|  |
| --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**  **ĐOÀN THỊ LINH**  **THIẾT KẾ WEBSITE BÁN THỰC PHẨM**  **ĐỒ ÁN 4**  **HƯNG YÊN - 2019** |

|  |
| --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**  **ĐOÀN THỊ LINH**  **THIẾT KẾ WEBSITE BÁN THỰC PHẨM**  NGÀNH: CÔNG NGHỆ THÔNG TIN  CHUYÊN NGÀNH: CÔNG NGHỆ WEB  **ĐỒ ÁN 4**  **NGƯỜI HƯỚNG DẪN**  **PHẠM NGỌC HƯNG**  **HƯNG YÊN - 2019** |

MỤC LỤC

[MỤC LỤC 3](#_Toc8117652)

[DANH SÁCH HÌNH VẼ 5](#_Toc8117653)

[DANH SÁCH BẢNG BIỂU 6](#_Toc8117654)

[DANH SÁCH TỪ VIẾT TẮT 7](#_Toc8117655)

[**CHƯƠNG 1:** **TỔNG QUAN VỀ ĐỀ TÀI** 8](#_Toc8117656)

[1.1 Lý do chọn đề tài 8](#_Toc8117657)

[1.2 Mục tiêu của đề tài 9](#_Toc8117658)

[1.3 Giới hạn và phạm vi của đề tài 9](#_Toc8117659)

[1.4 Nội dung thực hiện 9](#_Toc8117660)

[1.5 Phương pháp tiếp cận 10](#_Toc8117661)

[**CHƯƠNG 2:** **CƠ SỞ LÝ THUYẾT** 11](#_Toc8117662)

[2.1 Tổng quan về Framework Laravel 11](#_Toc8117663)

[2.1.1 Framework là gì? 11](#_Toc8117664)

[2.1.2 Mô hình MVC là gì? 11](#_Toc8117665)

[2.1.3 Laravel là gì? 11](#_Toc8117666)

[2.1.4 Tại sao dùng framework Laravel? 11](#_Toc8117667)

[2.2 Nguyên tắc thiết kế Framework Laravel 12](#_Toc8117668)

[2.2.1 Nguyên tắc 1: Tổ chức website chặt chẽ và dễ sử dụng 12](#_Toc8117669)

[2.2.2 Nguyên tắc 2: Đảm bảo các liên kết được khám phá dễ dàng 12](#_Toc8117670)

[**CHƯƠNG 3:** **NỘI DUNG THỰC HIỆN (Website bán thực phẩm)** 14](#_Toc8117671)

[3.1 Tìm hiểu về UML 14](#_Toc8117672)

[3.1.1 Tổng quan về UML 14](#_Toc8117673)

[3.1.2 Một số biểu đồ UML cơ bản 15](#_Toc8117674)

[3.1.3 Biểu đồ User Case cho hệ thống 17](#_Toc8117675)

[3.2 Thiết kế cơ sở dữ liệu của hệ thống. 30](#_Toc8117676)

[3.2.1 Thiết kế cơ sở dữ liệu quản lý nội dung 30](#_Toc8117677)

[3.3 Giao diện của hệ thống. 35](#_Toc8117678)

[3.3.1 Giao diện trang chủ 35](#_Toc8117679)

[3.3.2 Giao diện trang giỏ hàng 36](#_Toc8117680)

[3.3.3 Giao diện trang đăng kí 37](#_Toc8117681)

[3.3.4 Giao diện trang admin 37](#_Toc8117682)

[3.3.5 Giao diện trang admin quản lí sản phẩm 38](#_Toc8117683)

[3.3.6 Giao diện trang admin thêm sản phẩm 38](#_Toc8117684)

[**CHƯƠNG 4:** **KẾT LUẬN** 39](#_Toc8117685)

[4.1 Kết quả đạt được của đề tài 39](#_Toc8117686)

[4.2 Hạn chế của đề tài 39](#_Toc8117687)

[4.3 Hướng phát triển của đề tài 39](#_Toc8117688)

[TÀI LIỆU THAM KHẢO 41](#_Toc8117689)

[PHỤ LỤC 42](#_Toc8117690)

DANH SÁCH HÌNH VẼ

[Hình 3- 1: Biểu đồ Use Case tổng quát của hệ thống 17](#_Toc8217097)

[Hình 3- 2: Biểu đồ Use Case qản lí loại hàng 18](#_Toc8217098)

[Hình 3- 3: Biểu đồ Use Case quản lí sản phẩm 19](#_Toc8217099)

[Hình 3- 4: Biểu đồ Use Case quản lí nhà cung cấp 20](#_Toc8217100)

[Hình 3- 5: Biểu đồ Use Case quản lí khách hàng 21](#_Toc8217101)

[Hình 3- 6: Biểu đồ Use Case quản lí User 22](#_Toc8217102)

[Hình 3- 7:Biểu đồ Use Case quản lí nhân viên 23](#_Toc8217103)

[Hình 3- 8: Biểu đồ Use Case quản lí nhập hàng 24](#_Toc8217104)

[Hình 3- 9: Biểu đồ Use Case quản lí bán hàng 25](#_Toc8217105)

[Hình 3- 10: Biểu đồ Use Case quản lí tin tức 26](#_Toc8217106)

[Hình 3- 11: Biểu đồ Use Case thống kê 27](#_Toc8217107)

[Hình 3- 12: Biểu đồ tuần tự Chức năng thêm hàng hóa 28](#_Toc8217108)

[Hình 3- 13:Biểu đồ tuần tự cho chức năng sửa thông tin hàng hóa 28](#_Toc8217109)

[Hình 3- 14:Biểu đồ tuần tự chức năng xóa hàng hóa 29](#_Toc8217110)

[Hình 3- 15:Biểu đồ tuần tự chức năng đăng nhập 29](#_Toc8217111)

[Hình 3- 16: Mô hình lưu trữ cấu trúc website trên Database 30](#_Toc8217112)

[Hình 3- 17:Giao diện trang chủ 36](#_Toc8217113)

[Hình 3- 18:Giao diện trang giỏ hàng 36](#_Toc8217114)

[Hình 3- 19:Giao diện trang đăng kí 37](#_Toc8217115)

[Hình 3- 20:Giao diện trang admin 37](#_Toc8217116)

[Hình 3- 21:Giao diện trang quản lý sản phẩm 38](#_Toc8217117)

[Hình 3- 22:Giao diện trang thêm sản phẩm 38](#_Toc8217118)

DANH SÁCH BẢNG BIỂU

[Bảng 3- 1: Các kí hiệu trong UML 3](#_Toc8052529)

[Bảng 3- 2: Thiết kế bảng loại sản phẩm 3](#_Toc8052530)

[Bảng 3- 3: Thiết kế bảng sản phẩm 3](#_Toc8052531)

[Bảng 3- 5:Thiết kế bảng nhà cung cấp 3](#_Toc8052532)

[Bảng 3- 6:Thiết kế bảng nhân viên 3](#_Toc8052533)

[Bảng 3- 7:Thiết kế bảng khách hàng 3](#_Toc8052534)

[Bảng 3- 8:Thiết kế bảng User 3](#_Toc8052535)

[Bảng 3- 9:Thiết kế bảng hóa đơn nhập 3](#_Toc8052536)

[Bảng 3- 10:Thiết kế bảng chi tiết hóa đơn nhập 3](#_Toc8052537)

