BỘ CÔNG THƯƠNG

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

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

KIỀU MINH QUANG

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

NGÀNH KỸ THUẬT PHẦN MỀM

**Xây dựng website bán đồng hồ đeo tay sử dụng asp.net mvc**

CBHD: Th.s Vũ Minh Yến

Sinh viên:Kiều Minh Quang

KỸ THUẬT PHẦN MỀM

Mã số sinh viên: 2019601235

Hà Nội – Năm 2023

**LỜI NÓI ĐẦU**

Ngày nay, việc ứng dụng công nghệ thông tin vào quản lý kinh doanh đã phát triển mạnh mẽ ở các nước tiên tiến trên thế giới. Đặc biệt trong mùa dịch, việc mua bán hàng là rất cần thiết. Trước tình hình đó, vấn đề áp dụng Tin học để quản lý, mua bán hàng dường như đã thay thế rất nhiều cho các phương thức mua hàng truyền thống, nổi trội như hàng loạt các ứng dụng mua hàng ra đời như Shopee, Lazada, Tiki.

Việc mua bán hàng online đang là một trong những nhu cầu hết sức thiết yếu hiện nay trong đó có cả những mặt hàng về đồng hồ và phụ kiện, do đó hệ thống hỗ trợ mua đồng hồ đeo tay cũng là một trong những vấn đề đang được quan tâm. Việc xây dựng website thương mại điện tử bán đồng hồ đeo tay trực tuyến góp phần giúp người dùng dễ dàng tìm hiểu và đặt mua chiếc đồng hồ chính hãng và sang trọng một cách dễ dàng.

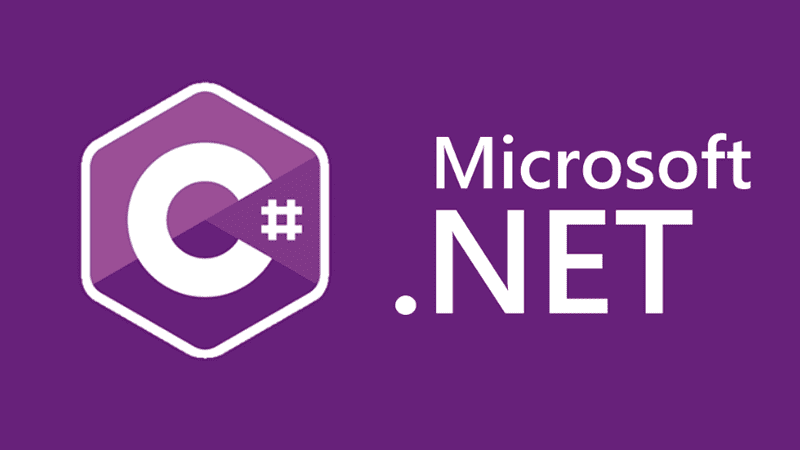
Sau khi tìm hiểu và dưới sự hướng dẫn của cô Vũ Minh Yến, em đã phân tích và thiết kế được **Website bán đồng hồ đeo tay sử dụng ASP.Net MVC**. Để giúp khách hàng đặt mua được những chiếc đồng hồ theo ý thích của mình, trước hết cần có một hệ thống tốt, một website có các dữ liệu chính xác, các đánh giá tốt của người dùng. Đó cũng là lý do mà em chọn đề tài này.

Để hoàn thành được đồ án tốt nghiệp này, em xin được gửi lời cảm ơn chân thành đến các thầy cô trong khoa Công nghệ thông tin Trường Đại học Công Nghiệp Hà Nội đã tận tình giảng dạy và trang bị kiến thức cho em trong suốt thời gian em học tập tại trường. Cô giáo hướng dẫn đề tài – T.S Vũ Minh Yến, giảng viên Khoa Công nghệ Thông tin Trường Đại học Công Nghiệp Hà Nội- đã tận tụy hết lòng giúp đỡ, hướng dẫn, chỉ dẫn tận tình để giúp em hoàn thành được đồ án. Bạn bè đã góp ý và giúp đỡ, bên cạnh động viên và khích lệ em trong suốt thời gian em làm đồ án.

**Hà Nội Ngày …. tháng … năm 2023**

**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

1. Tổng quan về các công nghệ:



*Hình 1: ngôn ngữ c#*

### 1.1 Ngôn ngữ C Sharp (C#)

C# là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư dẫn đầu là Andres Hejlsberg và Scott Wiltamuth của Microsoft vào năm 2000. Nó là một ngôn ngữ được xây dựng dựa trên nền tảng của C++ và Java. Nhờ vậy mà chúng hầu như đều có cấu trúc chương trình khá giống nhau. Một số cải tiến của nó đó chính là cấu trúc được rút gọn sao cho dễ nhớ và đơn giản hơn.

Ngôn ngữ lập trình C# được các chuyên gia nhận xét là loại ngôn ngữ thuần hướng các đối tượng. So với những loại ngôn ngữ lập trình khác thì C# sử dụng lượng từ khóa ít hơn rất nhiều. Nhờ đặc điểm này mà các lập trình viên đều thuận tiện hơn rất nhiều cho việc xây dựng lên các đối tượng dành riêng cho mình.

C# được thiết kế cho các ngôn ngữ chung cơ sở hạ tầng (Common Language Infrastructure – CLI), trong đó bao gồm các mã (Executable Code) và môi trường thực thi (Runtime Environment) cho phép sử dụng các ngôn ngữ cao cấp khác nhau trên đa nền tảng máy tính và kiến trúc khác nhau.

**Các đặc điểm của C Sharp (C#):**

* **Đơn giản:** Đặc trưng đầu tiên của C# là loại bỏ những vấn đề phức tạo đã có trong Java và C++ như macro, template, tính đa kế thừa, lớp cơ sở ảo (hay còn gọi virtual base class). Các cú pháp, toán tử, biểu thức và cả tính năng của C# khá tương đương Java và C++ song đã qua cải tiến nên đơn giản hơn nhiều.
* **Hiện đại:** C# sở hữu nhiều khả năng như xử lý ngoại lệ, tự động trong thu gom bộ nhớ, bảo mật mã nguồn, dữ liệu mở rộng,... Đây là tất cả những đặc điểm được mong chờ ở một ngôn ngữ lập trình hiện đại.
* **Hướng đối tượng:** C# là một trong những ngôn ngữ được đánh giá là thuần hướng đối tượng. Nó sở hữu cả 4 tính chất quan trọng, đặc trưng là tính kế thừa, tính đóng gói, tính trừu tượng và tính đa hình.
* **Ít từ khóa:** Một trong những đặc trưng cơ bản của C# là ít từ khóa. Từ khóa được dùng trong ngôn ngữ này chỉ nhằm mục đích mô tả thông tin. Tuy ít từ khóa song C# vẫn rất mạnh mẽ. Lập trình viên có thể sử dụng nó để thực hiện mọi nhiệm vụ.
* **Mã nguồn mở:** C# là một trong những ngôn ngữ lập trình mã nguồn mở, được phát triển, điều hành một cách độc lập với Microsoft. Đây là một trong những nét độc đáo khiến ngôn ngữ này được biết đến và ưa chuộng rộng rãi.
* **Đa nền tảng:** C# là ngôn ngữ được sử dụng trong lập trình trên nhiều nền tảng. Các ứng dụng hoặc website được xây dựng bằng ngôn ngữ này có thể hoạt động tốt trên nhiều nền tảng như Windows, Linux và Mac.
* **Tiến hóa:** C# vẫn đang được nâng cấp và cho ra mắt các phiên bản mới với nhiều tính năng vượt trội và khả năng làm việc mạnh mẽ hơn. Hiện C# có thể làm việc với console, điện toán đám mây, phần mềm học máy,.

### 1.2 Hệ quản trị cơ sở dữ liệu:

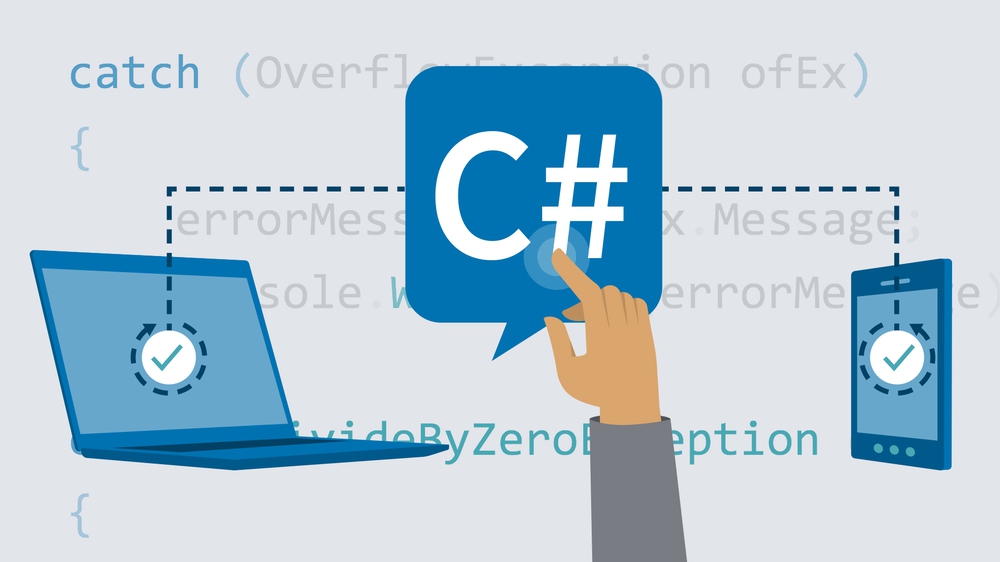


*Hình 2: hệ quản trị sql server*

SQL Server chính là một hệ quản trị cơ sở dữ liệu quan hệ sử dụng câu lệnh SQL để trao đổi dữ liệu giữa máy cài SQL Server và máy Client. Một Relational Database Management System (RDBMS) gồm có: databases, database engine và những dữ liệu khác

Phần mềm SQL Server được sử dụng khá rộng rãi vì nó được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn lên đến Tera – Byte cùng lúc phục vụ cho hàng ngàn user. Bên cạnh đó, ứng dụng này cung cấp đa dạng kiểu lập trình SQL từ ANSI SQL (SQL truyền thống) đến SQL và cả T-SQL (Transaction-SQL) được sử dụng cho cơ sở dữ liệu quan hệ nâng cao.

