**1.1 Lý do chọn đề tài**

Hiện nay, ô tô đã trở thành một phương tiện không thể thiếu đối với người dân Việt Nam. Không chỉ đơn thuần là một phương tiện di chuyển, ô tô còn mang lại nhiều lợi ích đáng kể cho cá nhân và xã hội. Sự tiện lợi mà ô tô mang lại giúp người sử dụng tự do lựa chọn lộ trình và thời gian di chuyển mà không cần phụ thuộc vào phương tiện công cộng, qua đó tiết kiệm thời gian và tăng cường sự linh hoạt trong các kế hoạch hàng ngày.

Để đáp ứng nhu cầu mua xe của khách hàng mà không cần phải đến tận nơi, các website bán ô tô online đã ra đời. Xây dựng một trang web bán ô tô không chỉ mở ra cơ hội kinh doanh mà còn là bước tiến quan trọng trong việc đáp ứng xu hướng mua sắm trực tuyến ngày càng gia tăng. Trong thời đại số hóa, người tiêu dùng đang chuyển dần từ việc mua sắm truyền thống tại các đại lý sang việc tìm kiếm và mua sắm trực tuyến. Việc các cửa hàng bán ô tô sở hữu một trang web sẽ mở ra cơ hội tiếp cận lượng khách hàng tiềm năng lớn, bao gồm cả những người không tiện đến cửa hàng truyền thống.

Một trang web bán ô tô mang lại sự tiện lợi và linh hoạt cho cả người mua và người bán. Thay vì phải đến các đại lý ô tô, khách hàng có thể duyệt qua các sản phẩm, so sánh giá cả và tính năng một cách thuận tiện từ nhà. Đối với các nhà cung cấp và đại lý, trang web giúp tiết kiệm chi phí vận hành và quảng bá, đồng thời mở rộng thị trường tiếp cận.

Hơn nữa, việc xây dựng một trang web bán ô tô là cơ hội để tạo ra trải nghiệm mua sắm độc đáo và tương tác cho người dùng. Sử dụng công nghệ hiện đại như hình ảnh 360 độ, video chất lượng cao và các công cụ tương tác như bộ lọc sản phẩm và công cụ tư vấn mua sắm, trang web có thể mang đến một trải nghiệm mua sắm thú vị và hấp dẫn. Điều này không chỉ thu hút khách hàng mà còn tạo ấn tượng tích cực và tăng cường niềm tin vào thương hiệu.

Cuối cùng, việc xây dựng một trang web bán ô tô không chỉ nhằm mục đích bán hàng mà còn xây dựng một cộng đồng và mối quan hệ gắn kết với khách hàng. Bằng cách cung cấp các tính năng như đánh giá, bình luận và cộng đồng trực tuyến, trang web có thể tạo ra một không gian nơi khách hàng chia sẻ kinh nghiệm, ý kiến và niềm đam mê về ô tô. Điều này giúp tạo ra một cộng đồng đam mê và tăng cường sự tương tác giữa thương hiệu và khách hàng.

**1.2 Đối tượng, mục tiêu, phạm vi nghiên cứu**

**1.2.1 Đối tượng nghiên cứu**

* HTML, CSS, Javascript.
* MySQL Database.
* Ngôn ngữ lập trình Java.
* Spring Framework: Spring MVC, Spring Boot.

**1.2.2 Mục tiêu nghiên cứu**

* Tìm hiểu các ngôn ngữ: HTML, CSS, JavaScript, biết cách xây dựng một giao diện website thân thiện và thu hút khách hàng.
* Tìm hiểu về MySQL, biết cách thiết kế database, quản lý dữ liệu.
* Tìm hiểu về ngôn ngữ lập trình Java, biết cách xây dựng các chức năng cho website bằng Java.
* Áp dụng các kiến thức đã tìm hiểu được và được học để xây dựng website bán ô tô CarForYou.

**1.2.3 Phạm vi nghiên cứu**

* Giao diện người dùng (UI): Nghiên cứu và phát triển giao diện người dùng thân thiện và thu hút, bao gồm thiết kế trang chủ, trang sản phẩm, trang thanh toán và các trang liên quan khác. Phạm vi này cũng bao gồm việc tối ưu hóa trải nghiệm người dùng trên các thiết bị di động và máy tính bảng.
* Cơ sở dữ liệu (Database): Thiết kế và triển khai cơ sở dữ liệu để lưu trữ thông tin về các sản phẩm ô tô, thông tin khách hàng, đơn đặt hàng và các dữ liệu liên quan khác. Nghiên cứu này cũng bao gồm việc xây dựng các quan hệ giữa các bảng và tối ưu hóa cấu trúc cơ sở dữ liệu để đảm bảo hiệu suất và độ tin cậy.
* Chức năng và tính năng (Functionality): Phát triển các chức năng cần thiết cho trang web bán ô tô, bao gồm tìm kiếm sản phẩm, đăng ký tài khoản, đặt hàng, thống kê, quản lý tài khoản người dùng, quản lý sản phẩm và các chức năng quản lý hệ thống khác.
* Bảo mật và bảo vệ (Security): Nghiên cứu và triển khai các biện pháp bảo mật để bảo vệ dữ liệu cá nhân của người dùng, đảm bảo an toàn cho giao dịch thanh toán và ngăn chặn các mối đe dọa từ hacker và các cuộc tấn công mạng khác.
* Kiểm thử  và triển khai (Testing and Deployment): Phạm vi này bao gồm việc thử nghiệm các tính năng và chức năng của trang web để đảm bảo chất lượng và sự ổn định trước khi triển khai vào môi trường sản xuất. Nghiên cứu bao gồm việc lên kế hoạch và triển khai trang web vào một môi trường hoạt động thực tế.