[Bảng 3- 11:Thiết kế bảng hóa đơn bán 3](#_Toc8052538)

[Bảng 3- 12:Thiết kế bảng chi tiết hóa đơn bán 3](#_Toc8052539)

[Bảng 3- 13:Thiết kế bảng phản hồi của khách hàng 3](#_Toc8052540)

[Bảng 3- 14:Thiết kế bảng tin tức 3](#_Toc8052541)

[Bảng 3- 15:Thiết kế bảng Slide 3](#_Toc8052542)

DANH SÁCH TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| Từ viết tắt | Từ đầy đủ | Giải thích |
| CMS | Content Management System | Hệ quản trị nội dung |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# **TỔNG QUAN VỀ ĐỀ TÀI**

* 1. Lý do chọn đề tài

Hoạt động quản lý bán thực phẩm được mô tả như sau:

* Trang chủ của website
* Trang này hiển thị toàn bộ những nội dung chính của website. Trên đây bạn có thể thực hiện tất cả các công việc mà bạn muốn bằng cách click vào những mục chọn tương ứng chương trình sẽ liên kết đến các trang tương ứng đáp ứng yêu cầu của bạn
* Đầu trang Web là phần tìm kiếm giúp khách hàng dễ dàng tìm kiếm sản phẩm muốn mua. Góc bên phải là phần đăng ký, đăng nhập vào hệ thống của thucphamsach.vn . Ở giữa là menu ngang liệt kê các danh mục: Trang chủ, menu sản phẩm, giới thiệu về website. Trong phần menu sản phẩm khi bạn di chuột vào sẽ hiển thị menu dọc danh mục các loại sản phẩm có trên website như: Trái cây, Rau xanh, nấm,…

Phía dưới sẽ là slide ảnh với chủ đề thực phẩm sạch.

Giữa trang sẽ hiển thị lên các sản phẩm nổi bật được khách hàng lựa chọn nhiều nhất, tiếp đến là một vài sản phẩm mới

Phần tiếp sẽ là thông tin của một vài nhà cung cấp. Và phần tin tức phía trên phần footer.

Cuối trang là phần giới thiệu , trang liên kết, hỗ trợ khách hàng, địa chỉ của thucphamsach,…

* Hoạt động đăng kí khách hàng
* Để trở thành thành viên website cũng như bạn hàng của công ty khách hàng phải đăng ký tài khoản nhằm mục đích quản lý thông tin của khách hàng.
* Khi điền xong các thông tin về : Họ tên,username,địa chỉ, email, sdt,mật khẩu,khách hàng nhấn “Đăng ký” để đăng ký tài khoản trên website.Hệ thống sẽ hiển thị thông báo cho khách hàng.
* Hoạt động mua hàng
* Khi khách hàng lựa chọn một mặt hàng nào đó từ danh mục các mặt hàng thì khách hàng sẽ được liên kết tới trang giới thiệu về sản phẩm bao gồm các thông tin :Tên sản phẩm, Hình ảnh, đơn giá, mô tả .Khách hàng khi chọn mua mặt hàng hệ thống sẽ yêu cầu các thông tin cần thiết để hoàn thành đơn hàng bao gồm: Hình thức thanh toán, thông tin cá nhân, hình thức nhận hàng, cuối cùng xác nhận đơn hàng và đặt mua.
* Sau khi khách hàng đã đặt mua thì họ có thể vào quản lý đơn hàng để theo dõi thông tin đơn hàng của mình. Trong phần đơn hàng khách hàng có thể hủy đơn hàng hoặc gia hạn thanh toán.
* Hoạt động quản trị
* Là người làm chủ hệ thống có quyền kiểm soát mọi hoạt động của hệ thống. Người làm quản trị có Usename và Password để đăng nhập vào hệ thống thực hiện các chức năng của mình. Khi đăng nhập thành công nhà quản trị có thể thực hiện các công việc sau:
* Quản lý mặt hàng :Quản lí thông tin về sản phẩm bao gồm: Tên sản phẩm, Tên loại sản phẩm,Nhà cung cấp, Đơn giá, Số lượng,Mô tả.
* Quản lí nhập hàng: Sau khi nhập sản phẩm về thì quản lý của hệ thống sẽ tiến hành nhập thông tin của sản phẩm: Mã loại sản phẩm, Tên loại sản phẩm, Mã sản phẩm, Tên sản phẩm, Nhà cung cấp sản phẩm,Mô tả sản phẩm, Số lượng, Giá tiền, Hình ảnh. Từ những thông tin đó người quản lý lưu trữ lại thông tin, cập nhật sản phẩm và quản lý
* Quản lí khách hàng: Khi khách hàng đặt mua sản phẩm nào đó trên website thì hệ thống sẽ tự động tiến hành lập hóa đơn và thanh toán gồm các thông tin sau: Mã hóa đơn, Mã sản phẩm, Mã khách hàng, Số lượng, Giá tiền. Từ thông tin đó người quản lý sẽ lưu trữ lại thông tin và quản lý. Trong trường hợp khách hàng hủy đơn hàng thì người quản lý sẽ tiến hành xóa đơn hàng.
* Người quản lý sẽ quản lý thông tin khách hàng bao gồm: Mã khách hàng, Tên khách hàng, Số điện thoại, Địa chỉ, Email
* Duyệt đơn hàng:Sau khi lập hóa đơn và có được thông tin của khách hàng thì người quản lý sẽ tiến hành xác thực thông tin khách hàng đã đặt hàng để có thể giao hàng luôn.
* Bán hàng: Thực hiện theo đơn đặt của khách hàng. Nếu hàng không đúng với đơn đặt hàng hoặc bị hỏng hóc thì cửa hàng có những hình thức giải quyết phù hợp theo đúng phạm vi và trách nhiêm của mình.
* Quản lý ý kiến phản hồi khách hàng, các bình luận khi được cập nhập trên hệ thống phải được kiểm tra nội dung đẻ đảm bảo thông tin trong sáng, lành mạnh.
* Quản lý tin tức: cập nhập, đăng tin lên website
* Quản lý quảng cáo: Có thể cho phép chèn quảng cáo lên website
* Cuối tháng người quản lý thống kê xem sản phẩm nào còn sản phẩm nào sắp hết để nhập hàng cho phù hợp. Thống kê doanh thu của cửa hàng trong 1 ngày và 1 tháng
  1. Mục tiêu của đề tài
* Quản lí thông tin loại các sản phẩm
* Quản lí thông tin các sản phẩm
* Quản lí Nhà cung cấp
* Quản lí khách hàng
* Quản lí tài khoản
* Quản lí phản hồi của khách hàng
* Quản lí nhân viên
* Quản lí hóa đơn
* Thống kê , báo cáo
* Quản lý giỏ hàng
* Quản lí tin tức
  1. Giới hạn và phạm vi của đề tài

Đề tài xây dựng dựng một Web Framework sử dụng các công nghệ mới trong lập trình như Laravel,MySQL và Công cụ lập trình SubLime Text.

