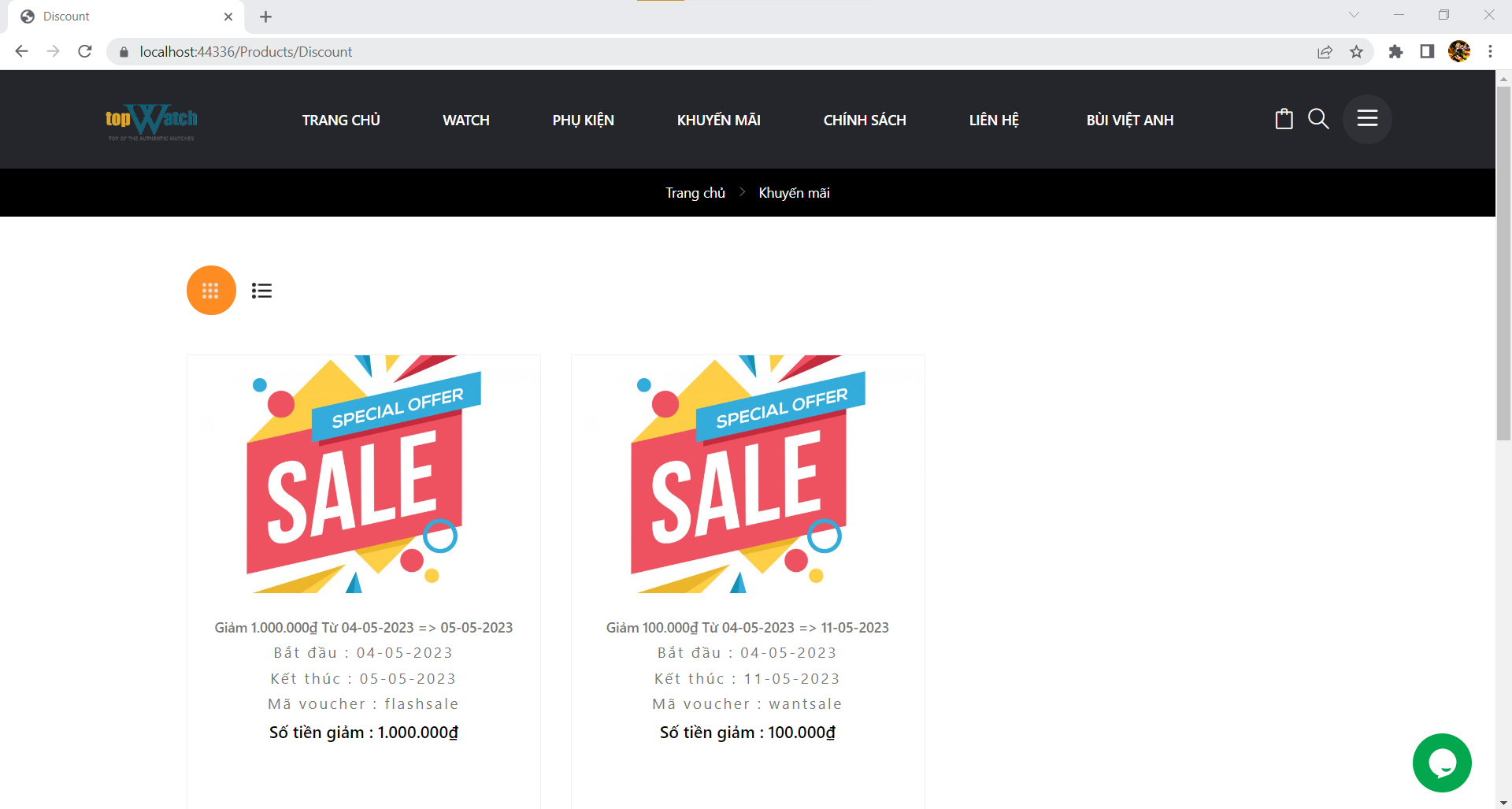
Hình 312 *Xem sản phẩm*

Hiển thị toàn bộ sản phẩm, cũng như các danh mục để người dùng có thể dễ dàng tìm kiếm