## Ưu điểm nổi bật của ngôn ngữ lập trình C#:

[](https://bizflyportal.mediacdn.vn/bizflyportal/images/c-l16178488146711.jpg)Hình 3: Ưu điểm nổi bật của ngôn ngữ lập trình C#

C# gần gũi với Java và C++, nhờ vậy mà nó kế thừa được tất cả các ‘tinh hoa’ của hai ngôn ngữ này. Lập trình viên có kiến thức về hai ngôn ngữ trên có thể dùng C# dễ dàng.

Cộng đồng những người sử dụng C# đang phát triển với tốc độ chóng mặt. Lập trình viên có thể tham khảo và tìm kiếm thông tin dễ dàng.

C# có khả năng tạo ra mọi ứng dụng và phổ biến trong giới lập trình. Đặc biệt là lập trình game.

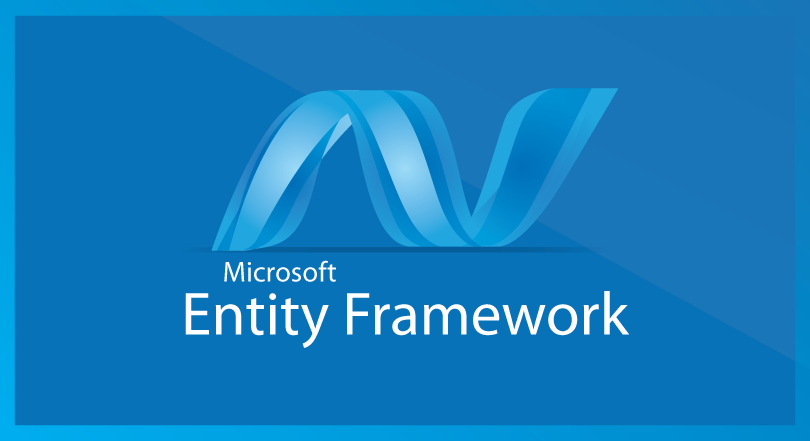
**Ứng dụng của ngôn ngữ lập trình C#**

Trên windows: C# với framework .NET được dùng để tạo ra các ứng dụng trên Windows như Microsoft Office, Visual Studio, Skype, Photoshop,...

Trên Web: C# hỗ trợ lập trình viên tạo các ứng dụng web nhờ sự hỗ trợ của asp.net. Với ngôn ngữ này, các ứng dụng có thể chạy mượt mà trên máy chủ.

Thành phần, điều khiển: C# còn được ứng dụng trong xây dựng nhiều thành phần của máy chủ. Đây là một trong các ứng dụng quan trọng của ngôn ngữ lập trình C#.

### 1.3 Giới thiệu framework:



*Hình 4: Entity framework*

Entity Framework là một khung ORM mã nguồn mở cho các ứng dụng .NET được Microsoft hỗ trợ. Nó cho phép các nhà phát triển làm việc với dữ liệu bằng cách sử dụng các đối tượng của các lớp cụ thể của miền mà không cần tập trung vào các bảng và cột cơ sở dữ liệu cơ bản nơi dữ liệu được lưu trữ. Với Entity Framework, các nhà phát triển có thể làm việc ở mức độ trừu tượng cao hơn khi họ xử lý dữ liệu và có thể tạo và duy trì các ứng dụng hướng dữ liệu với ít mã hơn so với các ứng dụng truyền thống.

Entity Framework tạo EDM (mô hình dữ liệu thực thể) dựa trên các thực thể POCO (Plain Old CLR Object) với các thuộc tính get/set của các loại dữ liệu khác nhau. Nó sử dụng mô hình này khi truy vấn hoặc lưu dữ liệu thực thể vào các cơ sở dữ liệu cơ bản.

Cho phép sử dụng các truy vấn LINQ (C# / VB) để truy xuất dữ liệu từ cơ sở dữ liệu cơ bản. Nhà cung cấp cơ sở dữ liệu sẽ dịch các truy vấn LINQ này sang ngôn ngữ truy vấn dành riêng cho cơ sở dữ liệu.

Thực thi các lệnh INSERT, UPDATE và DELETE vào cơ sở dữ liệu dựa trên những thay đổi xảy ra với các thực thể của bạn khi bạn gọi phương thức Savechange().

Cung cấp một tập hợp các lệnh di chuyển có thể được thực thi trên bảng điều khiển trình quản lý gói NuGet hoặc giao diện dòng lệnh để tạo hoặc quản lý lược đồ cơ sở dữ liệu cơ bản.

### 1.4 Môi trường lập trình:



*Hình 5: Visual Studio*

Visual Studio là một trong những công cụ hỗ trợ lập trình website rất nổi tiếng hiện nay của Microsoft và chưa có một phần mềm nào có thể thay thế được. Visual Studio được viết bằng 2 ngôn ngữ chính là C# và VB+. Đây là 2 ngôn ngữ lập trình giúp người dùng có thể lập trình được hệ thống một cách dễ dàng và nhanh chóng thông qua Visual Studio.

Hiện nay qua nhiều phiên bản Visual Studio hỗ trợ đa nền tảng từ C/C++, HTML, CSS, JavaScript.

Hỗ trợ khả năng Debug hiệu quả và dễ dàng thông qua việc chạy từng câu lệnh và theo dõi sự thay đổi trạng thái của chương trình qua giá trị các biến, cách vận hành của mã.

Giao diện dễ dàng, thân thiện cho phép xây dựng chuyên nghiệp kéo thả, hỗ trợ cho người mới bắt đầu tiếp cận nhanh hơn.

2. Xác định yêu cầu:

2.1 Yêu cầu chung:

* Kiểu website: đơn giản, thân thiện, có menu chức năng từng thẻ, phần và mục.
* Phong cách thiết kế: đơn giản
* Không sử dụng các font chữ không chuẩn, khó đọc. Nên sử dụng các font Unicode chuẩn như Roboto, Arial, Tahoma và Time News Roman.
* Giao diện thân thiện với người dùng, dễ sử dụng
* Trang trí vừa phải, màu sắc dễ nhìn, ổn định

2.2 Các tác vụ cơ bản:

Giao diện của người quản trị:

* Tiếp cận người dùng, dễ dàng sử dụng.
* Màu sắc đơn giản, thể hiện đủ các thông tin.

Tác vụ thêm, sửa, xóa:

* Cho phép quản trị viên thêm, sửa, xoá các chức năng trong trang quản trị.
* Cho phép người dùng có thể chỉnh sửa thông tin cá nhân.

Tác vụ thống kê:

* Bao gồm: thống kê đơn hàng mới, đơn hàng đang xử lý, đơn hàng đã hoàn thành, đơn hàng đã hủy, sản phẩm mới, số lượng đã bán, tổng doanh thu.

2.3 Yêu cầu hệ thống:

Hệ thống sử dụng hệ quản trị cơ sở dữ liệu đủ đáp ứng được dữ liệu lưu vào và truy xuất thông tin đầy đủ.

Thông tin về sản phẩm đồng bộ và phân quyền chặt chẽ.

Bảo mật thông tin tốt không để mất dữ liệu hoặc mất thông tin người dùng.

Các chức năng của hệ thống hoạt động ổn định.

2.4 Yêu cầu chức năng:

Website phải có các tính năng cơ bản phù hợp với các thao tác nghiệp vụ của những cửa hàng kinh doanh đồng hồ . Các nghiệp vụ chính bao gồm: Hệ thống, người dùng và quản trị viên.

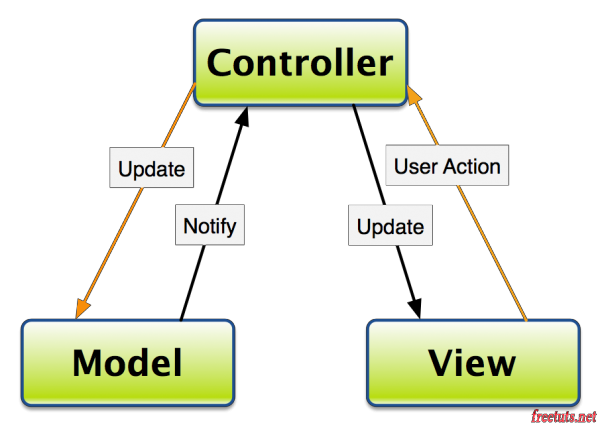
Hệ thống: Đăng ký, đăng nhập

Người dùng: Quản lý và chỉnh sửa thông tin cá nhân, thêm sản phẩm vào giỏ hàng và đặt hàng, thanh, đánh giá sản phẩm.

Quản trị viên: Quản lý và chỉnh sửa các mục chức năng, thống kê.

3. Mô hình giải pháp

Website sử dụng mô hình MVC để thực hiện các yêu cầu của khách hàng.