Hệ thống bán thực phẩm cung cấp các chức năng quản lí sản phẩm, quản lí mua hàng

* 1. Nội dung thực hiện

Nội dung thực hiện/nghiên cứu cụ thể như sau:

* Nghiên cứu Framework Laravel.
* Phân tích yêu cầu của hệ thống.
* Thiết kế đặc tả hệ thống.
* Xây dựng Cơ sở dữ liệu.
* Lập trình cho các Module của hệ thống
* Triển khai thực nghiệm hệ thống trên sever.
  1. Phương pháp tiếp cận
* Cách tiếp cận : Nghiên cứu các Web Framework trên các nền tảng công nghệ khác nhau như PHP.
* Sử dụng các phương pháp nghiên cứu:
  + Phương pháp đọc tài liệu;
  + Phương pháp phân tích mẫu;
  + Phương pháp thực nghiệm.

# **CƠ SỞ LÝ THUYẾT**

* 1. Tổng quan về Framework Laravel
     1. Framework là gì?

Framework là một thư viện các lớp đã được xây dựng hoàn chỉnh, bộ khung để phát triển các Phần mềm ứng dụng. Có thể ví Framework như tập các “ Vật liệu” ở từng lĩnh vực cho người lập trình viên, thay vì họ phải mất nhiều thời gian để tự thiết kế trước khi dùng. Do vậy, người lập trình viên chỉ cần tìm hiểu và khai thác các vật liệu này rồi thực hiện để gắn kết chúng lại với nhau, tạo ra sản phẩm

* + 1. Mô hình MVC là gì?
* MVC= Model + View + Controller
* Model: Chính là dự liệu ( cụ thể hơn trong ZF sẽ là các class làm nhiệm vụ thao tác trực tiếp xuống DBMS)
* Mô hình Views có nhiệm vụ liên kết với Mô hình Model và xuất các dữ liệu ra trình duyệt theo nhu cầu đòi hỏi của người tiêu dùng( user). Điển hình là các văn bản HTML View : Làm nhiệm vụ render trang web từ các action do Controller truyền sang + dữ liệu từ model ( có thể hiểu nó như template render)
* Controller: chính là phần cốt lõi, điều hành trang web của bạn, 1 trang web có thể có nhiều module ( có thể  hiểu như compoment của Joomla), một module có thể có nhiều controller. Một controller sẽ gồm nhiều action.
* Ví dụ trong Forum sẽ có Post Controller bao gồm các action như listAction, readAction, writeAction,…
  + 1. Laravel là gì?
* Laravel là một PHP Framework mã nguồn mở và miễn phí, được phát triển bởi Taylor Otwell và nhắm mục tiêu hỗ trợ phát triển các ứng dụng web theo cấu trúc model- view- controller (MVC). Những tính năng nổi bật của Laravel bao gồm cú pháp dễ hiểu- rõ ràng, một hệ thống đóng gói Modular và quản lý gói phụ thuộc, nhiều cách khác nhau để truy cập vào các cơ sở dữ liệu quan hệ, nhiều tiện ích khác nhau hỗ trợ việc triển khai vào bảo trì ứng dụng.
* Vào khoảng Tháng 3 năm 2015, các lập trình viên đã có một cuộc bình chọn PHP framework phổ biến nhất, Laravel đã giành vị trí quán quân cho PHP framework phổ biến nhất năm 2015, theo sau lần lượt là Symfony2, Nette, CodeIgniter, Yii2 vào một số khác. Trước đó, Tháng 8 năm 2014, Laravel đã trở thành project PHP phổ biến nhất và được theo dõi nhiều nhất trên Github
* Laravel được phát hành theo giấy phép MIT, với source code được lưu trữ tại Gitthub
  + 1. Tại sao dùng framework Laravel?

Laravel có một hệ thống có kiến trúc thống nhất, khoa học và cực đơn giản giúp những người mới tiếp cận framework PHP có thể nắm bắt được ngay. Các framework khác thật sự khá là khó để bắt đầu, đặc biệt là Zend, Yii. Tuy rất đơn giản nhưng Laravel hoàn toàn có thể xây dựng những hệ thống ứng dụng web cực phức tạp, với hiệu năng cao.

Các tài nguyên liên quan đến Laravel rất đa dạng, dễ sử dụng, kể đến có tài liệu chính thức của Laravel được chính Taylor Otwell biên soạn, tác giả của framework này muốn đưa việc lập trình lên một tầm cao mới đó là khám phá và trải nghiệm, ngoài ra cộng đồng Laravel rất lớn trên Laracast, bạn có tìm được câu trả lời cho bất kỳ tình huống nào bạn gặp phải trong Laravel hoặc tiếp cận rất nhiều các video hướng dẫn cụ thể.

Trực quan, đó là những gì mọi người nói về viết mã trong Laravel, giúp chúng ta tập trung vào các nhiệm vụ chính và các đoạn mã viết trong Laravel được rõ ràng và rất dễ dàng trong bảo trì mã nguồn.

Tích hợp công cụ dòng lệnh Artisan, giúp các tác vụ trong phát triển ứng dụng thật đơn giản. Những người mới tiếp cận với Laravel sẽ rất ít để ý đến Artisan vì có một suy nghĩ là tại sao viết code lại liên quan đến một công cụ dòng lệnh? Artisan là một phụ tá đắc lực trong quá trình phát triển ứng dụng, giúp các công việc như tối ưu hóa ứng dụng, migrate dữ liệu, tạo các template, gọi các ứng dụng khác trong xử lý song song với hàng đợi… được thực hiện đơn giản. Có lẽ do Taylor Otwell đã quen với Microsoft Visual Studio .NET khi lập trình .NET nên cũng mong muốn có những công cụ tốt cho những lập trình viên PHP.

Bộ máy blade template thực sự rất hay, nó giúp cho làm việc giữa PHP và HTML thật đơn giản, sáng sủa và tách biệt giữa mã hiển thị và mã logic nghiệp vụ, cái mà trước đó chỉ có thể biết đến ở .NET hoặc Java.

Laravel hỗ trợ việc cài đặt các gói thư viện và bản thân nó sử dụng rất nhiều các gói thư viện khác để tạo ra các tính năng tuyệt vời, ví dụ: nếu bạn làm việc với Laravel Mix bạn sẽ nhận thấy Laravel sử dụng gói thư viện Webpack kết hợp với npm, yarn… giúp bạn tự động hóa, giảm rất nhiều các tác vụ thủ công trong quá trình build ứng dụng và triển khai sản phẩm.

* 1. Nguyên tắc thiết kế Framework Laravel
     1. Nguyên tắc 1: Tổ chức website chặt chẽ và dễ sử dụng

Cách tổ chức và phân chia nội dung cho website là một trong những vấn đề rất quan trọng đối với người dùng truy cập web. Ngoài ra, bạn còn cần phải quan tâm đến bố cục của website sao cho hợp lý để người dùng dễ dàng tiếp cận các thông tin cần thiết khi truy cập. Bởi website tổ chức không tốt thì người dùng sẽ khó tiếp cận được những nội dung theo đúng ý muốn của đơn vị. Từ đó, người truy cập sẽ nhanh chóng thoát ra khỏi web và cũng sẽ không muốn quay lại khi có nhu cầu tìm kiếm các thông tin liên quan khác.

Về cách phân chia và tổ chức nội dung, trước khi Framework được thiết kế, nhóm đề tài đã định hình một số khung giao diện thường gặp và thiết kế chúng trở thành Skin. Sau đó phân chia các Control theo các Module và vị trí. Qua việc phân tích nhu cầu tìm đọc của người dùng, nhóm đề tài đã lựa chọn xây dựng các module chính, module tiện ích giúp người dùng có thể tìm đến bản tin một cách nhanh nhất.