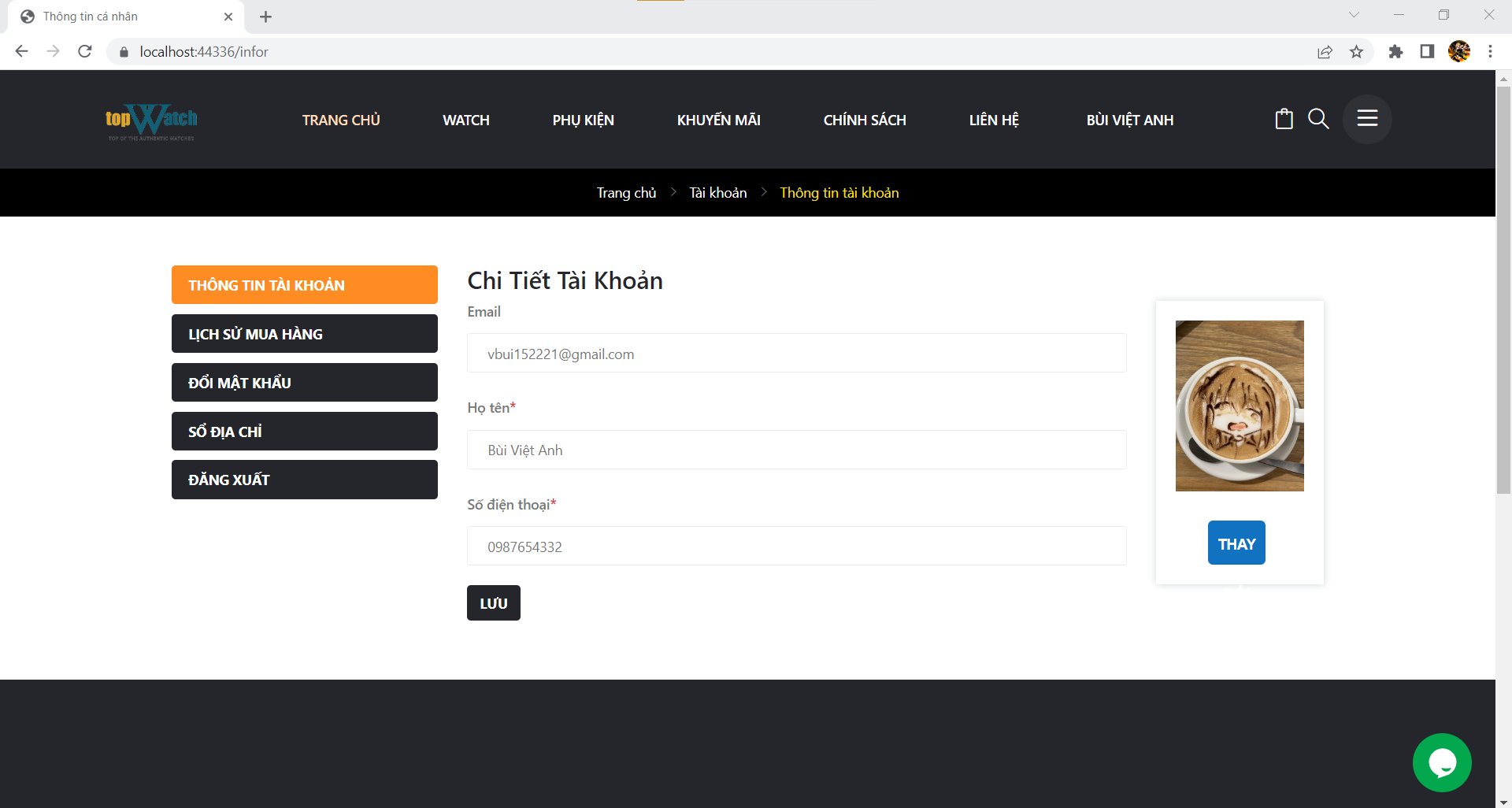
Hình 323 *Xem phụ kiện*

Hiển thị danh sách phụ kiến có thể dung để trưng bày cũng như thay thế các phụ tùng của đồng hồ.