*Hình 6: Mô hình MVC*

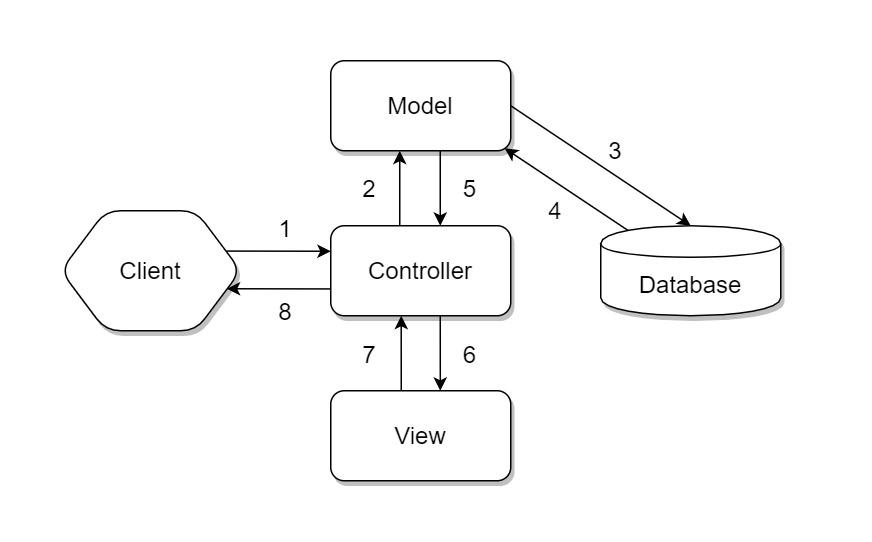
Mô hình MVC được chia làm 3 lớp xử lý gồm Model – View – Controller

**Model**: Là nơi chứa những nghiệp vụ tương tác với dữ liệu hoặc hệ quản trị cơ sở dữ liệu. Nó bào gồm các class/function xử lý nhiều nghiệp vụ như kết nối database, truy vấn dữ liệu, thêm sửa xóa dữ liệu…

**View**: Là nơi chứa những giao diện như một nút bấm, khung nhập, menu, hình ảnh… nó đảm nhiệm nhiệm vụ hiển thị dữ liệu giúp người dùng tương tác với hệ thống.

**Controller**: Là nơi tiếp nhận những yêu cầu xử lý được gửi từ người dùng, nó sẽ gồm những class/function xử lý nghiệp vụ logic giúp lấy đúng dữ liệu thông tin cần thiết nhờ các nghiệp vụ lớp Model cung cấp và hiển thị dữ liệu đó ra cho người dùng qua lớp View.

Luồng xử lý trong MVC

****

*Hình 7: Luồng hoạt động của mô hình MVC*

* Đầu tiên, client sẽ gửi một request tới server thông qua Controller.
* Controller tiếp nhận request và xử lý luồng đi tiếp theo của yêu cầu. Trong trường hợp chỉ chuyển từ trang này sang trang khác mà không yêu cầu gửi và nhận dữ liệu thì lúc này Controller sẽ chuyển hướng lại cho browser và kết thúc luồng.
* Nếu request của người dùng yêu cầu phải xử lý dữ liệu thì Controller gọi xuống Model để lấy dữ liệu. Lúc này Model sẽ tương tác với Database để lấy dữ liệu, dữ liệu được Model gửi về Controller, Controller gọi đến View phù hợp với request kèm theo dữ liệu cho View. View sẽ lắp dữ liệu tương ứng vào HTML và gửi lại một HTML cho Controller sau khi thực hiện xong nhiệm vụ. Cuối cùng Controller sẽ trả kết quả về Browser.
* Nếu request của người dùng chỉ yêu cầu dữ liệu nhưng không chuyển trang thì thay vì Controller sẽ trả dữ liệu về cho View thì Controller sẽ response lại cho trình duyệt thông qua API, dữ liệu trả về thường sẽ là dạng JSON.

**Ưu điểm**

* Thể hiện tính chuyên nghiệp trong lập trình, phân tích thiết kế.

Do được chia thành các thành phần độc lập nên giúp chúng phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp, bảo trì….

**Nhược điểm**

* Đối với dự án nhỏ việc áp dụng mô hình MVC gây cồng kềnh.
* tốn thời gian trong quá trình phát triển.
* Tốn thời gian trung chuyển dữ liệu của các thành phần.

**CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

1. **Giới thiệu bài toán**

Hiện nay, việc kinh doanh trực tuyến đã và đang trở thành xu hướng, là một hình thức kinh doanh trên mạng internet thông qua các trang thương mại điện tử và mạng xã hội như: Thiết kế website bán hàng bán hàng qua facebook, shopee, tiki, …, để từ đó quảng bá thương hiệu một cách dễ dàng, hiệu quả và thu lại lợi nhuận khá tốt cho các cửa hàng và doanh nghiệp trên toàn thế giới chứ không riêng gì Việt Nam.

Qua khảo sát thị trường, nắm bắt được sự hạn chế của phần nhiều website bán hàng còn chưa được linh hoạt, người bán bị phụ thuộc nhiều vào các chức năng được xây dựng cố định và khi muốn cập nhật cần liên hệ với các nhà phát triển web tốn kém nhiều chi phí và thời gian. Vì vậy em chọn đề tài Website bán đồng hồ đeo tay bằng asp.net mvc với mục đích giải quyết các bài toán hạn chế trên.

1. **Giới thiệu chung về hệ thống**

* Hệ thống website bán đồng hồ đeo tay cho phép người quản trị thêm, chỉnh sửa, xóa thông tin của các mặt hàng, danh mục sản phẩm, khuyến mãi, …
* Hệ thống website cho phép kiểm soát số lượng hàng: khi bán hàng, người quản trị cần biết được số lượng hàng còn tồn tại trong kho.
* Thống kê: Hệ thống sẽ thống kê lại doanh thu theo các mốc thời gian cần thiết.

1. **Sơ đồ tổ chức nhân sự của hệ thống**

Hệ thống chia làm 5 nhóm người dùng chính:

+ Quản trị viên(Admin)

+ Nhân viên bán hàng(Sale person)

+ Biên tập viên(Editor)

+ Trợ lý, cộng tác viên(Assistant)

+ Nhân viên giao hàng(shiper)

Admin

Shiper

Assistant

Editor

Sale Person

Hình 8: sơ đồ tổ chức nhân sự của hệ thống

Quản trị viên(Admin)

**+** Vai trò: Đây là vai trò cao nhất trong hệ thống. Người quản trị có quyền quản lý, thao tác, chỉnh sửa, cập nhật, thêm mới toàn bộ thông tin trong hệ thống.

Nhân viên bán hàng(Sale person)

**+** Vai trò: Quản lý về sản phẩm, giao hàng, các đơn hàng, báo cáo thống kê kinh doanh cho người quản trị admin.

Biên tập viên(Editor)

+ Vai trò: Sáng tạo các nội dung về sản phẩmm danh mục sản phẩm nhằm mục tiêu thu hút khách hàng, đóng góp và tổng hợp ý kiến nội dung sáng tạo cho người quản trị admin.

Trợ lý, cộng tác viên(Assistant)

+ Vai trò: Quảng bá website cùng với các thương hiệu sản phẩm có trong website nhằm tăng lượng người xem và tăng lượng khách hàng mua sản phẩm.

Nhân viên giao hàng(shiper)

+ Vai trò: Nhận thông tin các mặt hàng cần giao từ Sale person sau đó liên hệ vận chuyển hàng đến địa chỉ của khách hàng.

1. **Hiện trạng hệ thống**

* Sản phẩm dịch vụ của bạn là gì?
* Trả lời: Các mặt hàng đồng hồ đeo tay và phụ kiện
* Khu vực bạn bán sản phẩm?
* Trả lời: khu vực Việt Nam
* Bạn kinh doanh bán sỉ hay bản lẻ:
* Trả lời: Chỉ bán lẻ
* Cửa hàng của bạn là cửa hàng nhỏ lẻ hay chuỗi cửa hàng?
* Trả lời: Cửa hàng lẻ
* Quy trình hoạt động cửa hàng của bạn như thế nào?
* Quy trình:

1. Nhận thông tin đơn hàng của khách hàng
2. Kiểm tra mẫu mã, bảo hành, chất lượng sản phẩm khách hàng yêu cầu
3. Lên đơn cho khách và ghi phiếu bảo hành
4. Giao sản phẩm đến địa chỉ của khách hàng
5. Trường hợp khách hàng không hài lòng về sản phẩm thì sẽ chuyển sản phẩm quay lại cửa hàng.

**5. Yêu cầu với hệ thống**

**5.1 Yêu cầu chức năng:**

- Quản lý toàn bộ thông tin sản phẩm

- Quản lý toàn bộ danh mục sản phẩm

- Quản lý toàn bộ thông tin đơn hàng

- Quản lý toàn bộ thông tin khuyến mãi

- Phân quyền admin.