**1.2.4 Cấu trúc báo cáo**

- Báo cáo đồ án bao gồm những nội dung chính sau:

* Chương 1. Tổng quan về đề tài.
* Chương 2. Phân tích thiết kế hệ thống.
* Chương 3. Kết quả và kiếm thử hệ thống

# TỔNG QUAN CƠ SỞ LÝ THUYẾT XÂY DỰNG WEBSITE BÁN Ô TÔ CARFORYOU

Chương 1 cung cấp tổng quan về ba thành phần quan trọng trong phát triển ứng dụng Java: ngôn ngữ lập trình Java, Spring Framework, và hệ quản trị cơ sở dữ liệu MySQL. Java là ngôn ngữ lập trình phổ biến với tính năng "Write Once, Run Anywhere", an toàn, dễ học và hỗ trợ đa mô hình lập trình. Spring Framework, với Spring MVC và Spring Boot, hỗ trợ phát triển ứng dụng web theo mô hình MVC và tạo ứng dụng nhanh chóng với cấu hình tự động. MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở, dễ sử dụng, bảo mật cao, đa tính năng và nhanh chóng, nhưng có một số hạn chế về chức năng và độ tin cậy. Chương này đặt nền tảng cho việc phát triển ứng dụng Java hiệu quả và an toàn.

* 1. **Giới thiệu tổng quan về ngôn ngữ lập trình Java**

- Java [4] là một ngôn ngữ lập trình phổ biến và mạnh mẽ được phát triển bởi Sun Microsystems (hiện là một phần của Oracle Corporation) vào những năm đầu của thập kỷ 1990. Java được thiết kế để có thể chạy trên nhiều nền tảng khác nhau mà không cần sửa đổi lại mã nguồn, điều này là nhờ vào mô hình "Write Once, Run Anywhere" (viết một lần, chạy ở mọi nơi) của nó.

- Một số điểm nổi bật của ngôn ngữ lập trình Java:

* Đa nền tảng: Java được thiết kế để có thể chạy trên nhiều hệ điều hành và môi trường máy ảo Java (JVM), bao gồm Windows, macOS, Linux và nhiều hệ điều hành khác.
* Độ an toàn cao: Java có các tính năng bảo mật tích hợp sâu trong ngôn ngữ, bao gồm quản lý bộ nhớ tự động (garbage collection), kiểm soát truy cập vào biến và quản lý ngoại lệ (exception handling), giúp người lập trình xây dựng các ứng dụng an toàn và ổn định.
* Dễ học và sử dụng: Java có cú pháp rõ ràng và dễ hiểu, với một cộng đồng lớn và nhiều tài liệu học trực tuyến, giúp người mới bắt đầu nhanh chóng tiếp cận và làm chủ ngôn ngữ này.
* Hỗ trợ đa mô hình lập trình: Java hỗ trợ nhiều mô hình lập trình khác nhau, bao gồm lập trình hướng đối tượng (OOP), lập trình hàm (functional programming) và lập trình song song (concurrency), giúp người lập trình dễ dàng thiết kế và triển khai các ứng dụng phức tạp.
* Thư viện phong phú: Java đi kèm với một loạt các thư viện tiêu chuẩn và các framework phổ biến như Spring, Hibernate và Apache Struts, giúp giảm thiểu thời gian phát triển và tăng cường hiệu suất của ứng dụng.
* Ứng dụng đa dạng: Java được sử dụng rộng rãi trong nhiều lĩnh vực, bao gồm phát triển ứng dụng web, ứng dụng di động (Android), ứng dụng máy chủ, trò chơi điện tử và nhiều ứng dụng khác.
  1. **Spring FrameWork: Spring MVC và Spring Boot**

+ Spring Framework [7] là một bộ công cụ phát triển ứng dụng Java mạnh mẽ, được sử dụng rộng rãi trong việc xây dựng các ứng dụng doanh nghiệp. Nó cung cấp các module và tính năng để giảm thiểu sự phức tạp của việc phát triển ứng dụng, từ việc quản lý Dependency Injection, xử lý ngoại lệ, tới tích hợp các công nghệ như JDBC, JPA, REST và nhiều hơn nữa.