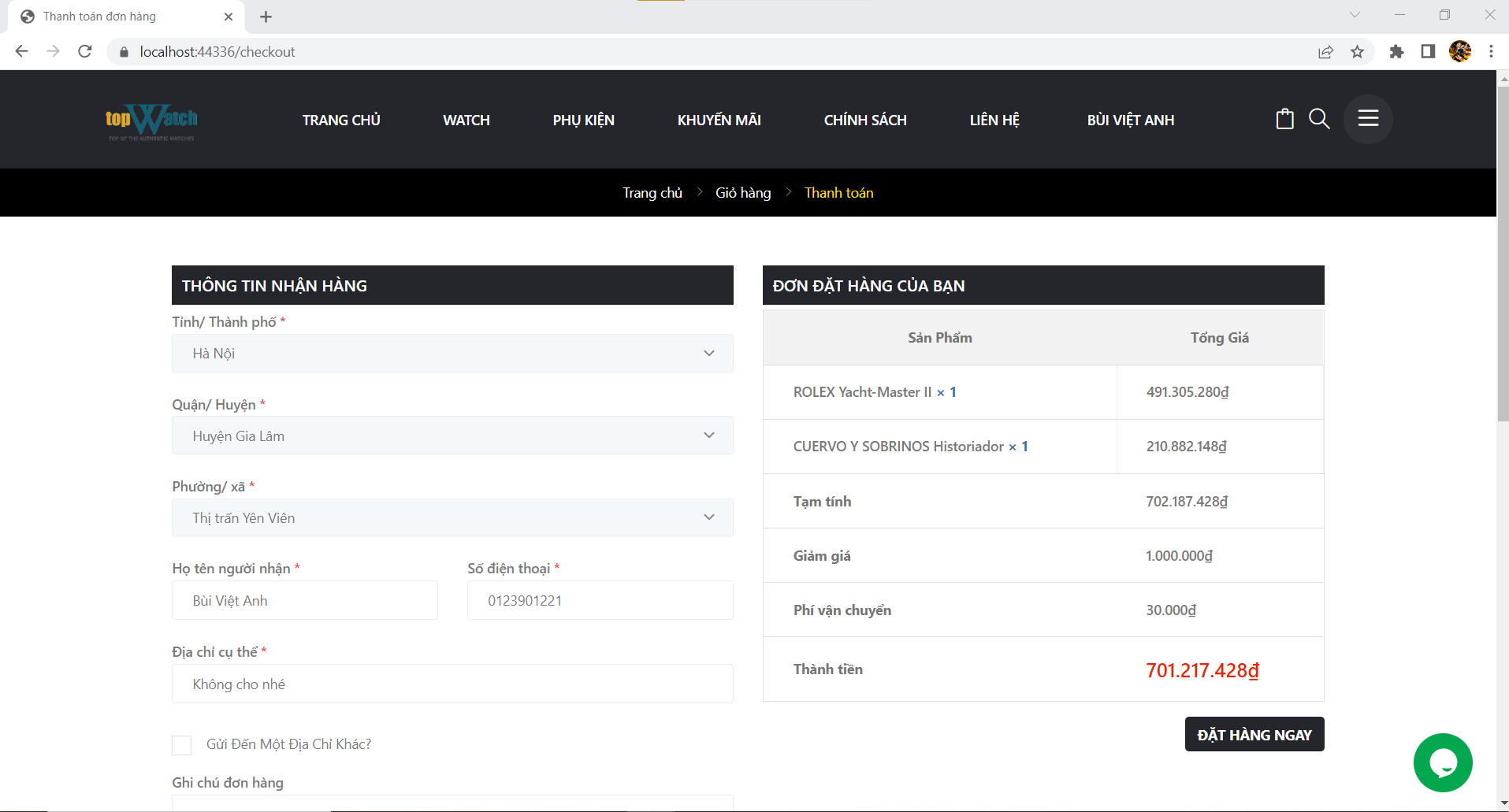
Hình 334 *Xem khuyến mãi*

Hiển thị các chương trình khuyến mãi bao gồm giá khuyến mãi, thời gian bắt đầu cũng như thời gian kết thúc, mã khuyến mãi