* + 1. Nguyên tắc 2: Đảm bảo các liên kết được khám phá dễ dàng

Các đường liên kết là một điểm mạnh của website so với những tập thông tin bằng giấy. Do đó, nhóm đề tài đã tận dụng tối đa ưu điểm này để phân chia nội dung nhằm giúp người tiếp cận thông tin của bạn một cách dễ dàng nhất.

Một vấn đề đặt ra, tại sao trên các mạng xã hội, thông tin được truyền tải đi rất nhanh và rộng rãi. Bởi các nhà thiết kế đã tạo ra rất nhiều liên kết toàn bộ trên các thông tin của người dùng với bạn bè của họ. Tương tự như vậy, muốn người dùng khám phá nhiều thông tin trên website, nhóm đề tài đã phân chia nội dung và bố cục có những đường liên kết đến ngay các vị trí thích hợp cho người dùng.

# **NỘI DUNG THỰC HIỆN (Website bán thực phẩm)**

* 1. Tìm hiểu về UML
     1. Tổng quan về UML
* UML (Unified Model Language) là một ngôn ngữ dùng cho phân tích thiết kế hướng đối tượng (OOAD – Object Oriented Analys and Design)
* UML được duy trì và phát triển bởi OMG (Object Management Group), do Jacobson, Booch, Rumbaugh sáng lập. Ngoài ra còn có hàng trăm các tập đoàn lớn khác bảo trợ phát triển.
* Các biểu đồ UML cho ta cái nhìn rõ hơn về hệ thống (cái nhìn tĩnh và động)
* Hiện nay UML được sử dụng rất phổ biến trong các dự án phần mềm.
* UML thể hiện phương pháp phân tích hướng đối tượng nên không lệ thuộc ngôn ngữ lập trình.
* Có rất nhiều công cụ phần mềm hỗ trợ phân tích thiết kế dùng UML.
* Nhiều công cụ có thể sinh ra mã từ UML và ngược lại.
* UML không phải là ngôn ngữ lập trình.
* UML là một ngôn ngữ dùng để:
* Trực quan hóa (Visualizing):

+ Dùng tập các ký hiệu đồ họa phong phú để biểu diễn hệ thống đang được nghiên cứu.

+ Hệ thống ký hiệu đều có ngữ nghĩa chặt chẽ, có thể hiểu bởi nhiều công cụ khác nhau.

+ Giúp cho các nhà thiết kế, nhà lập trình khác biệt về ngôn ngữ đều có thể hiểu được.

* Đặc tả (Specifying):

+ UML giúp xây dựng các mô hình chính xác, đầy đủ và không nhập nhằng.

+ Tất cả các công đoạn từ phân tích, thiết kế cho đến triển khai đều có các biểu đồ UML biểu diễn.

+ Use case (dùng cho phân tích); Class, Sequence, Activity... (cho thiết kế); Component, Deployment (cho triển khai).

* Xây dựng (Constructing):

+ Các mô hình của UML có thể kết nối với nhiều ngôn ngữ lập trình. Tức là có thể ánh xạ các mô hình UML về một ngôn ngữ lập trình như C++, Java...

+ Việc chuyển các mô hình trong UML thành Code trong ngôn ngữ lập trình 🡪 Forward engineering

+ Việc chuyển ngược trở lại code trong một ngôn ngữ lập trình thành UML 🡪 Reverse Engineering.

+ Cần công cụ để chuyển đổi “xuôi” & “ngược”

* Viết tài liệu (Documenting):

+ Giúp xây dựng tài liệu đặc tả - requirements

+ Tài liệu kiến trúc (architecture)

+ Tài liệu thiết kế

+ Source code

+ Tài liệu để kiểm thử - Test

+ Tài liệu mẫu - Prototype

+ Tài liệu triển khai – Deployment ...

* + 1. Một số biểu đồ UML cơ bản

1. **Biểu đồ Use Case Diagram**

* Use case (Use Case Diagram) cung cấp một bức tranh toàn cảnh về những gì đang xảy ra trong hệ thống hiện tại hoặc những gì sẽ xảy ra trong hệ thống mới.
* Use case mô tả một chuỗi các hành động mà hệ thống sẽ thực hiện để đạt được kết quả có ý nghĩa đối với một tác nhân.
* Các use case được tạo ra ở giai đoạn đầu của một dự án. Các use case cũng có thể được dung ở giai đoạn sau của qui trình phát triển dự án, ví dụ để đặc tả các tình huống kiểm tra. Use case đưa ra cơ sở để xác định giao tiếp người-máy đối với hệ thống. Để người dùng cuối có thể hiểu được và có thể giao tiếp với hệ thống ở mức tổng thể.
* Actor là người hoặc hệ thống tương tác với các use case. Thường actor là người dùng hệ thống. Trong biểu đồ use case, mỗi actor được vẽ bằng một biểu tượng hình người với tên vai trò (role name) đặt bên dưới.
* Khi actor là người thì tên actor là tên vai trò mà actor đảm nhiệm chứ không phải là tên công việc.

1. **Biểu đồ lớp Class Diagram**

* Biểu đồ lớp (Class Diagram) cho ta một khung nhìn tĩnh của các lớp trong mô hình hoặc một phần của mô hình. Nó chỉ cho ta thấy các thuộc tính và các thao tác của lớp, cũng như các loại quan hệ giữa các lớp.
* Có rất nhiều kiểu lớp dùng trong biểu đồ lớp, ở đây chúng ta chỉ xét đến và sử dụng 3 loại lớp chính là : lớp biên (phần giao diện), lớp điều khiển và lớp cơ sở dữ liệu:

+ Lớp biên: là lớp thể hiện phần giao diện mà người dùng trực tiếp tương tác với hệ thống qua những giao diện của lớp này.

+ Lớp thực thể (lớp cơ sở dữ liệu): là lớp biểu diễn cho những thực thể có mối quan hệ với nhau và liên quan đến hệ thống.

1. **Biểu đồ tuần tự Sequence Diagram**

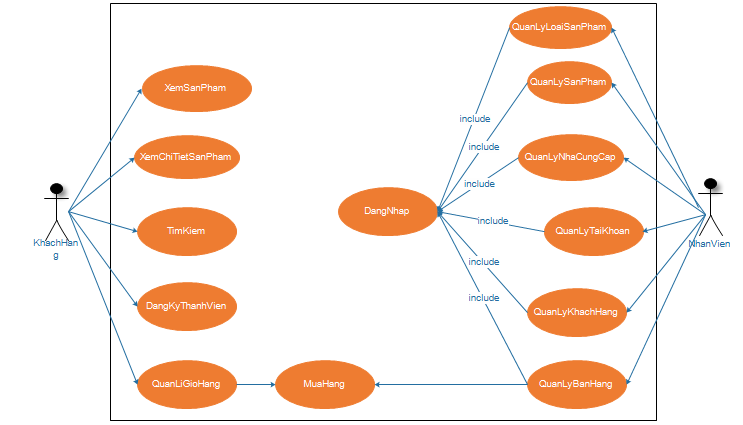
Biểu đồ tuần tự (***Sequence Diagram***) mô tả sự tương tác của các đối tượng theo trình tự về thời gian.Có sự liên kết chặt chẽ với biểu đồ lớp và mô tả thứ tự của tương tác bằng cách đánh số thứ tự các thông điệp. Mỗi biểu đồ tuần tự mô tả một tình huống xử lý.