+ Tổng quan về Spring Framework bao gồm:

- Spring MVC (Model-View-Controller):

* Spring MVC là một phần của Spring Framework, được sử dụng để phát triển các ứng dụng web dựa trên mô hình MVC.
* Nó cung cấp một cách tiếp cận linh hoạt và rõ ràng để phát triển các ứng dụng web, tách biệt logic của ứng dụng thành các thành phần riêng biệt như Model, View và Controller.
* Spring MVC tích hợp dễ dàng với các công nghệ khác như Hibernate, JPA, và thường được sử dụng trong việc xây dựng các ứng dụng web với Java.

- Spring Boot:

* Spring Boot là một dự án con của Spring Framework, nhằm mục đích giúp tạo ra các ứng dụng Java nhanh chóng, dễ dàng và có thể tự động cấu hình.
* Nó loại bỏ sự phức tạp trong việc cấu hình và triển khai ứng dụng bằng cách cung cấp các cấu hình mặc định thông minh và tự động quản lý các dependencies.
* Spring Boot cung cấp một số tính năng quan trọng như tích hợp server nhúng (embedded server), cung cấp các Starter POMs (Project Object Model) để nhanh chóng bắt đầu các dự án và tạo ra các ứng dụng độc lập tự động (self-contained executable)
  1. **MySQL**

MySQL [5] là một hệ thống quản trị cơ sở dữ liệu quan hệ mã nguồn mở (RDBMS - Relational Database Management System) hoạt động theo mô hình client-server. MySQL thường được sử dụng cùng với Apache và PHP trong việc phát triển ứng dụng web. MySQL quản lý dữ liệu thông qua các cơ sở dữ liệu, mỗi cơ sở dữ liệu có thể chứa nhiều bảng quan hệ lưu trữ dữ liệu. SQL là ngôn ngữ chuẩn cho việc truy vấn và quản lý cơ sở dữ liệu quan hệ, và MySQL sử dụng cú pháp SQL để thực hiện các thao tác này.

***Ưu điểm của MySQL:***

* Dễ sử dụng: MySQL là cơ sở dữ liệu tốc độ cao, ổn định, dễ sử dụng và hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.
* Độ bảo mật cao: MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên Internet khi sở hữu nhiều nhiều tính năng bảo mật thậm chí là ở cấp cao.
* Đa tính năng: MySQL hỗ trợ rất nhiều chức năng SQL được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ cả trực tiếp lẫn gián tiếp.
* Khả năng mở rộng và mạnh mẽ: MySQL có thể xử lý rất nhiều dữ liệu và hơn thế nữa nó có thể được mở rộng nếu cần thiết.
* Nhanh chóng: Việc đưa ra một số tiêu chuẩn cho phép MySQL để làm việc rất hiệu quả và tiết kiệm chi phí, làm tăng tốc độ thực thi.

***Nhược điểm của MySQL:***

* Giới hạn chức năng: MySQL vẫn còn một số hạn chế về chức năng, có thể không đáp ứng đủ nhu cầu của một vài ứng dụng phức tạp.
* Hạn chế về dung lượng: Khi số lượng bản ghi tăng lên, việc truy xuất dữ liệu có thể trở nên khó khăn hơn. Trong những trường hợp này, cần áp dụng các biện pháp như phân chia cơ sở dữ liệu ra nhiều server hoặc tạo bộ nhớ đệm (cache) cho MySQL để tăng tốc độ truy xuất dữ liệu.
* Độ tin cậy hạn chế: Cách MySQL xử lý một số chức năng cụ thể khiến nó kém tin cậy hơn so với một số hệ quản trị cơ sở dữ liệu quan hệ khác.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

Chương 2 tập trung vào việc xây dựng website bán ô tô CarForYou, bao gồm các yêu cầu nghiệp vụ và chức năng cụ thể của hệ thống. Phía người dùng, website cho phép duyệt và tìm kiếm sản phẩm, xem chi tiết, thêm vào giỏ hàng, đặt hàng, và chat trực tuyến với nhân viên tư vấn. Người dùng cần đăng ký và đăng nhập để thực hiện các chức năng này. Phía người quản trị, hệ thống hỗ trợ quản lý sản phẩm, danh mục, đơn hàng, tài khoản, và bài viết, cùng với các tính năng cập nhật trạng thái đơn hàng và thống kê doanh số. Chương này cũng mô tả các yêu cầu phi chức năng như giao diện thân thiện, bảo mật thông tin và hiệu suất hoạt động. Cuối cùng, các use case được mô tả chi tiết, minh họa cách thức người dùng và quản trị viên tương tác với hệ thống, kèm theo các biểu đồ trình tự và biểu đồ lớp để hỗ trợ việc thiết kế và phát triển.