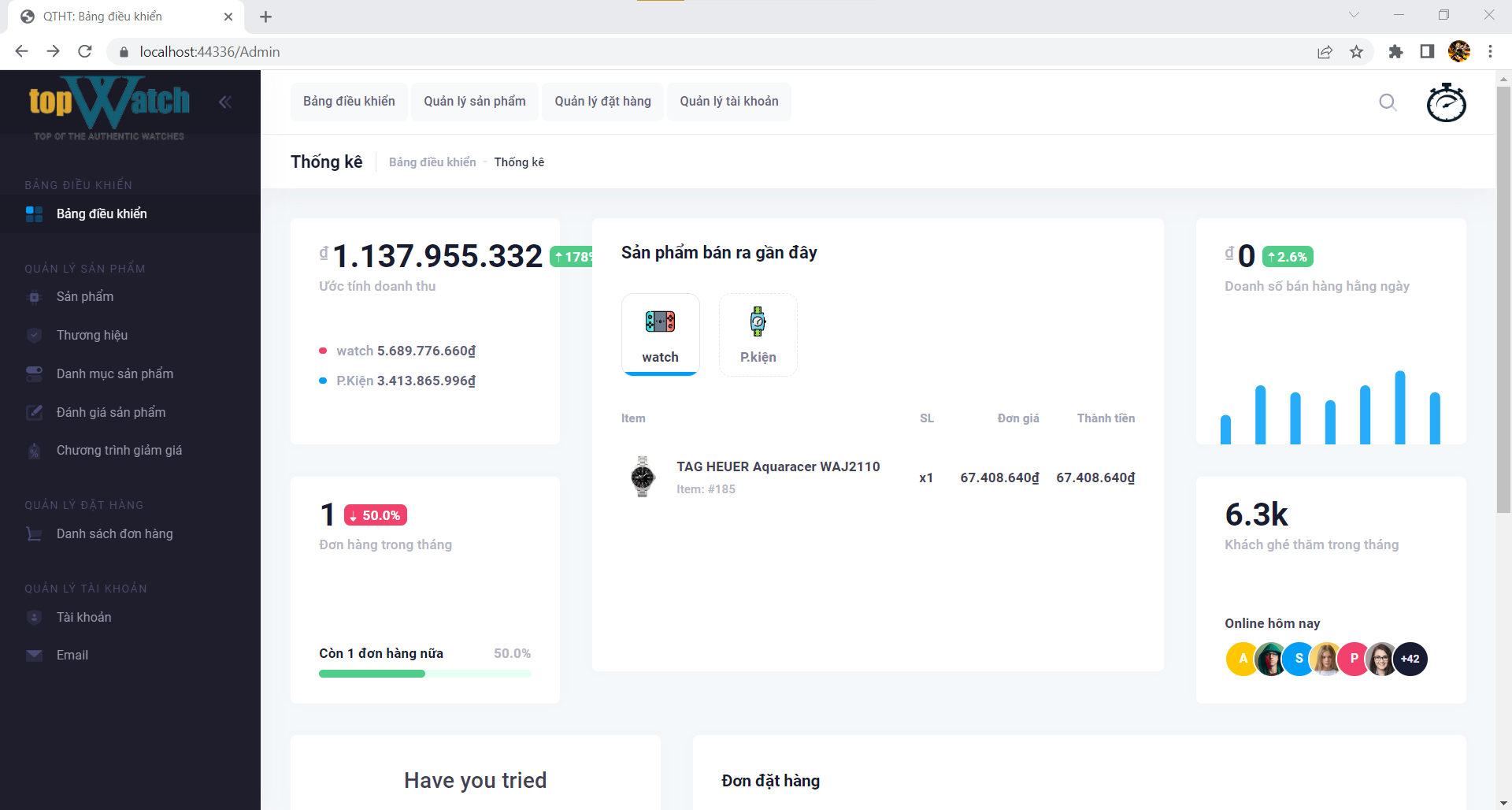
Hình 345 *Chỉnh sửa thông tin cá nhân của người dùng*

Người dùng có thể cập nhật các thông tin cá nhân bao gồm họ tên, số điện thoại và thêm cũng như chỉnh sửa các thông tin nơi giao hàng.