1. **Các ký hiệu**

Bảng 3- 1: Các kí hiệu trong UML

|  |  |
| --- | --- |
|  | Biểu diễn các Actor (tác nhân) biểu diễn người hay hệ thống khác tương tác với hệ thống hiện tại. |
|  | Biểu diễn các Use Case (tình huống sử dụng) biểu diễn chuỗi hành động mà hệ thống thực hiện. |
|  | Biểu diễn mối quan hệ giữa Actor (người dùng) và UseCase (chức năng của hệ thống) |
|  | Quan hệ <<Include>> thể hiện Use Case này chứa Use Case khác.  Use case A được gọi là Include B nếu trong xử lý của A có gọi đến B ít nhất 1 lần. |
|  | Quan hệ <<Extend>> thể hiện User Case này mở rộng Use Case khác.  Use case B được gọi là Exntend A nếu use case B được gọi bởi A nếu thỏa mãn điều kiện nào đó. |
| 1 | Quan hệ tổng quát hóa (thừa kế):  Use case A được gọi là Generalization Use Case B nếu Use Case B là một trường hợp riêng của Use Case A. |
|  | Biểu đồ Lớp ClassDiagram:  - Mô tả các đối tượng và mối quan hệ của chúng trong hệ thống.  - Mô tả các thuộc tính và các hành vi (Behavior) của đối tượng. |
|  | Quan hệ Generalization: Thể hiện rằng một lớp A kế thừa từ một lớp B (Hay Class A là trường hợp riêng của Class B; Class B là tổng quát của Class A).  Gọi là quan hệ*Là một* (*Is a*) |
| 1 | Quan hệ Composition:  Thể hiện rằng một lớp A bao hàm lớp B. Nhưng lớp B không thể tồn tại độc lập (tức không thuộc lớp nào). Tức là, nếu có B thì phải suy ra được A. |
|  | Biểu đồ tuần tự:  - Mô tả sự tương tác của các đối tượng theo trình tự về thời gian.  - Có sự liên kết chặt chẽ với biểu đồ lớp.  - Mỗi biểu đồ tuần tự mô tả một tình huống xử lý nhất định. |

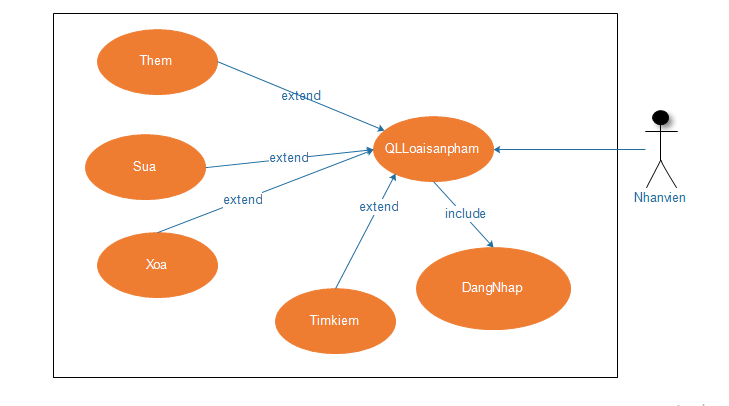
* + 1. Biểu đồ User Case cho hệ thống
* Use Case tổng quát



Hình 3- 1: Biểu đồ Use Case tổng quát của hệ thống

* Use Case phân rã

**Quản lí loại hàng**



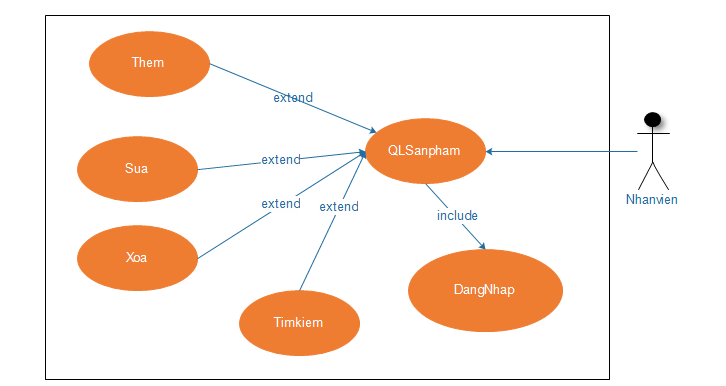
Hình 3- 2: Biểu đồ Use Case qản lí loại hàng

* **Mục đích**: Nhân viên có thể thêm, sửa , xóa thông tin các sản phẩm
* **Tác nhân**: Nhân viên kho
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm loại sản phẩm | + Nhân viên chọn chức năng thêm loại hàng.  + Hệ thống tự động sinh mã loại mới  + Nhân viên điền thông tin về loại hàng.  + Nhân viên nhấn xác nhận, hệ thống sẽ lưu thông tin vào csdl |
| * Sửa loại sản phẩm | + Nhân viên nhấn vào tìm kiếm, tìm kiếm loại hàng muốn sửa thông tin theo tên loại.  + Chọn chức năng sửa thông tin.  + Hệ thống hiển thị thông tin của loại sản phẩm lên các ô textbox  + Nhân viên thực hiện sửa các thông tin cần thiết.  + Nếu nhấn xác nhận thì hệ thống sẽ thay đổi thông tin cũ bằng các thông tin mới khách hàng vừa sửa  + Nếu hủy thì hệ thống sẽ hủy thao tác. Những thông tin sẽ vẫn được giữ nguyên. |
| * Xóa loại sản phẩm | + Nhân viên gõ ô tìm kiếm, nhập loại sản phẩm muốn xóa  + Sau khi tìm thấy sản phẩm muốn xóa chọn chức năng xóa để xóa sản phẩm. |
| * Tìm kiếm | + Nhân viên chọn chức năng tìm kiếm  + Nhân viên nhập từ khóa vào ô tìm kiếm  + Hệ thống sẽ hiển thị những sản phẩm gần với từ khóa tìm kiếm nhất |

**Quản lí sản phẩm**



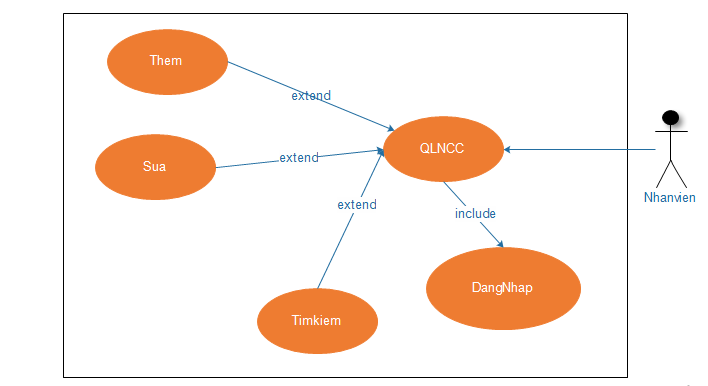
Hình 3- 3: Biểu đồ Use Case quản lí sản phẩm