## 2.1 Mô tả bài toán

- Bài toán xây dựng website bán ô tô CarForYou với một số yêu cầu nghiệp vụ như sau:

* Về phía người dùng: Website cho phép người dùng có thể xem danh sách sản phẩm theo danh mục, model, nơi sản xuất hoặc theo giá, cho phép tìm kiếm sản phẩm bằng từ khóa. Khách hàng có thể xem chi tiết từng sản phẩm, thêm sản phẩm vào giỏ hàng. Bên cạnh đó, người dùng cần phải đăng ký tài khoản và đăng nhập để có thể đặt hàng và theo dõi đơn hàng đã đặt của mình. Khách hàng cần cung cấp đầy đủ thông tin cho việc đặt hàng gồm: họ tên, email, số điện thoại, địa chỉ giao hàng…Website có chức năng chat online cho phép người dùng nhắn tin trong trường hợp có bất kỳ câu hỏi thắc mắc gì để nhân viên tư vấn giải đáp.
* Về phía người quản trị: Hệ thống cho phép họ quản lý các sản phẩm, danh mục hàng, đơn hàng, tài khoản, bài viết, nhân viên, thống kê,….Người quản trị có thể cập nhật trạng thái đơn hàng, in hóa đơn, thống kê doanh số.

## 2.2 Các yêu cầu chức năng

***Yêu cầu chức năng đối với người dùng:***

* Đăng ký: Người dùng phải nhập đầy đủ thông tin để có thể đăng ký thành công tài khoản để có thể mua sắm trên website.
* Đăng nhập: Khi người dùng nhập tên tài khoản và mật khẩu đúng với tài khoản đã đăng ký thì có thể đăng nhập vào hệ thống.
* Xem bài viết: Người dùng có thể vào “Blog” trên thanh menu để có thể xem các bài viết được đăng tải.
* Tìm kiếm: Người dùng hay khách ghé thăm có thể tìm kiếm sản phẩm bằng cách nhập từ khóa vào ô tìm kiếm hoặc có thể chọn những tiêu chí mà mình muốn trên màn hình để tìm kiếm.
* Xem danh sách sản phẩm: Người dùng nhấp chuột vào phần tên của mình trên màn hình và chọn “My Vehicles” xem danh sách toàn bộ sản phẩm.
* Xem chi tiết sản phẩm: Người dùng có thể nhấp chuột vào một sản phẩm trên danh sách để có thể xem chi tiết về tên, giá, số lượng còn, hình ảnh, mô tả của sản phẩm đó.
* Lọc sản phẩm: Người dùng có thể lọc sản phẩm theo bộ sưu tập, theo danh mục và theo khoảng giá.
* Xem thông tin tài khoản: Người dùng có thể xem thông tin chi tiết của tài khoản đã đăng nhập.
* Xem thông tin của shop: Người dùng có thể kick vào “About us ” để xem thông tin của cửa hàng
* Chat online: Người dùng có thể nhấp chuột vào biểu tượng chat trên màn hình để tiến hành vào trang chat và chat online .Trong đó luôn có nhân viên của shop để giải đáp.
* Gửi mail và gọi điện: Người dùng có thể kick vào link gửi Email hoặc số điện thoại của shop trên màn hình để nhắn tin trực tiếp đến shop. Người dùng phải đăng nhập trước khi thực hiện thao tác này
* Thêm sản phẩm vào giỏ và đặt hàng: Người dùng có  thể xem chi tiết từng sản phẩm, thêm sản phẩm vào giỏ và tiến hành đặt hàng.

***Yêu cầu đối với người quản trị:***

* Người quản trị có thể quản lý thêm, sửa xóa sản phẩm, danh mục, đơn hàng, tài khoản , bài viết, nhân viên , thống kê,….
* Người quản trị có thể quản lý đơn hàng, cập nhật trạng thái đơn hàng, lập và in hóa đơn, báo cáo thống kê doanh thu.

## 2.3. Các yêu cầu phi chức năng

Giao diện:

* Giao diện tiếng Việt.
* Cung cấp giao diện đơn giản, gần gũi, trực quan về dễ sử dụng đối với người dùng.

Bảo mật:

* Thực hiện bảo mật bằng các cách mã hóa các thông tin cần bảo mật của người dùng.
* Phân quyền truy cập, chỉ cho phép người dùng truy cập những chức năng được cho phép.
* Kiểm tra những dữ liệu người dùng nhập vào.

Tốc độ xử lý và thời gian hoạt động:

* Tốc độ xử lý nhanh chóng, thực hiện loading khi lấy hoặc ghi dữ liệu.
* Có thể hoạt động hiệu quả 24/24.