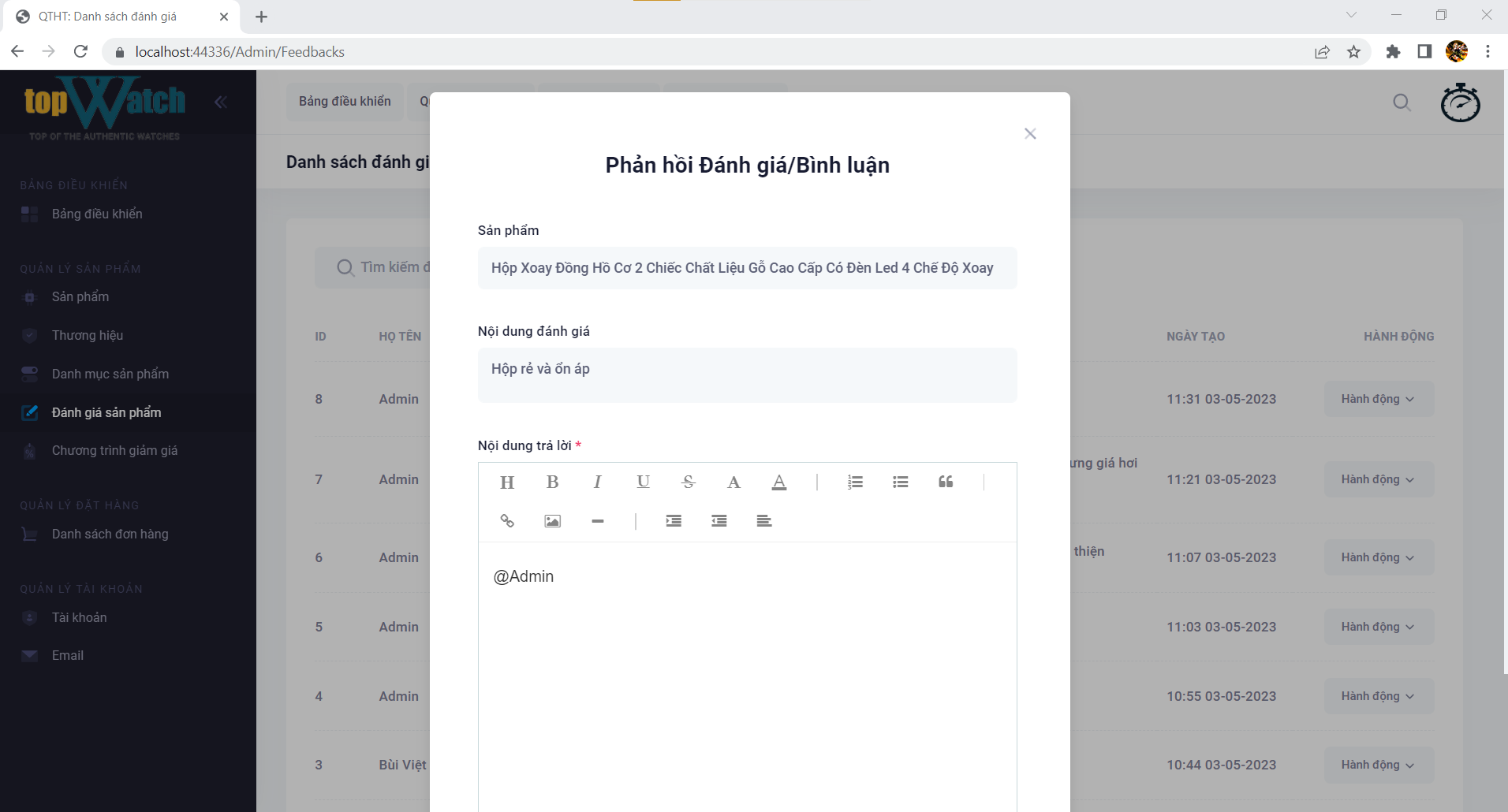
Hình 356 *Thanh toán đơn hàng*

Người dùng có thể xem lại chi tiết đơn hàng cũng như thay đổi địa chỉ nhận hàng.