**5.2 Yêu cầu phi chức năng:**

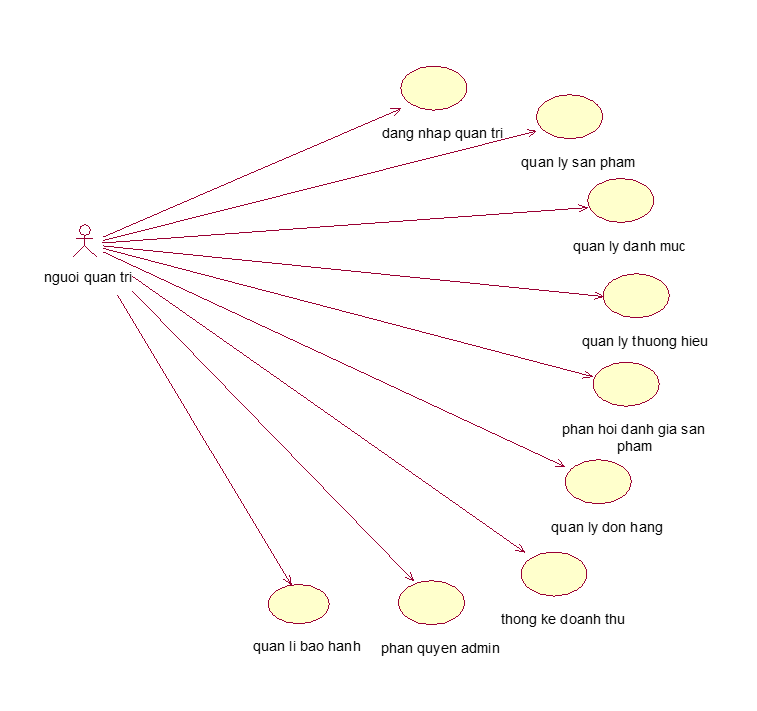
- Khả năng tiếp cận:

* Hệ thống có thể truy cập từ bất kỳ thiết bị nào có kết nối internet như PC, laptop, máy tính bảng, smart phone….
* Tính khả dụng:
* Người dùng có thể truy cập vào hệ thống bất cứ thời gian nào
* Tính bảo mật:
* Mật khẩu phải được mã hóa trong cơ sở dữ liệu
* Đảm bảo thông tin cá nhân của người dùng không được lan truyền ra bên ngoài
* Hiệu năng:
* Thời gian phản hồi nhanh
* Các request không kéo dài quá 10 giây
* Khả năng mở rộng:
* Hệ thống có thể được mở rộng theo yêu cầu và hoạt động tốt
* Back-end và Font-end có thể dễ dàng mở rộng linh hoạt.

**6. Biểu đồ usecase tổng quát**

****

*Hình 9: Biểu đồ use case tổng quát font-end*



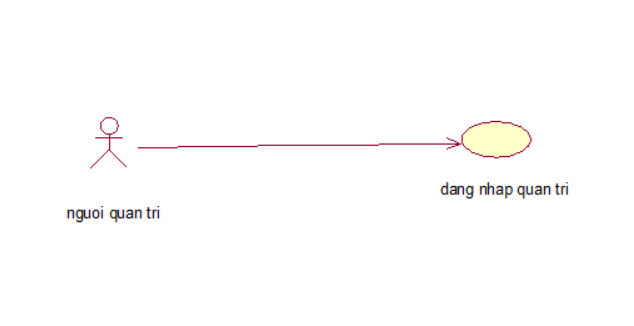
*Hình 10: Biểu đồ use case tổng quát back-end*

**7. Đặc tả usecase**

7.1 Đặc tả usecase phía back-end

7.1.1 Use case Đăng nhập quản trị

a) Biểu đồ use case:



*Hình 11: Biểu đồ use case đăng nhập phía back-end*

b) Đặc tả use case:

Use case này cho phép người quản trị đăng nhập để xác định quyền truy cập vào hệ thống.

* Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút “Đăng nhập” trên góc phải màn hình.
2. Hệ thống yêu cầu người quản trị nhập các thông tin bao gồm: Tên tài khoản và mật khẩu.
3. Người quản trị nhập thông tin và bấm nút “Đăng nhập”
4. Hệ thống xác thực tài khoản và mật khẩu, xác thực vai trò của người admin rồi chuyển hướng sang trang chủ.
5. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Sai tài khoản hoặc mật khẩu: tại bước 3 trong luồng cơ bản nếu người quản trị nhập tài khoản hoặc mật khẩu sai, hệ thống hiển thị thông báo lỗi. Người dùng có thể nhập lại hoặc bỏ qua thao tác. Use case kết thúc
2. Đăng ký: tại bước 3 trong luồng cơ bản, nếu người dùng bấm nút “Đăng ký” use case kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi, use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

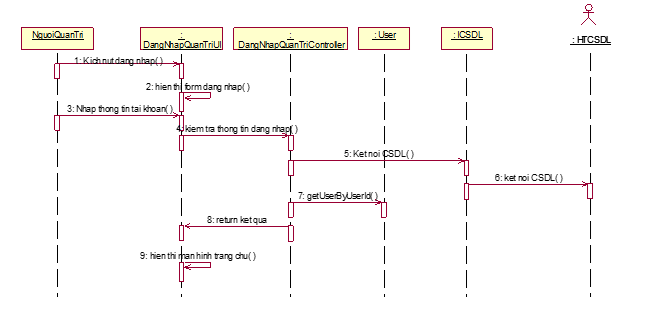
* Hậu điều kiện:

Nếu use case thành công, người quản trị sẽ đăng nhập được vào hệ thống. Nếu không trạng thái của hệ thống không thay đổi

* Điểm mở rộng:

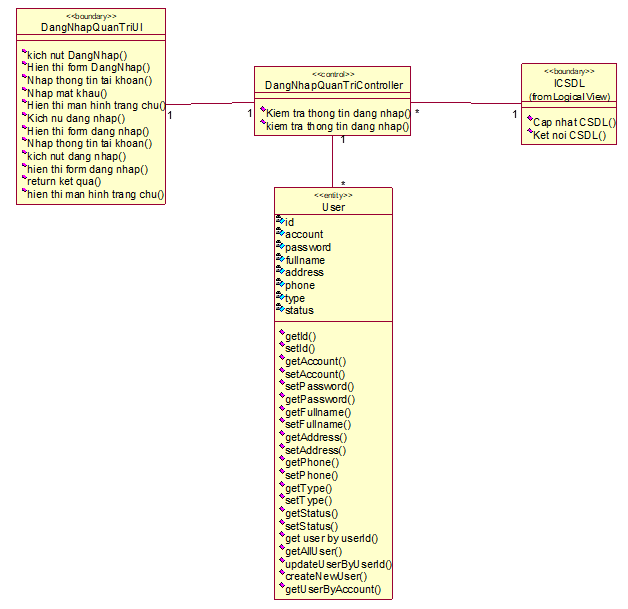
Không có.

c) Biểu đồ trình tự



*Hình 12: Biểu đồ trình tự use case đăng nhập phía back-end*

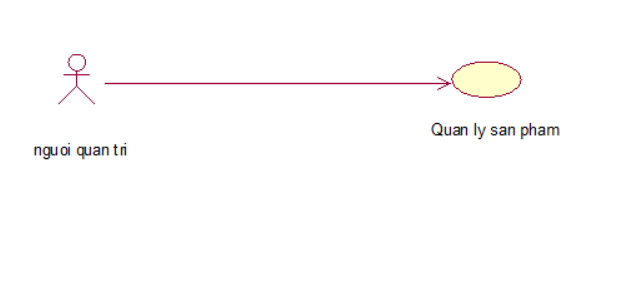
d) Biểu đồ lớp phân tích



*Hình 13: Biểu đồ lớp phân tích use case đăng nhập phía back-end*

7.1.2 Use case quản lý sản phẩm

a) Biểu đồ use case:



*Hình 14: Biểu đồ use case quản lý sản phẩm*

b) Đặc tả use case:

Use case này cho phép người quản trị quản lý các thông tin về sản phẩm trên hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút quản lý sản phẩm. Hệ thống lấy thông tin các sản phẩm gồm id, product\_name, title, description, quantity hiển thị lên màn hình.
2. Thêm mới sản phẩm
3. Người quản trị ấn vào nút thêm mới. Hệ thống hiển thị form thêm mới sản phẩm lên màn hình.
4. Người quản trị nhập thông tin thêm mới sản phẩm sau đó kích vào nút thêm. Hệ thống lưu các thông tin vào bảng product và hiển thị danh sách sản phẩm sau khi thêm mới.
5. Cập nhật sản phẩm
6. Người quản trị chọn sản phẩm cần cập nhật và kích vào icon cập nhật. Hệ thống lấy thông tin sản phẩm đã chọn theo id và hiển thị form cập nhật sản phẩm lên màn hình.
7. Người quản trị nhập thông tin cần cập nhật và ấn vào nút lưu. Hệ thống lưu thông tin sản phẩm đã cập nhật vào bảng product và hiển thị danh sách sản phẩm đã cập nhật lên màn hình.
8. Xóa sản phẩm
9. Người quản trị chọn sản phẩm cần xóa và kích vào icon xóa. Hệ thống hiển thị lên thông báo xác nhận xóa
10. Người quản trị kích vào nút xác nhận xóa sản phẩm. Hệ thống xóa sản phẩm trong bảng product và hiển thị thông báo xóa thành công lên màn hình.
11. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị một thông báo lỗi và use case kết thúc

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Người quản trị đăng nhập thành công.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

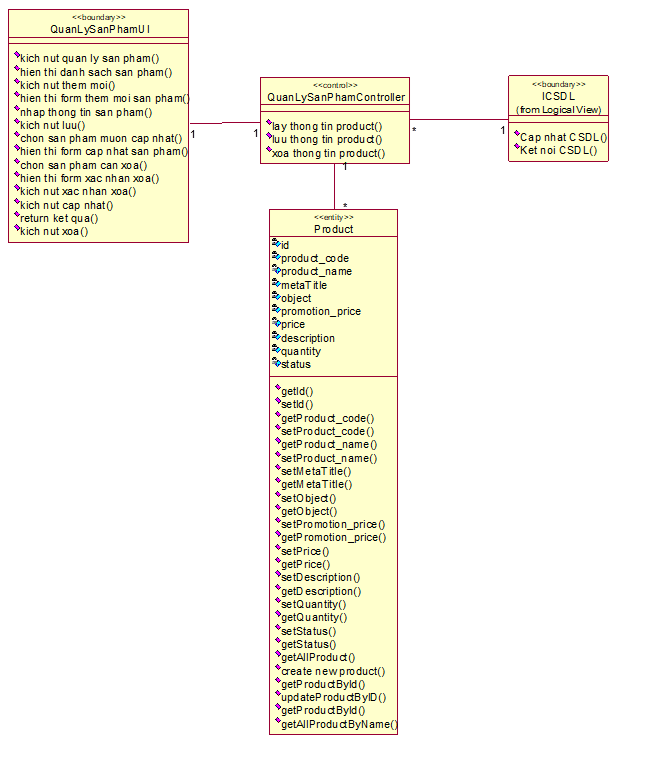
Không có.