* **Mục đích**: Nhân viên có thể thêm, sửa , xóa thông tin các sản phẩm
* **Tác nhân**: Nhân viên kho
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm sản phẩm | + Nhân viên chọn chức năng thêm sản phẩm.  + Hệ thống tự động sinh mã sản phẩm mới theo quy định  + Nhân viên điền thông tin về sản phẩm mới.  + Nhân viên nhấn xác nhận, hệ thống sẽ lưu thông tin vào csdl |
| * Sửa sản phẩm | + Nhân viên nhấn vào tìm kiếm, tìm kiếm sản phẩm muốn sửa thông tin.  + Chọn chức năng sửa thông tin.  + Hệ thống hiển thị thông tin của loại sản phẩm lên các ô textbox  + Nhân viên thực hiện sửa các thông tin cần thiết.  + Nếu nhấn xác nhận thì hệ thống sẽ thay đổi thông tin cũ bằng các thông tin mới khách hàng vừa sửa  + Nếu hủy thì hệ thống sẽ hủy thao tác. Những thông tin sẽ vẫn được giữ nguyên. |
| * Xóa sản phẩm | + Nhân viên gõ ô tìm kiếm, nhập sản phẩm muốn xóa  + Sau khi tìm thấy sản phẩm muốn xóa chọn chức năng xóa để xóa sản phẩm. |
| * Tìm kiếm | + Nhân viên chọn chức năng tìm kiếm  + Nhân viên nhập từ khóa vào ô tìm kiếm  + Hệ thống sẽ hiển thị những sản phẩm gần với từ khóa tìm kiếm nhất |

**Quản lí nhà cung cấp**



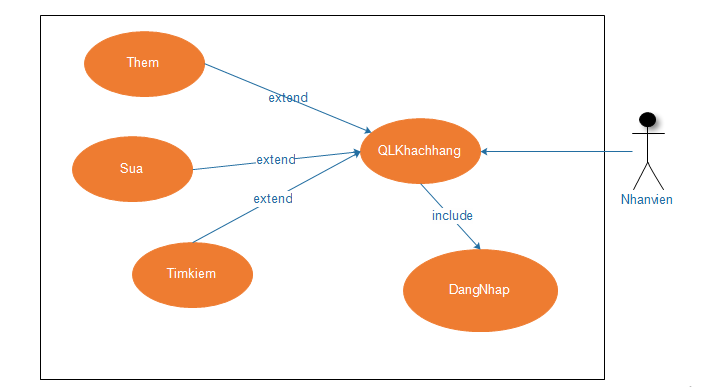
Hình 3- 4: Biểu đồ Use Case quản lí nhà cung cấp

* **Mục đích**: Nhân viên có thể thêm thông tin của nhà cung cấp
* **Tác nhân**: Nhân viên kho
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm nhà cung cấp | + Nhân viên chọn chức năng thêm nhà cung cấp.  + Hệ thống tự động sinh mã nhà cung cấp theo yêu cầu.  + Nhân viên điền thông tin về nhà cung cấp.  + Nhân viên nhấn xác nhận, hệ thống sẽ lưu thông tin vào csdl |
| * Sửa thông tin NCC | + Nhân viên nhấn vào tìm kiếm, tìm kiếm nhà cung cấp muốn sửa thông tin.  + Chọn chức năng sửa thông tin.  + Hệ thống hiển thị thông tin của nhà cung cấp lên các ô textbox  + Nhân viên thực hiện sửa các thông tin cần thiết.  + Nếu nhấn xác nhận thì hệ thống sẽ thay đổi thông tin cũ bằng các thông tin mới khách hàng vừa sửa  + Nếu hủy thì hệ thống sẽ hủy thao tác. Những thông tin sẽ vẫn được giữ nguyên. |
| * Tìm kiếm | + Nhân viên chọn chức năng tìm kiếm  + Nhân viên nhập từ khóa vào ô tìm kiếm  + Hệ thống sẽ hiển thị thông tin của những NCC gần với từ khóa tìm kiếm nhất |

**Quản lí khách hàng**



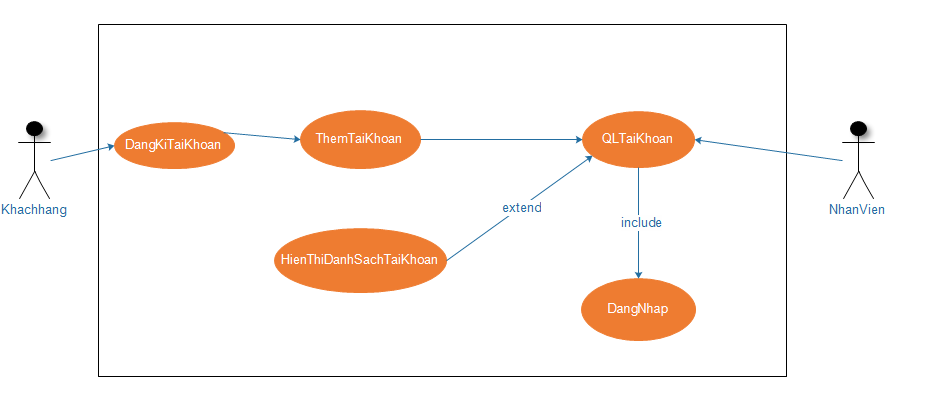
Hình 3- 5: Biểu đồ Use Case quản lí khách hàng

* **Mục đích**: Nhân viên có thể quản lí thông tin của khách hàng
* **Tác nhân**: Nhân viên
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm khách hàng | + Nhân viên chọn chức năng thêm khách hàng.  + Hệ thống tự động sinh mã khách hàng theo yêu cầu.  + Nhân viên điền thông tin khách hàng  + Nhân viên nhấn xác nhận, hệ thống sẽ lưu thông tin vào csdl |
| * Sửa khách hàng | + Nhân viên nhấn vào tìm kiếm, tìm kiếm khách hàng muốn sửa thông tin.  + Chọn chức năng sửa thông tin.  + Hệ thống hiển thị thông tin của khách hàng lên các ô textbox  + Nhân viên thực hiện sửa các thông tin cần thiết.  + Nếu nhấn xác nhận thì hệ thống sẽ thay đổi thông tin cũ bằng các thông tin mới khách hàng vừa sửa  + Nếu hủy thì hệ thống sẽ hủy thao tác. Những thông tin sẽ vẫn được giữ nguyên. |
| * Tìm kiếm | + Nhân viên chọn chức năng tìm kiếm  + Nhân viên nhập từ khóa vào ô tìm kiếm  + Hệ thống sẽ hiển thị thông tin của những khách hàng gần với từ khóa tìm kiếm nhất |

**Quản lí tài khoản**



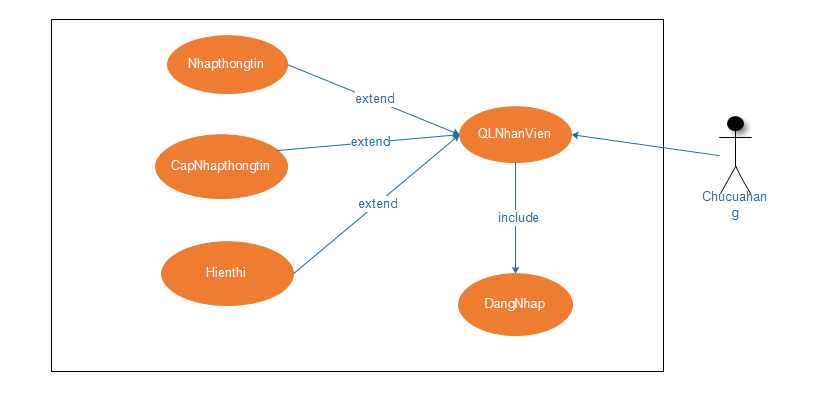
Hình 3- 6: Biểu đồ Use Case quản lí User