## 2.4 Mô hình hóa chức năng

### 2.4.1. Danh sách các Actor

|  |  |  |
| --- | --- | --- |
| **STT** | **Actor** | **Ghi chú** |
| 1 | Khách hàng ghé thăm | Khách hàng truy cập vào website nhưng chưa đăng nhập. |
| 2 | Khách hàng | Khách hàng đã đăng nhập tài khoản. |
| 3 | Admin | Quản trị viên quản lý toàn bộ hệ thống. |
| 4 | Hệ thống thanh toán online | Cho phép người dùng thanh toán online. |
| 5 | Chat online | Cho phép người dùng có thể chat online. |

Bảng 2.1: Danh sách Actor

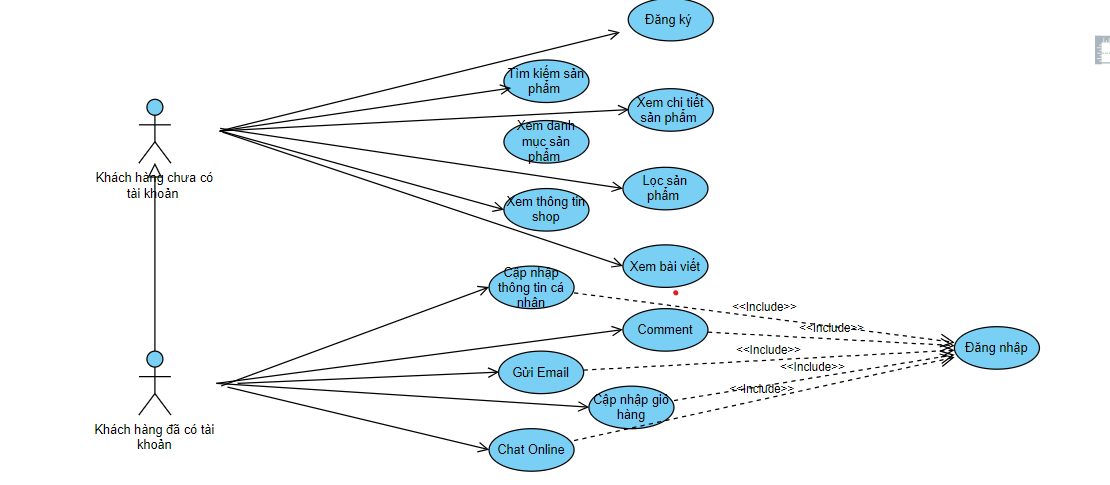
Ngoài các Actor kể trên ra, còn một Actor nữa đó là Actor CSDL. Actor này tác động đến tất cả hoạt động của hệ thống. Để đơn giản hóa các biểu đồ thì sẽ không đưa Actor này vào trong các biểu đồ .

### 2.4.2 Biểu đồ use case

****

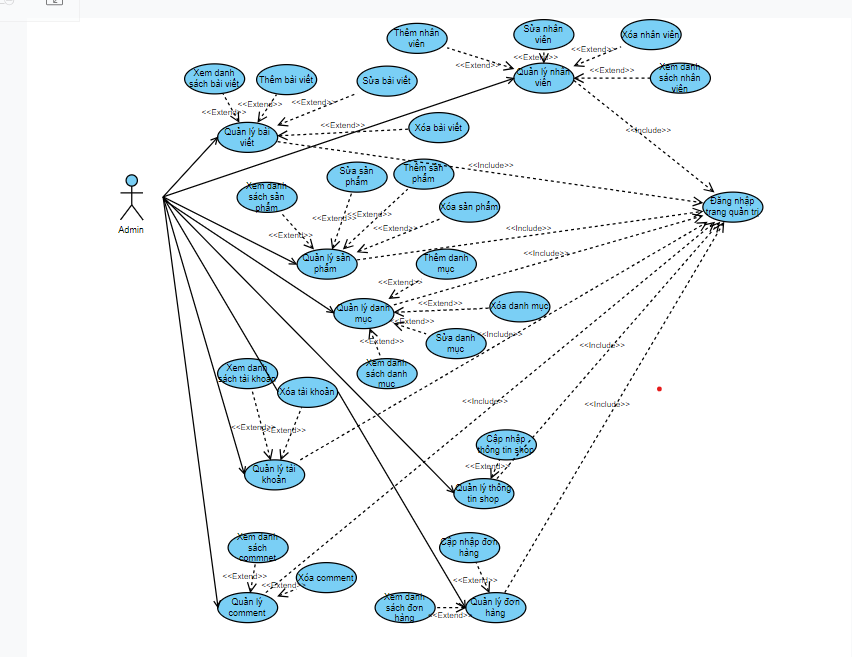
Hình 2.1:Biểu đồ Use Case tổng quan

### 2.4.3 Biểu đồ use case người dùng

****

Hình 2.2:Biểu đồ Use Case phía người dùng

### 2.4.4 Biểu đồ use case Admin

****

Hình 2.3:Biểu đồ Use Case phía Admin

## 2.5 Mô tả chi tiết các Use Case

### 2.5.1 Use case tìm kiếm sản phẩm

Use case này cho phép người dùng tìm kiếm sản phẩm của mình.