c) Biểu đồ trình tự



*Hình 15: Biểu đồ trình tự use case quản lý sản phẩm*

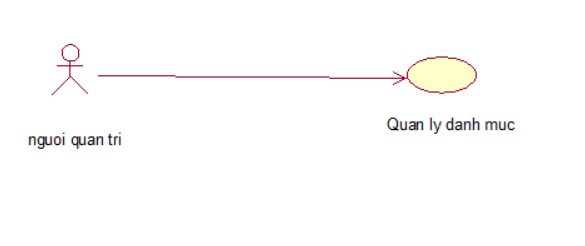
d) Biểu đồ lớp phân tích



*Hình 16: Biểu đồ lớp phân tích use case quản lý sản phẩm*

7.1.3 Use case quản lý danh mục

a) Biểu đồ use case:



*Hình 17: Biểu đồ use case quản lý danh mục*

b) Đặc tả use case:

Use case này cho phép người quản trị quản lý các thông tin về danh mục sản phẩm trên hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút quản lý danh mục. Hệ thống lấy thông tin các danh mục gồm id, name, metaTitle, description hiển thị lên màn hình.
2. Thêm mới danh mục
3. Người quản trị ấn vào nút thêm mới. Hệ thống hiển thị form thêm mới danh mục lên màn hình.
4. Người quản trị nhập thông tin thêm mới danh mục sau đó kích vào nút thêm. Hệ thống lưu các thông tin vào bảng category và hiển thị danh sách danh mục sau khi thêm mới.
5. Cập nhật danh mục
6. Người quản trị chọn danh mục cần cập nhật và kích vào icon cập nhật. Hệ thống lấy thông tin danh mục đã chọn theo id và hiển thị form cập nhật danh mục lên màn hình.
7. Người quản trị nhập thông tin cần cập nhật và ấn vào nút lưu. Hệ thống lưu thông tin danh mục đã cập nhật vào bảng category và hiển thị danh sách danh mục đã cập nhật lên màn hình.
8. Xóa danh mục
9. Người quản trị chọn danh mục cần xóa và kích vào icon xóa. Hệ thống hiển thị lên thông báo xác nhận xóa
10. Người quản trị kích vào nút xác nhận xóa danh mục. Hệ thống xóa danh mục trong bảng category và hiển thị thông báo xóa thành công lên màn hình.
11. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị một thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Người quản trị đăng nhập thành công.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

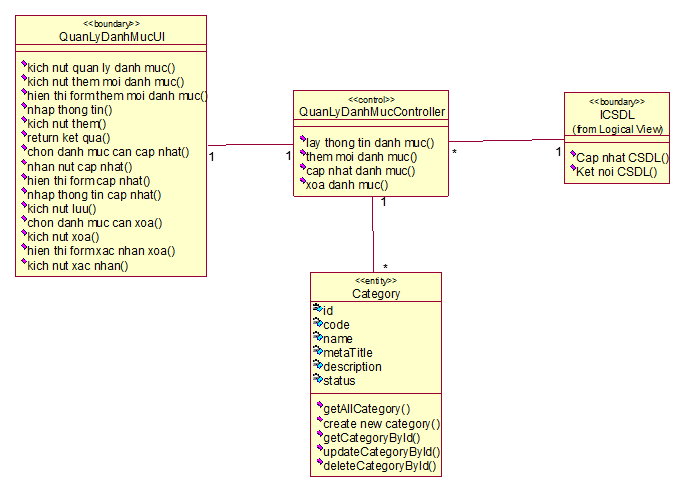
Không có.

c) Biểu đồ trình tự



*Hình 18: Biểu đồ trình tự use case quản lý danh mục*

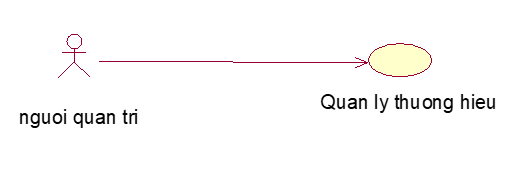
d) Biểu đồ lớp phân tích



*Hình 19: Biểu đồ lớp phân tích use case quản lý danh mục*

7.1.4 Use case quản lý thương hiệu

a) Biểu đồ use case:



*Hình 20: Biểu đồ use case quản lý thương hiệu*

b) Đặc tả use case:

Use case này cho phép người quản trị quản lý các thông tin về thương hiệu sản phẩm trên hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút quản lý thương hiệu. Hệ thống lấy thông tin các thương hiệu gồm id, name, metaTitle, description hiển thị lên màn hình.
2. Thêm mới thương hiệu
3. Người quản trị ấn vào nút thêm mới. Hệ thống hiển thị form thêm mới thương hiệu lên màn hình.
4. Người quản trị nhập thông tin thêm mới thương hiệu sau đó kích vào nút thêm. Hệ thống lưu các thông tin vào bảng provider và hiển thị danh sách thương hiệu sau khi thêm mới.
5. Cập nhật thương hiệu
6. Người quản trị chọn thương hiệu cần cập nhật và kích vào icon cập nhật. Hệ thống lấy thông tin danh mục đã chọn theo id và hiển thị form cập nhật thương hiệu lên màn hình.
7. Người quản trị nhập thông tin cần cập nhật và ấn vào nút lưu. Hệ thống lưu thông tin thương hiệu đã cập nhật vào bảng provider và hiển thị danh sách thương đã cập nhật lên màn hình.
8. Xóa thương hiệu
9. Người quản trị chọn thương hiệu cần xóa và kích vào icon xóa. Hệ thống hiển thị lên thông báo xác nhận xóa
10. Người quản trị kích vào nút xác nhận xóa thương hiệu. Hệ thống xóa thương hiệu trong bảng provider và hiển thị thông báo xóa thành công lên màn hình.
11. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị một thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Người quản trị đăng nhập thành công.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

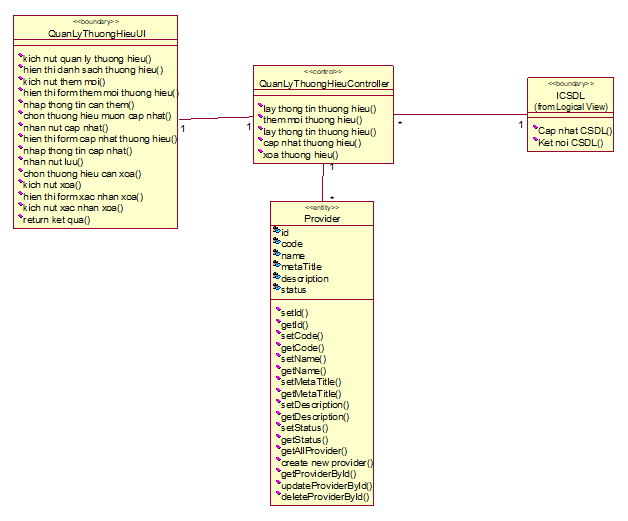
Không có.

c) Biểu đồ trình tự



*Hình 21: Biểu đồ trình tự use case quản lý thương hiệu*

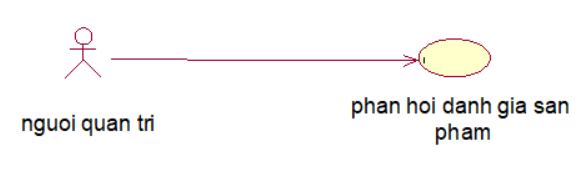
d) Biểu đồ lớp phân tích



*Hình 22: Biểu đồ lớp phân tích use case quản lý thương hiệu*

7.1.5 Use case phản hồi đánh giá sản phẩm

a) Biểu đồ use case:



*Hình 23: Biểu đồ use case phản hồi đánh giá sản phẩm*

b) Đặc tả use case:

Use case này cho phép người quản trị phản hồi đánh giá sản phẩm trên hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút phản hồi đánh giá sản phẩm. Hệ thống lấy thông tin các bài đánh giá gồm id, content, rating, createDate, productId hiển thị lên màn hình.
2. Người quản trị kích chọn một bài đánh giá để phản hồi. Hệ thống lấy thống tin của bài đánh giá theo id và hiển thị lên màn hình.
3. Người quản trị kích vào nút phản hồi. Hệ thống hiển thị một form nhập nội dung phản hồi lên màn hình. Người quản trị nhập thông tin phản hồi sau đó kích vào nút gửi. Hệ thống thêm thông tin phản hồi vào bảng review sau đó gửi thông báo xác nhận thành công lên màn hình.
4. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị một thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Người quản trị đăng nhập thành công.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

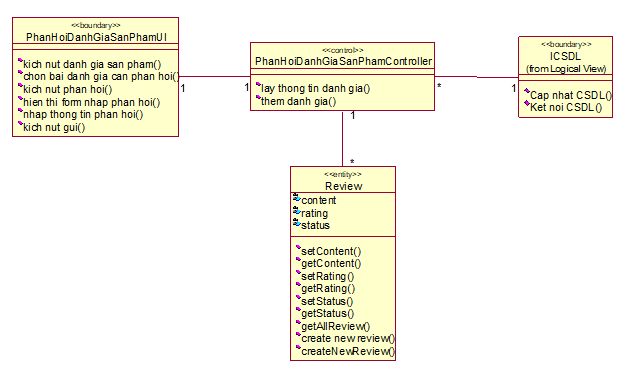
Không có.