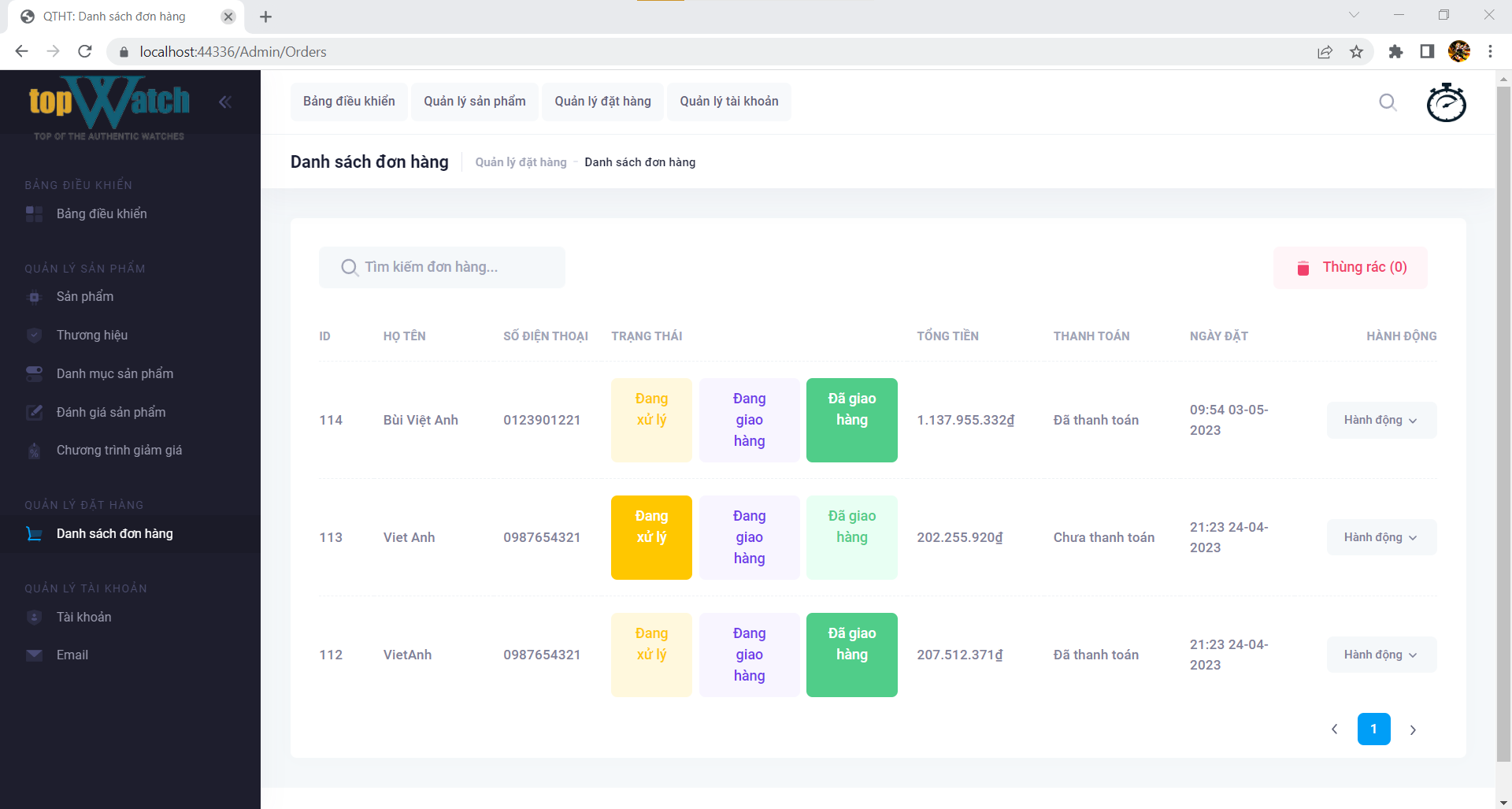
Hình 367 *Hiện thị thống kê*

Admin có thể xem chi tiết thống kê số tiền doanh thu, các mặt hàng được bán ra, xem các đơn hàng đã và đang chờ xử lý.



Hình 378 *Trả lời đánh giá sản phẩm*

Admin và nhân viên có thể trả lời các đánh giá sản phẩm từ phía khách hàng để có thể cung cấp cũng như trao đổi thông tin với phía khách hàng.



Hình 389 *Danh sách đơn hàng*

Admin và nhân viên có thể xem và duyệt trạng thái của đơn hàng cũng như có thể hủy đơn hàng của người dùng vào thùng rác trong trường hợp gặp bất chắc với điều kiện là đơn hàng chưa được nhận và chưa thanh toán.

## Kiểm thử phần mềm

### Kiểm thử Trang chủ

* Có đưa ra được những danh mục sản phẩm bán chạy, sản phẩm mới.
* Không bị scroll ngang.
* Đã có responsive website, để việc sử dụng trên những thiết bị khác nhau được thuận tiện, không bị đảo lộn các thẻ html.
* Đã có hover khi di chuột vào những đường link.
* Những đường link khi click đều ra nội dung chính xác.

### Kiểm thử tìm kiếm

* Tìm kiếm chính xác nội dung cần tìm kiếm
* Đã có tìm kiếm lọc theo giá, sắp xếp theo chữ a-z, z-a vv…

### Kiểm thử trang xem thông tin chi tiết sản phẩm

* Có hiệu ứng hover khi di chuột vào ảnh của sản phẩm.