* **Mục đích**: Nhân viên quản lí những tài khoản được đăng kí trên website
* **Tác nhân**: Nhân viên ,khách hàng
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm tài khoản | + Khách hàng chọn chức năng đăng kí  + Khách hàng điền thông tin của mình vào form có sẵn  + Khách hàng nhấn xác nhận  + Hệ thống tự động sinh mã khách hàng theo yêu cầu và hệ thống sẽ lưu thông tin vào csdl |
| * Hiển thị danh sách tài khoản | + Nhân viên nhấn vào quản lí danh sách tài khoản  + Hệ thống hiển thị danh sách các tài khoản có trên hệ thống. Phần Password bị ẩn đi |

**Quản lí nhân viên**



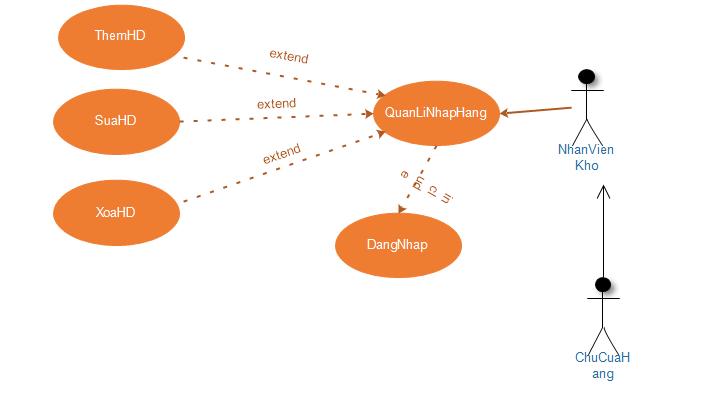
Hình 3- 7:Biểu đồ Use Case quản lí nhân viên

* **Mục đích**: Hê thống cung cấp chức năng này để giúp chủ cửa hàng có thể dễ dàng quản lí thông tin của các nhân viên.
* **Tác nhân**: Người quản lí( Chủ cửa hàng)
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm thông tin nhân viên | + Người quản lí chọn chức năng thêm nhân viên.  + Người quản lí điền thông tin của nhân viên  + Nhân viên nhấn xác nhận, hệ thống tự động sinh mã khách hàng theo yêu cầu và hệ thống sẽ lưu thông tin vào csdl |
| * Sửa thông tin nhân viên | + Người quản lí nhấn vào tìm kiếm, tìm kiếm nhân viên muốn sửa thông tin.  + Chọn chức năng sửa thông tin.  + Hệ thống hiển thị thông tin của nhân viên lên các ô textbox  + Người quản lí thực hiện sửa các thông tin cần thiết.  + Nếu nhấn xác nhận thì hệ thống sẽ thay đổi thông tin cũ bằng các thông tin mới nhân viên vừa sửa  + Nếu hủy thì hệ thống sẽ hủy thao tác. Những thông tin sẽ vẫn được giữ nguyên. |

**Quản lí nhập hàng**



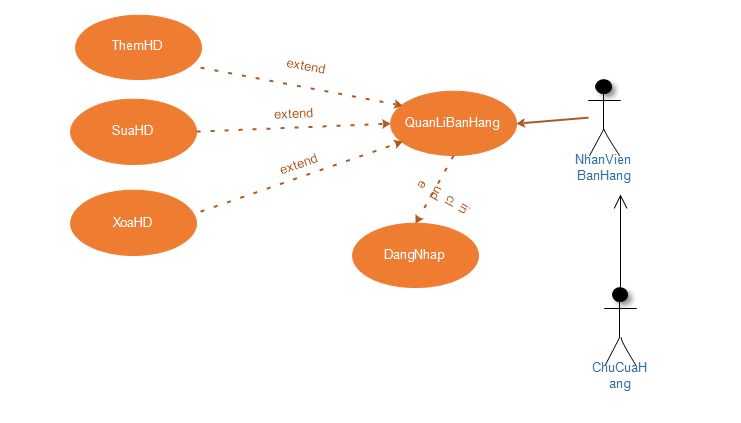
Hình 3- 8: Biểu đồ Use Case quản lí nhập hàng

* **Mục đích**: Nhân viên có thể thêm thông tin nhập hàng
* **Tác nhân**: Nhân viên kho
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm hóa đơn | + Nhân viên chọn chức năng thêm hóa đơn nhập.  + Hệ thống tự động sinh mã hóa đơn theo yêu cầu.  + Nhân viên điền thông tin về hóa đơn.  + Nhân viên nhấn xác nhận, hệ thống sẽ lưu thông tin vào csdl |
| * Sửa hóa đơn | + Nhân viên nhấn vào tìm kiếm, tìm kiếm hóa đơn muốn sửa thông tin.  + Chọn chức năng sửa thông tin.  + Hệ thống hiển thị thông tin của hóa đơn lên các ô textbox  + Nhân viên thực hiện sửa các thông tin cần thiết.  + Nếu nhấn xác nhận thì hệ thống sẽ thay đổi thông tin cũ bằng các thông tin mới khách hàng vừa sửa  + Nếu hủy thì hệ thống sẽ hủy thao tác. Những thông tin sẽ vẫn được giữ nguyên. |

**Quản lí bán hàng**



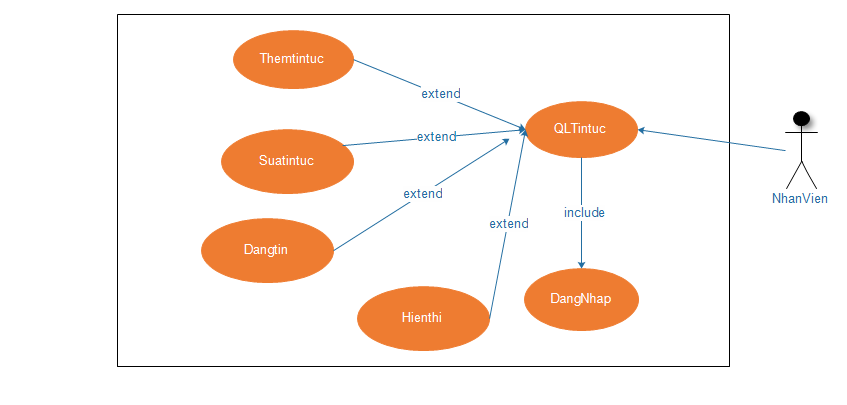
Hình 3- 9: Biểu đồ Use Case quản lí bán hàng