c) Biểu đồ trình tự



*Hình 24: Biểu đồ trình tự use case phản hồi đánh giá sản phẩm*

d) Biểu đồ lớp phân tích



*Hình 25: Biểu đồ lớp phân tích use case phản hồi đánh giá sản phẩm*

7.1.6 Use case quản lý đơn hàng

a) Biểu đồ use case:



*Hình 26: Biểu đồ use case quản lý đơn hàng*

b) Đặc tả use case:

Use case này cho phép người quản trị quản lý đơn hàng trên hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút quản lý đơn hàng. Hệ thống lấy thông tin trong bảng order gồm id, customerName, customerAddress, customerPhone, email, totalMoney và các thông tin trong bảng orderDetail gồm id, quantity, size, color, price hiển thị lên màn hình.
2. Cập nhật đơn hàng
3. Người quản trị chọn một đơn hàng cần cập nhật. Hệ thống lấy thông tin đơn hàng và chi tiết đơn hàng theo id sau đó hiển thị lên màn hình.
4. Người quản trị nhập thông tin cập nhật đơn hàng. Hệ thống cập nhật bảng order và orderDetail sau đó gửi thông báo lên màn hình.
5. Xóa đơn hàng
6. Người quản trị chọn đơn hàng cần xóa và kích vào icon xóa. Hệ thống hiển thị lên thông báo xác nhận xóa.
7. Người quản trị kích vào nút xác nhận xóa đơn hàng. Hệ thống xóa thông tin đơn hàng và chi tiết đơn hàng trong bảng order và orderDetail sau đó hiển thị thông báo xóa thành công lên màn hình.

4) Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị một thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Người quản trị đăng nhập thành công.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

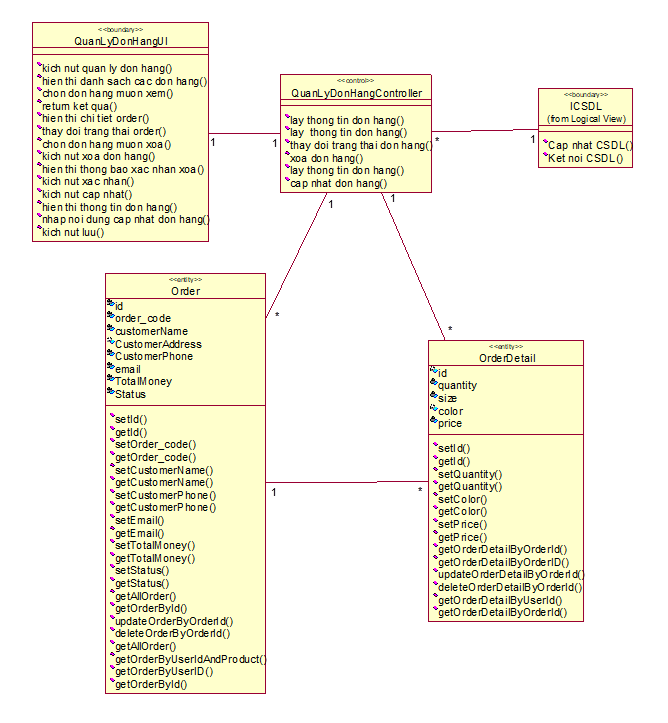
Không có.

c) Biểu đồ trình tự



*Hình 27: Biểu đồ trình tự use case quản lý đơn hàng*

d) Biểu đồ lớp phân tích



*Hình 28: Biểu đồ lớp phân tích use quản lý đơn hàng*

7.1.7 Use case quản lý thông tin bảo hành

a) Biểu đồ use case:



*Hình 29: Biểu đồ use case quản lý bảo hành*

b) Đặc tả use case:

Use case này cho phép người quản trị quản lý thông tin bảo hành sản phẩm trên hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút bảo hành sản phẩm. Hệ thống lấy thông tin trong bảng warranty gồm id, customer\_name, email, phone, product\_name, product\_price, date\_start, date\_end hiển thị lên màn hình.
2. Thêm mới thông tin bảo hành
3. Người quản trị kích chọn vào nút thêm mới. Hệ thống hiển thị form thêm mới.
4. Người quản trị nhập thông tin thêm mới bảo hành sản phẩm sau đó kích nút thêm. Hệ thống thêm thông tin bảo hành vào bảng warranty sau đó hiển thị lên thông báo thêm mới thành công.
5. Cập nhật thông tin bảo hành
6. Người quản trị chọn một bảo hành cần cập nhật sau đó kích vào icon cập nhật. Hệ thống lấy thông tin bảo hành theo id sau đó hiển thị lên màn hình.
7. Người quản trị nhập thông tin cập nhật bảo hành sau đó bấm nút lưu. Hệ thống cập nhật bảng warranty sau đó gửi thông báo cập nhật thành công lên màn hình.
8. Xóa thông tin bảo hành
9. Người quản trị chọn bảo hành cần xóa và kích vào icon xóa. Hệ thống hiển thị lên thông báo xác nhận xóa.
10. Người quản trị kích vào nút xác nhận xóa bảo hành. Hệ thống xóa thông tin bảo hành trong bảng warranty sau đó hiển thị thông báo xóa thành công lên màn hình.

4) Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị một thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Người quản trị đăng nhập thành công.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

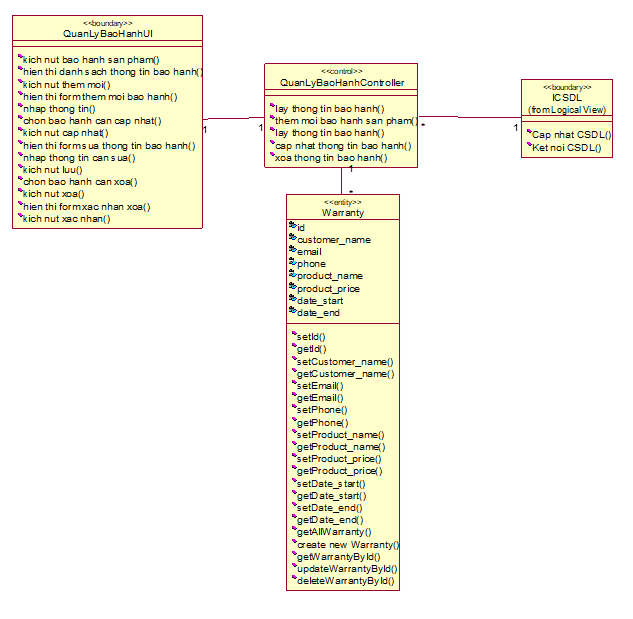
Không có.

c) Biểu đồ trình tự



*Hình 30: Biểu đồ trình tự use case quản lý bảo hành*

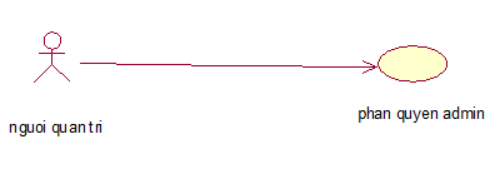
d) Biểu đồ lớp phân tích



*Hình 31: Biểu đồ lớp phân tích use case quản lý bảo hành*

7.1.8 Use case phân quyền admin

a) Biểu đồ use case:



*Hình 32: Biểu đồ use case phân quyền admin*

b) Đặc tả use case:

Use case này cho phép người quản trị phân quyền admin.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút phân quyền admin. Hệ thống lấy thông tin tất cả các user trong bảng user gồm id, account, password, fullName, address, phone, status hiển thị lên màn hình.
2. Phân quyền admin

Người quản trị chọn một tài khoản sau đó kích nút thay đổi trạng thái tài khoản và chọn làm admin. Hệ thống cập nhật thông tin tài khoản trong bảng user và hiển thị thông báo lên màn hình.

3) Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị một thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Người quản trị đăng nhập thành công.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

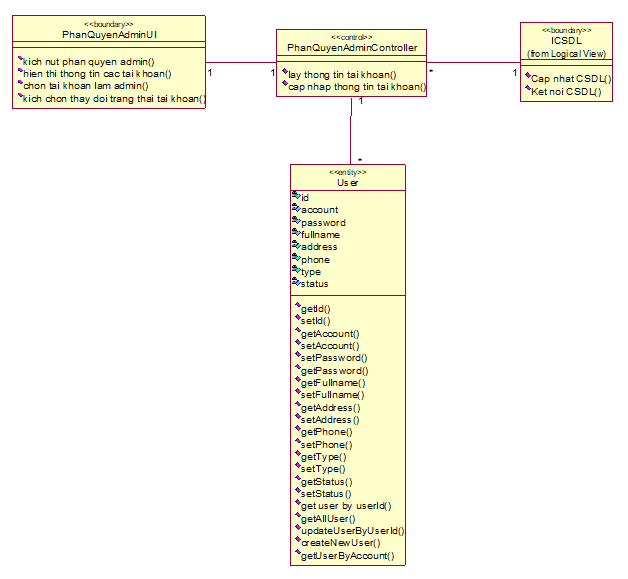
Không có.

c) Biểu đồ trình tự



*Hình 33: Biểu đồ trình tự use case phân quyền admin*

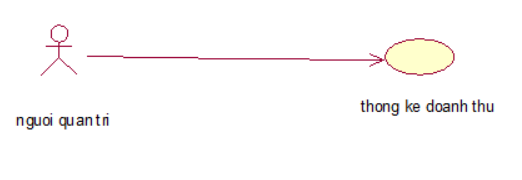
d) Biểu đồ lớp phân tích



*Hình 34: Biểu đồ lớp phân tích use case phân quyền admin*

7.1.9 Use case thống kê doanh thu

a) Biểu đồ use case:



*Hình 35: Biểu đồ use case thống kê doanh thu*

b) Đặc tả use case:

Use case này cho phép người quản trị thống kê doanh thu từ hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút thống kê doanh thu. Hệ thống sẽ lấy dữ liệu từ bảng order và hiển thị biểu đồ thống kê doanh thu theo tháng và theo ngày. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị một thông báo lỗi và use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Người quản trị đăng nhập thành công.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

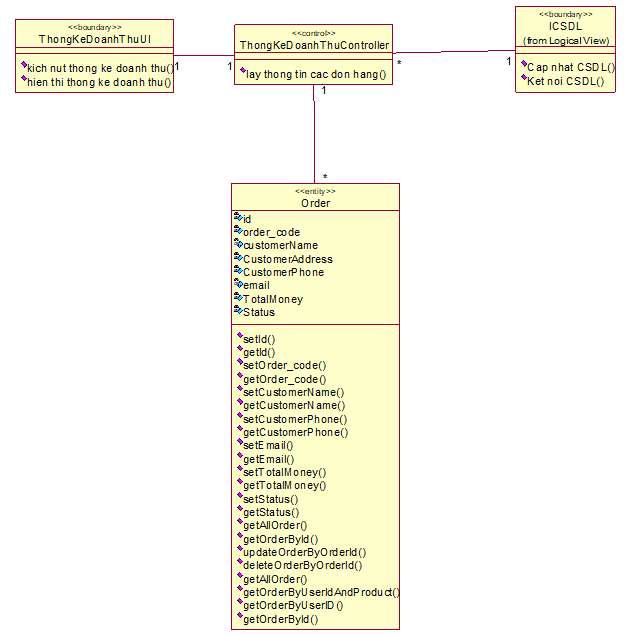
Không có.

c) Biểu đồ trình tự



*Hình 36: Biểu đồ trình tự use case thống kê doanh thu*

d) Biểu đồ lớp phân tích



*Hình 37: Biểu đồ lớp phân tích use case thống kê doanh thu*

7.2 Đặc tả usecase phía font-end

7.2.1 Use case đăng ký

a) Biểu đồ use case:



*Hình 38: Biểu đồ use case đăng ký*

b) Đặc tả use case:

Use case này cho phép khách hàng đăng ký tài khoản.

* Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị bấm vào nút “Đăng ký”.
2. Hệ thống yêu cầu khách hàng nhập các thông tin bao gồm: Tên tài khoản, họ tên, địa chỉ, mật khẩu.
3. Người quản trị nhập thông tin và bấm nút “Đăng ký”
4. Hệ thống thêm thông tin vào bảng user và hiển thị lên thông báo đăng ký tài khoản thành công.
5. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tài khoản đã tồn tại: tại bước 3 trong luồng cơ bản nếu khách hàng nhập tên tài khoản đã tồn tại trong bảng user thì hệ thống hiển thị lên thông báo tài khoản đã tồn tại. Use case kết thúc
2. Đăng ký: tại bước 3 trong luồng cơ bản, nếu người dùng bấm nút “Đăng nhập” use case kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi, use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:
* Điểm mở rộng:

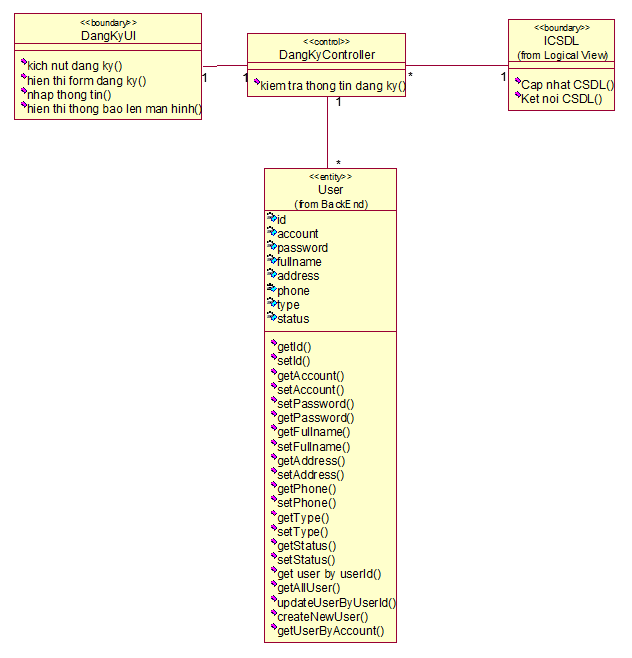
Không có.

c) Biểu đồ trình tự



*Hình 39: Biểu đồ trình tự use case đăng ký*

d) Biểu đồ lớp phân tích



*Hình 40: Biểu đồ lớp phân tích use case đăng ký*

7.2.2 Use case Đăng nhập

a) Biểu đồ use case:



*Hình 41: Biểu đồ use case đăng nhập*

b) Đặc tả use case:

Use case này cho phép khách hàng đăng nhập để xác định quyền truy cập vào hệ thống.

* Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng bấm vào nút “Đăng nhập” trên góc phải màn hình.
2. Hệ thống yêu cầu khách hàng nhập các thông tin bao gồm: Tên tài khoản và mật khẩu.
3. Khách nhập thông tin và bấm nút “Đăng nhập”
4. Hệ thống xác thực tài khoản và mật khẩu rồi chuyển hướng sang trang chủ.
5. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Sai tài khoản hoặc mật khẩu: tại bước 3 trong luồng cơ bản nếu khách hàng nhập tài khoản hoặc mật khẩu sai, hệ thống hiển thị thông báo lỗi. Người dùng có thể nhập lại hoặc bỏ qua thao tác. Use case kết thúc
2. Đăng ký: tại bước 3 trong luồng cơ bản, nếu người dùng bấm nút “Đăng ký” use case kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi, use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

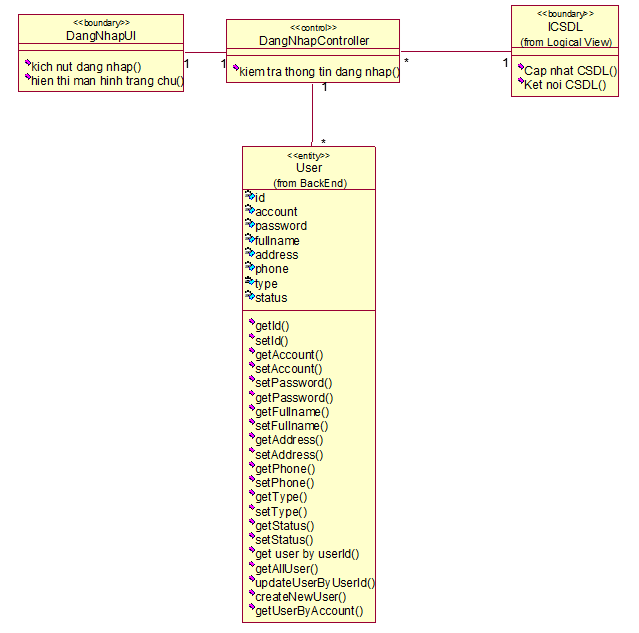
Không có.

c) Biểu đồ trình tự



*Hình 42: Biểu đồ trình tự use case đăng nhập*

d) Biểu đồ lớp phân tích



*Hình 43: Biểu đồ lớp phân tích use case đăng nhập*

7.2.3 Use case xem sản phẩm

a) Biểu đồ use case:



*Hình 44: Biểu đồ use case xem sản phẩm*

b) Đặc tả use case:

Use case này cho phép khách hàng xem sản phẩm.

* Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng bấm vào một sản phẩm bất kỳ.
2. Hệ thống lấy thông tin của sản phẩm theo id và hiển thị lên màn hình các thông tin tên sản phẩm, giá, hình ảnh minh họa, số lượng có.
3. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi, use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

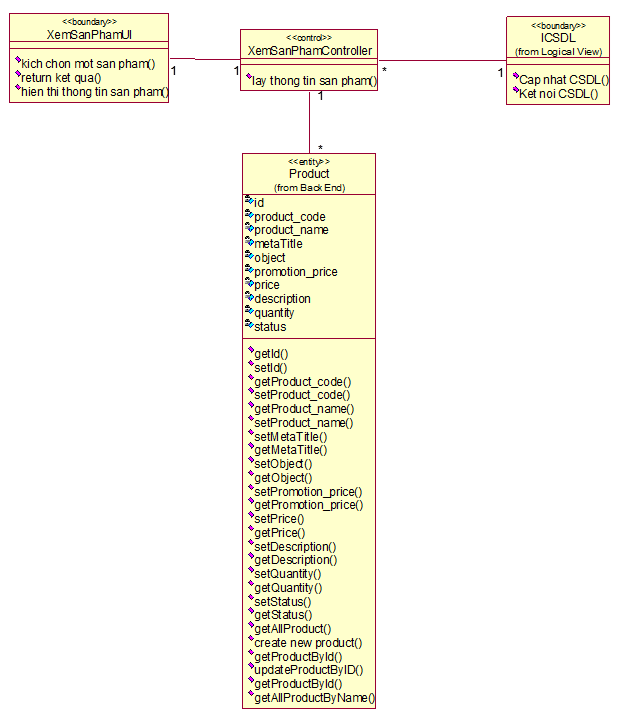
Không có.

c) Biểu đồ trình tự



*Hình 45: Biểu đồ trình tự use case xem sản phẩm*

d) Biểu đồ lớp phân tích



*Hình 46: Biểu đồ lớp phân tích use case xem sản phẩm*

7.2.4 Use case tìm kiếm

a) Biểu đồ use case:



*Hình 47: Biểu đồ use case tìm kiếm*

b) Đặc tả use case:

Use case này cho phép khách hàng tìm kiếm sản phẩm.

* Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng nhập thông tin sản phẩm vào ô tìm kiếm và kích vào icon search.
2. Hệ thống lấy thông tin của các sản phẩm theo thông tin mà khách hàng nhập trong ô tìm kiếm và hiển thị lên màn hình các thông tin tên sản phẩm, giá, hình ảnh minh họa, số lượng có.
3. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi, use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

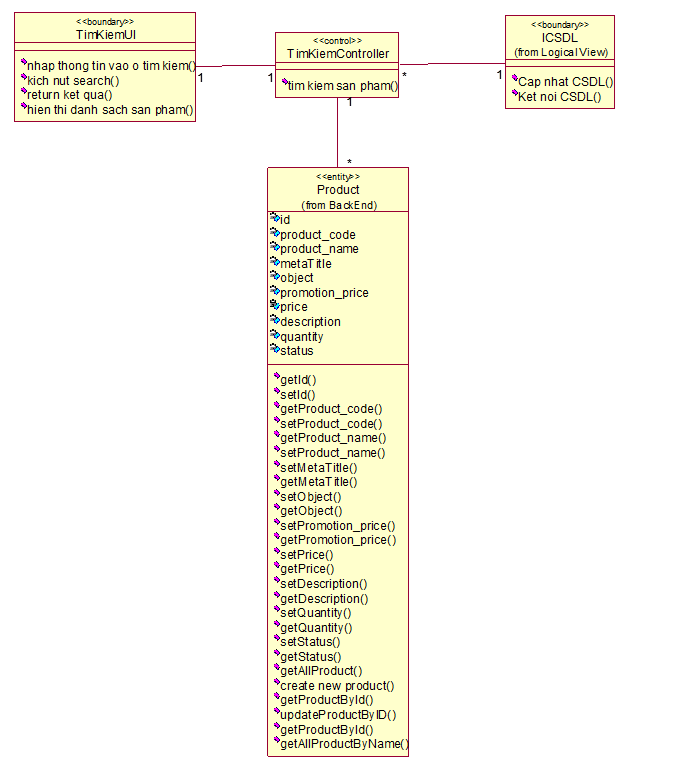
Không có.

c) Biểu đồ trình tự



*Hình 48: Biểu đồ trình tự use case tìm kiếm*

d) Biểu đồ lớp phân tích



*Hình 49: Biểu đồ lớp phân tích use case tìm kiếm*

7.2.5 Use case quản lý giỏ hàng

a) Biểu đồ use case:



*Hình 50: Biểu đồ use case quản lý giỏ hàng*

b) Đặc tả use case:

Use case này cho phép khách hàng quản lý giỏ hàng.

* Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào icon giỏ hàng trên trang chủ.
2. Hệ thống lấy thông tin của các sản phẩm trong giỏ hàng của người dùng và hiển thị lên màn hình.
3. Cập nhật số lượng sản phẩm trong giỏ hàng:

Khách hàng kích chọn một sản phẩm trong giỏ hàng và kích vào icon tăng hoặc giảm số lượng sản phẩm. Hệ thống sẽ cập nhật số lượng sản phẩm trong giỏ hàng.

1. Xóa sản phẩm trong giỏ hàng:

4.1) Khách hàng chọn một sản phẩm để xóa sau đó kích chọn vào icon xóa. Hệ thống hiển thị lên thông báo xác nhận xóa sản phẩm.

4.2) Khách hàng kích chọn xác nhận xóa. Hệ thống xóa sản phẩm trong giỏ hàng và hiển thị thông báo xóa thành công.

1. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi, use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Không có

* Điểm mở rộng:

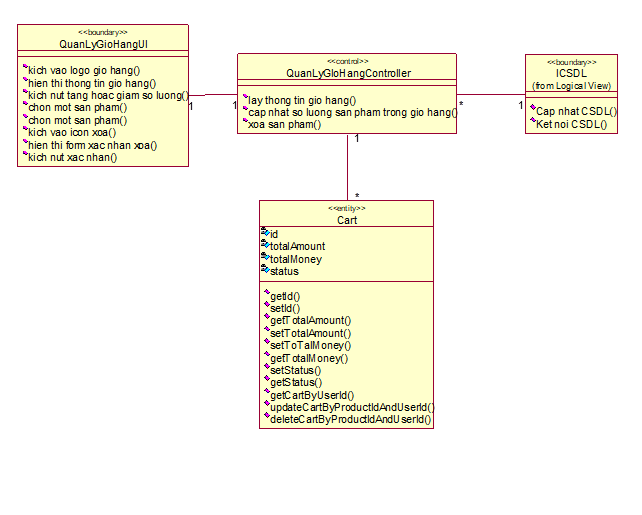
Không có.

c) Biểu đồ trình tự



*Hình 51: Biểu đồ trình tự use case quản lý giỏ hàng*

d) Biểu đồ lớp phân tích



*Hình 52: Biểu đồ lớp phân tích use case quản lý giỏ hàng*

7.2.6 Use case đánh giá sản phẩm

a) Biểu đồ use case:



*Hình 53: Biểu đồ use case đánh giá sản phẩm*

b) Đặc tả use case:

Use case này cho phép khách hàng đánh giá sản phẩm.

* Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào mục chi tiết đơn hàng và kích vào nút đánh giá sản phẩm
2. Hệ thống hiển thị thông tin form đánh giá sản phẩm lên màn hình.
3. Khách hàng nhập thông tin đánh giá sản phẩm và bấm nút gửi. Hệ thống thêm thông tin đánh giá vào bảng review và hiển thị thông báo lên màn hình
4. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi, use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Khách hàng phải đăng nhập vào hệ thống.

* Điểm mở rộng:

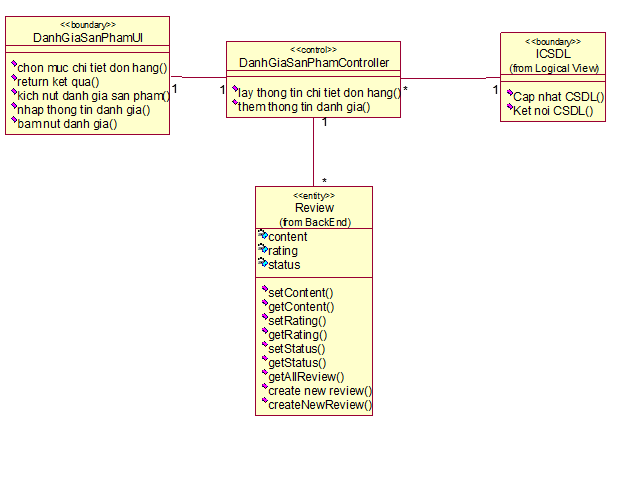
Không có.

c) Biểu đồ trình tự



*Hình 54: Biểu đồ trình tự use case đánh giá sản phẩm*

d) Biểu đồ lớp phân tích



*Hình 55: Biểu đồ lớp phân tích use case đánh giá sản phẩm*

7.2.7 Use case đặt hàng

a) Biểu đồ use case:



*Hình 56: Biểu đồ use case đặt hàng*

b) Đặc tả use case:

Use case này cho phép khách hàng đặt hàng.

* Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào logo giỏ hàng.
2. Hệ thống hiển thị thông tin các sản phẩm trong giỏ hàng.
3. Khách hàng kích vào nút mua hàng
4. Hệ thống hiển thị lên màn hình thông tin về các sản phẩm và tổng tiền của các sản phẩm đó.
5. Khách hàng kích chọn đặt hàng. Hệ thống hiển thị lên thông báo đặt hàng thành công và lưu thông tin vào bảng đơn hàng.
6. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi, use case kết thúc.

* Các yêu cầu đặc biệt:

Không có.

* Tiền điều kiện:

Không có.

* Hậu điều kiện:

Khách hàng phải đăng nhập.

* Điểm mở rộng:

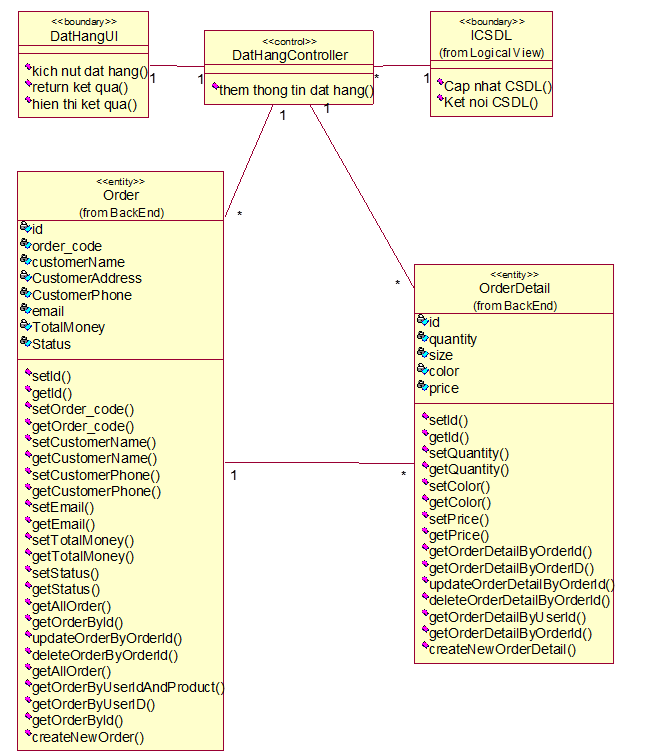
Không có.

c) Biểu đồ trình tự



*Hình 57: Biểu đồ trình tự use case đánh giá sản phẩm*

d) Biểu đồ lớp phân tích



*Hình 58: Biểu đồ lớp phân tích use case đặt hàng*