Luồng sự kiện:

* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng nhập thông tin mình muốn tìm kiếm vào trong ô “Search” và nhập vào biểu tượng Search . Hệ thống sẽ lấy ra những thông tin của sản phẩm mà có từ khóa trùng với từ khóa mà khách hàng tìm kiếm gồm tên sản phẩm, hình ảnh, giá tiền, model, nơi sản xuất, mô tả. Use case kết thúc.

* Luồng rẽ nhánh:

1. Tại tất cả các bước trong luồng cơ bản nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị một thông báo lỗi và 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.

### 2.5.2 Use case cập nhập giỏ hàng

Use case này cho phép người dùng quản lý giỏ hàng của mình.

Luồng sự kiện:

* Luồng cơ bản:

**1.** Use case này bắt đầu khi người dùng nhấp chuột vào “My Vehicles ” trên thanh menu của màn hình giao diện. Hệ thống sẽ chuyển sang màn hình thông tin giỏ hàng của người dùng gồm các sản phẩm, số lượng .

1. Khi người dùng xóa một sản phẩm khỏi giỏ hàng hệ thống sẽ cập nhật lại.
2. Khi người dùng nhấp chuột vào nút “Proceed ” để xử lý đơn hàng thì hệ thống sẽ chuyển sang màn hình xử lý đơn hàng. Use case kết thúc.

* Luồng rẽ nhánh:

**1.** Tại tất cả các bước trong luồng cơ bản nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị một thông báo lỗi và use case kết thúc.

Các yêu cầu đặc biệt:

Không có.

Tiền điều kiện:

Người dùng phải đăng nhập.

Hậu điều kiện:

Nếu use case kết thúc thành công bảng CartItem sẽ được cập nhật.

Điểm mở rộng:

Không có.

### 2.5.3 Use case xem chi tiết sản phẩm

Mô tả vắn tắt:

Use case này cho phép người dùng xem thông tin chi tiết của sản phẩm.

Luồng sự kiện:

* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng nhấp chuột vào một sản phẩm trên trang chủ. Hệ thống sẽ chuyển hướng sang màn hình thông tin chi tiết của sản phẩm đó bao gồm tên sản phẩm, hình ảnh, giá tiền, models, nơi sản xuất, mô tả lấy. Use case kết thúc.

* Luồng rẽ nhánh:

1. Tại tất cả các bước trong luồng cơ bản nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị thông báo lỗi và 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.

**2.5.4 Use case xem bài viết**

Use case này cho phép người dùng xem bài viết của shop

Luồng sự kiện:

* Luồng cơ bản:

**1.** Use case này bắt đầu khi người dùng kick vào “Blog” trên thanh menu . Hệ thống sẽ hiển thị ra bài viết shop bao gồm tên bài viết , nội dung bài viết , thời gian đăng. Use case kết thúc.

* Luồng rẽ nhánh:

**1.** Tại tất cả các bước trong luồng cơ bản nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị một thông báo lỗi và 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.

### 2.5.5 Use case cập nhập thông tin cá nhân

Use case này cho phép người dùng xem cập nhật thông tin cá nhân.

Luồng sự kiện:

* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng nhấp chuột vào nút “Profile Setting” trên thanh menu của hệ thống. Hệ thống sẽ chuyển sang màn hình thông tin tài khoản và hiển thị lên những thông tin gồm ảnh đại diện, tên đăng nhập, họ tên, email, số điện thoại, địa chỉ.
2. Khi người dùng nhập mới các thông tin tài khoản và nhấp chuột vào nút “Lưu” thay đổi, hệ thống sẽ cập nhật thông tin thay đổi của người dùng lên cơ sở dữ liệu. Use case kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 luồng cơ bản, nếu người dùng tên đăng nhập trùng với 1 người dùng khác trên hệ thống thì hệ thống sẽ báo lỗi và yêu cầu nhập lại.
2. Tại tất cả các bước trong luồng cơ bản nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị một thông báo lỗi và use case kết thúc.

Các yêu cầu đặc biệt:

Không có.

Tiền điều kiện:

Người dùng phải đăng nhập tài khoản.

Hậu điều kiện:

Nếu use case kết thúc thành công thì thông tin người dùng sẽ được cập nhật vào bảng users.

Điểm mở rộng:

Không có.

### 2.5.6 Use case đăng nhập

Mô tả vắn tắt:

Use case này cho phép người dùng (người dùng và quản trị) đăng nhập vào hệ thống.