* Hiển thị đầy đủ thông tin của sản phẩm.
* Hiển thị số lượng còn của sản phẩm.
* Có đánh giá sản phẩm.
* Có nhiều hình ảnh về sản phẩm giúp người mua hàng xem được các góc khác nhau của sản phẩm.

### Kiểm thử giỏ hàng

* Hiển thị đầy đủ thông tin cần thiết về 1 sản phẩm, số lượng mua, có mã giảm giá và hiển thị tổng tiền của 1 sản phẩm cũng như tổng tiền từ tất cả sản phẩm đã chọn.
* Có thể thay đổi số lượng muốn mua tùy ý.
* Có xóa sản phẩm trong giỏ hàng.

### Kiểm thử thanh toán

* Chưa có thanh toán online.
* Đã có hiển thị thông tin nhận hàng để khách hàng điều chỉnh.

### Kiểm thử các sản phẩm trong các Category

* Sản phẩm mới nhất hiển thị đúng.
* Sản phẩm bán chạy nhất hiển thị đúng.
* Danh sách đồng hồ hiển thị đúng.
* Danh sách phụ kiện hiển thị đúng.

### Kiểm thử các nội dung sau khi order

* Lịch sử mua hàng đã hiển thị đơn hàng khi đặt hàng.
* Xem thông tin chi tiết đơn hàng hiển thị đầy đủ những sản phẩm đã đặt ở đơn hàng đó.

### Kiểm thử các nội dung khác

* Đăng nhập thành công.
* Đăng kí thành công.
* Đổi mật khẩu thành công.
* Lọc sản phẩm theo các tiêu chí thành công.
* Người quản trị quản trị hệ thống thành công.
* Đã có phân quyền cho nhân viên và admin.

# KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM

## Những mục tiêu đề ra

* Code giao diện hoàn chỉnh với đầy đủ chức năng, hình ảnh, các hiệu ứng.
* Xây dựng cơ sở dữ liệu chặt chẽ.
* Sử dụng ngôn ngữ đã tìm hiểu để code chương trình.
* Hoàn thiện tất cả các chức năng cơ bản như thêm, sửa xóa đối với quản trị viên và các chức năng nâng cao như phân quyền.
* Hoàn thiện các chức năng của khách hàng: đăng kí tài khoản, đăng nhập, xem hàng, đặt hàng, thêm vào giỏ hàng và đồng thời có check tài khoản bị trùng email hay không.
* Nắm vững kĩ năng debugger trong khi làm chương trình.
* Chỉnh sửa giao diện trực tiếp trên trang web.
* Phân trang, mã hóa mật khẩu.

## Nội dung đã thực hiện

* Đã xây dựng được cơ sở dữ liệu hoàn chỉnh, chặt chẽ.
* Hoàn thiện giao diện khách hàng và người quản trị.
* Sử dụng ngôn ngữ asp.net để làm chương trình.
* Các chức năng của quản trị viên và khách hàng đều hoàn thành.
* Thực hiện được kĩ năng debugger để test tìm lỗi khi thực hiện code.
* Chỉnh sửa giao diện trực tiếp trên trang web giúp giảm thời gian code giao diện.
* Sử dụng thêm những phần mềm: case studio 2, rational rose để thực hiện hoàn thành báo cáo.

## Bài học kiến thức

* Những kiến thức và kỹ năng học được thông qua thực hiện đề tài của môn học như nắm chắc hơn những kiến thức về Asp.Net MVC, kiến thức về kết nối cơ sở dữ liệu, kĩ năng làm việc cá nhân, kĩ năng viết báo cáo.
* Những chuẩn đầu ra của học phần đã đạt được trong quá trình thực hiện thực tập như kiến thức về Asp.Net MVC, cách thực hiện với cơ sở dữ liệu.
* Những bài học kinh nghiệm được rút ra sau khi kết thúc thực tập như khi làm 1 dự án công nghệ thông tin.
* Và hiểu rõ hơn về ngôn ngữ C# cụ thể là Asp.Net MVC cũng như biết cách dùng phần mềm Visual Studio.
* Sau thời gian làm đồ án em đã biết cách sử dụng phần mềm Visual Studio, biết cách import các thư viện để hỗ trợ việc lập trình, biết cách kết nối Visual Studio với SQL Server để phục vụ việc quản lý dữ liệu nhanh chóng và tối ưu hơn.

## Hướng phát triển.

Hướng phát triển tiếp theo của đề tài là sẽ tiền hành lập trình mạng để phần mềm có thể tương tác dễ hơn trên các thiết bị khác nhau cũng như trên các nền tảng khác nhau, tích hợp thanh toán trực tiếp qua ngân hàng hoặc các ví điện tử khác. Ngoài ra còn thay đổi giao diện cho ứng dụng thân thiện với người dùng hơn và bổ sung thêm chức năng, tăng khả năng bảo mật cho phần mềm.

# Tài liệu tham khảo

* 1. <https://www.w3schools.com/>
  2. Trang chủ ASP.NET: <https://dotnet.microsoft.com/apps/aspnet>
  3. Tài liệu chính thức của Microsoft về ASP.NET: <https://docs.microsoft.com/en-us/aspnet/>
  4. ASP.NET Core Documentation: <https://docs.microsoft.com/en-us/aspnet/core/>
  5. ASP.NET tutorials on W3Schools: <https://www.w3schools.com/asp/>
  6. ASP.NET tutorials on tutorialspoint: <https://www.tutorialspoint.com/asp.net/>
  7. ASP.NET videos on Pluralsight: <https://www.pluralsight.com/paths/asp-net-core>
  8. ASP.NET videos on Udemy: <https://www.udemy.com/topic/asp-net/>
  9. ASP.NET videos on YouTube: <https://www.youtube.com/results?search_query=asp.net+tutorial>