Luồng sự kiện:

* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng kick vào “Login/SignIn” trên thanh công cụ để đăng nhập vào hệ thống. Hệ thống sẽ hiển thị màn hình yêu cầu nhập tên đăng nhập và mật khẩu.
2. Người dùng nhập tên và mật khẩu sau đó nhấp chuột vào nútLogin **.** Hệ thống kiểm tra tên đăng nhập, mật khẩu và quyền truy nhập và hiển thị màn hình trang chủ hệ thống. Use case kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 trong luồng cơ bản nếu người dùng nhập sai tên hoặc/và mật khẩu thì hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại. Người dùng nhập lại để tiếp tục và use case kết thúc.
2. Tại bước 2 trong luồng cơ bản nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị thông báo lỗivà 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.

### 2.5.7 Use case đăng ký

Mô tả vắn tắt:

Use case này cho phép người dùng đăng ký tài khoản.

Luồng sự kiện:

* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng nhấp chuột vào nút “SingIn” tại màn hình đăng nhập. Hệ thống sẽ chuyển người dùng sang màn hình đăng ký, yêu cầu người dùng nhập tên đăng nhập, mật khẩu, tên khách hàng, email.
2. Khi người dùng nhập thông tin và nhấp chuột vào nút đăng ký. Hệ thống sẽ lưu dữ liệu tài khoản của người dùng. Và use case kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 luồng cơ bản, nếu người dùng nhập tên đăng nhập đã có trong cơ sở dữ
2. hoặc bỏ bất kỳ trường thông tin nào thì hệ thống sẽ báo lỗi và yêu cầu người dùng nhập lại.
3. Tại tất cả các bước trong luồng cơ bản nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị một thông báo lỗi và 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.

### 2.5.8 Use case quản lý danh mục

Mô tả vắn tắt:

Use case này cho phép người quản trị cập nhật thông tin cho bảng danh mục.

Luồng sự kiện:

* Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị nhấp chuột vào nút **Quản lý Danh mục** trên thanh menu quản trị. Hệ thống sẽ truy vấn bảng Category, hiển thị danh sách các danh mục lên màn hình.
2. **Thêm danh mục:** khi người quản trị nhập thông tin chi tiết về danh mục gồm: tên danh mục, Status và nhấp chuột vào nút **Thêm mới**. Hệ thống sẽ kiểm tra dữ liệu nhập vào và tạo một bản ghi mới trong bảng Categories đồng thời cập nhật danh mục này lên danh sách.
3. **Sửa danh mục:** 
   1. Khi người quản trị nhấp chuột vào nút **Sửa** trên một dòng thông tin về danh mục. Hệ thống sẽ hiển thị các trường cho phép người quản lí sửa thông tin của danh mục.
   2. Người quản trị thay đổi thông tin danh mục gồm các thông tin được chỉ ra trong **Thêm danh mục** và nhấp chuột vào nút **Lưu**. Hệ thống kiểm tra thông tin thay đổi, cập nhật bảng Category và cập nhật danh sách danh mục.
4. **Xóa danh mục:** khi người quản trị nhấp chuột vào nút **Xóa** trên 1 dòng thông tin về danh mục. Hệ thống sẽ hiển thị thông báo xác nhận xóa danh mục:
5. Nếu người quản trị nhấp chuột vào nút xác nhận, hệ thống sẽ xóa bản ghi danh mục khỏi bảng categories và cập nhật danh sách danh mục.
6. Nếu người quản trị nhấp chuột nút hủy, hệ thống sẽ hủy bỏ thao tác xóa và use case kết thúc.

### 2.5.9 Use case quản lý sản phẩm

Mô tả vắn tắt:

Use case này cho phép người quản trị cập nhật thông tin cho bảng sản phẩm.

Luồng sự kiện:

* Luồng cơ bản:

**1.** Use case này bắt đầu khi người quản trị nhấp chuột vào nút **Quản lý sản phẩm** trên thanh menu quản trị. Hệ thống sẽ truy vấn bảng products, hiển thị danh sách các sản phẩm lên màn hình.

**2. Thêm sản phẩm:** khi người quản trị nhập thông tin chi tiết về sản phẩm gồm: tên sản phẩm, nơi sản xuất, giá tiền, models, hình ảnh, mô tả và nhấp chuột vào nút **Thêm mới** . Hệ thống sẽ kiểm tra dữ liệu nhập vào và tạo một bản ghi mới trong bảng Products đồng thời cập nhật danh mục này lên danh sách.

**3. Sửa sản phẩm:**

* 1. Khi người quản trị nhấp chuột vào nút **Sửa** trên một dòng thông tin về sản phẩm. Hệ thống sẽ hiển thị các trường cho phép người quản lí sửa thông tin của sản phẩm.
  2. Người quản trị thay đổi thông tin sản phẩm gồm các thông tin được chỉ ra trong **Thêm sản phẩm** và nhấp chuột vào nút **Lưu**. Hệ thống kiểm tra thông tin thay đổi, cập nhật bảng Products và cập nhật danh sách sản phẩm.

**4. Xóa sản phẩm :** khi người quản trị nhấp chuột vào nút **Xóa** trên một dòng thông tin về sản phẩm. Hệ thống sẽ hiển thị thông báo xác nhận xóa sản phẩm:

1. Nếu người quản trị nhấp chuột vào nút xác nhận, hệ thống sẽ xóa bản ghi danh mục khỏi bảng Product và cập nhật danh sách sản phẩm.
2. Nếu người quản trị nhấp chuột nút hủy, hệ thống sẽ hủy bỏ thao tác xóa và use case kết thúc.

### 2.5.10 Use case quản lý bài viết

Mô tả vắn tắt:

Use case này cho phép người quản trị cập nhật thông tin cho bảng bài viết.

Luồng sự kiện:

* Luồng cơ bản:

**1.** Use case này bắt đầu khi người quản trị nhấp chuột vào nút **Quản lý bài viết**  trên thanh menu quản trị. Hệ thống sẽ truy vấn bảng blog, hiển thị danh sách các bài viết lên màn hình.

**2. Thêm bài viết:** khi người quản trị nhập thông tin chi tiết về bài viết gồm: tên bài viết , nội dung bài viết , hình ảnh và nhấp chuột vào nút **Thêm mới** . Hệ thống sẽ kiểm tra dữ liệu nhập vào và tạo một bản ghi mới trong bảng Blog đồng thời cập nhật danh mục này lên danh sách.

**3. Sửa bài viết :**

* 1. Khi người quản trị nhấp chuột vào nút **Sửa** trên một dòng thông tin về bài viết. Hệ thống sẽ hiển thị các trường cho phép người quản lí sửa thông tin của bài viết.
  2. Người quản trị thay đổi thông tin danh mục gồm các thông tin được chỉ ra trong **Thêm bài viết** và nhấp chuột vào nút **Lưu**. Hệ thống kiểm tra thông tin thay đổi, cập nhật bảng Blog và cập nhật danh sách bài viết.

**4. Xóa bài viết:** khi người quản trị nhấp chuột vào nút **Xóa** trên 1 dòng thông tin về bài viết. Hệ thống sẽ hiển thị thông báo xác nhận xóa bài viết :

1. Nếu người quản trị nhấp chuột vào nút xác nhận, hệ thống sẽ xóa bản ghi danh mục khỏi bảng blog và cập nhật danh sách bài viết.
2. Nếu người quản trị nhấp chuột nút hủy, hệ thống sẽ hủy bỏ thao tác xóa và use case kết thúc.

### 2.5.11 Use case quản lý đơn hàng

Mô tả vắn tắt:

Use case này cho phép quản trị có thể theo dõi và cập nhật trạng thái của đơn hàng.

Luồng sự kiện:

* Luồng cơ bản:

1. Use case này bắt đầu khi người quản trị nhấp chuột vào nút **Quản lý đơn hàng** trên thanh menu. Hệ thống sẽ truy vấn bảng orders, hiển thị danh sách các đơn hàng. Người quản trị có thể theo dõi trạng thái và thông tin của các đơn hàng.
2. **Cập nhật trạng thái đơn hàng:** Khi người quản trị chọn 1 đơn hàng, hệ thống sẽ hiển thị các thông tin của đơn hàng đó và cho phép người quản trị cập nhật trạng thái của đơn hàng đó. Nếu người quản trị chọn 1 trạng thái cho đơn hàng và bấm lưu hệ thống sẽ cập nhật lại trạng thái đơn hàng trong bảng orders.
3. Use case kết thúc khi người quản trị chuyển sang màn hình khác.

* Luồng rẽ nhánh:

1. Trong quá trình thực hiện use case nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị một thông báo lỗi và use case kết thúc.

Các yêu cầu đặc biệt:

Không có.

Tiền điều kiện:

Người quản trị phải đăng nhập với quyền quản trị trước khi thực hiện use case.

Hậu điều kiện:

Không có.

Điểm mở rộng:

Không có.

### 2.5.12 Use case thống kê

Mô tả vắn tắt:

Use case này cho phép Admin xem được thông kê shop.

Luồng sự kiện:

* Luồng cơ bản:

1. Use case này bắt đầu khi Admin vào trang quản trị. Hệ thống sẽ hiển thị bảng thống kê về sản phẩm , đơn hàng. Use Case kết thúc.

* Luồng rẽ nhánh:

1.Nếu người dùng có kết nối Internet thì màn hình sẽ hiển thị ra thông báo lỗi .Use Case kết thúc.

Các yêu cầu đặc biệt:

Không có.

Tiền điều kiện:

Admin phải đăng nhập để vào được trang Admin.

Hậu điều kiện:

Không có.

Điểm mở rộng:

Không có.