* **Mục đích**: Nhân viên có thể thêm thông tin bán hàng
* **Tác nhân**: Nhân viên bán hàng
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm hóa đơn | + Nhân viên chọn chức năng thêm hóa đơn bán.  + Hệ thống tự động sinh mã hóa đơn theo yêu cầu.  + Nhân viên điền thông tin về hóa đơn.  + Nhân viên nhấn xác nhận, hệ thống sẽ lưu thông tin vào csdl |
| * Sửa hóa đơn | + Nhân viên nhấn vào tìm kiếm, tìm kiếm hóa đơn muốn sửa thông tin.  + Chọn chức năng sửa thông tin.  + Hệ thống hiển thị thông tin của hóa đơn lên các ô textbox  + Nhân viên thực hiện sửa các thông tin cần thiết.  + Nếu nhấn xác nhận thì hệ thống sẽ thay đổi thông tin cũ bằng các thông tin mới khách hàng vừa sửa  + Nếu hủy thì hệ thống sẽ hủy thao tác. Những thông tin sẽ vẫn được giữ nguyên. |

**Quản lí tin tức**



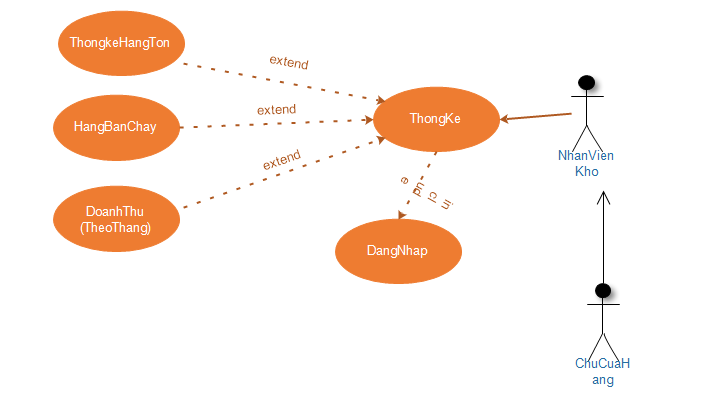
Hình 3- 10: Biểu đồ Use Case quản lí tin tức

* **Mục đích**: Nhân viên có thể quản lí những tin tức trên website
* **Tác nhân**: Nhân viên
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thêm tin tức | + Nhân viên chọn chức năng thêm tin tức.  + Hệ thống tự động sinh mã hóa đơn theo yêu cầu.  + Nhân viên điền thông tin về tin tức  + Nhân viên nhấn xác nhận, hệ thống sẽ lưu thông tin vào csdl |
| * Sửa tin tức | + Nhân viên nhấn vào tìm kiếm, tìm kiếm tin tức muốn sửa thông tin.  + Chọn chức năng sửa thông tin.  + Hệ thống hiển thị thông tin của hóa đơn lên các ô textbox  + Nhân viên thực hiện sửa các thông tin cần thiết.  + Nếu nhấn xác nhận thì hệ thống sẽ thay đổi thông tin cũ bằng các thông tin mới khách hàng vừa sửa  + Nếu hủy thì hệ thống sẽ hủy thao tác. Những thông tin sẽ vẫn được giữ nguyên. |
| * Đăng tin | + Nhân viên chọn chức năng đăng tin tức.  + Hệ thống hiển thị danh sách các tin tức  + Nhân viên chọn các tin muốn đăng  + Nhấn xác nhận để các tin sẽ xuất hiện trên website |

**Thống kê**



Hình 3- 11: Biểu đồ Use Case thống kê

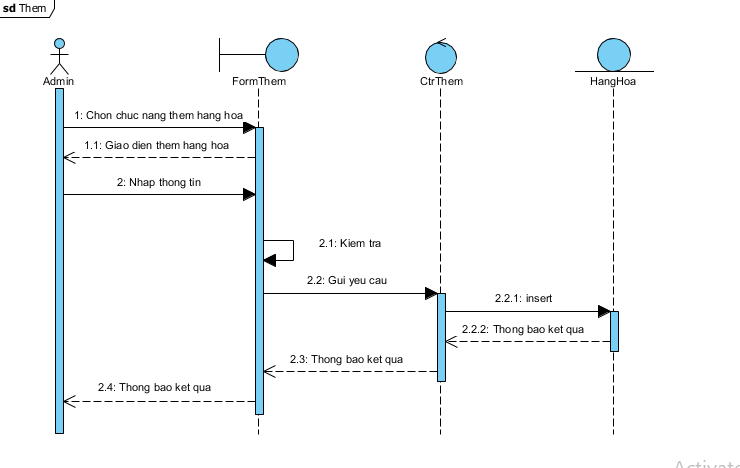
* **Mục đích**: Nhân viên sử dụng chức năng này để thực hiên thống kê vào cuối tháng
* **Tác nhân**: Nhân viên
* **Dòng sự kiện**

Sau khi đăng nhập vào hệ thống

|  |  |
| --- | --- |
| * Thống kê | + Nhân viên chọn chức năng thống kê  + Hệ thống hiển thị các lựa chon: Thống kê sản phẩm, doanh thu, tin tức, …  + Nhân viên chọn mục muốn thống kê  + Hệ thống đưa ra kết quả thống kê |
| * IN báo cáo |  |

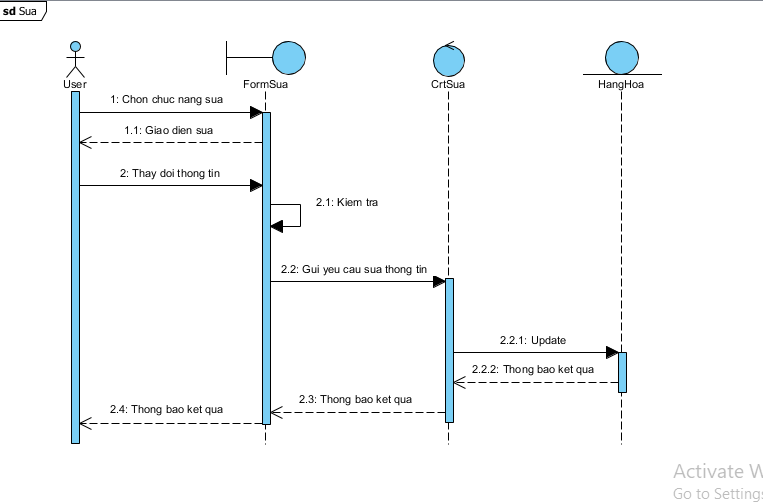
* + 1. Biểu đồ tuần tự

# **Biểu đồ tuần tự Chức năng thêm hàng hóa**



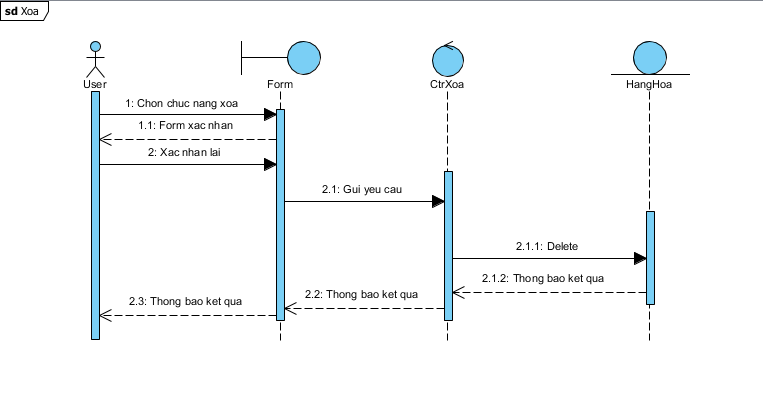
Hình 3- 12: Biểu đồ tuần tự Chức năng thêm hàng hóa

1. **Biểu đồ tuần tự Chức năng sửa thông tin hàng hóa**



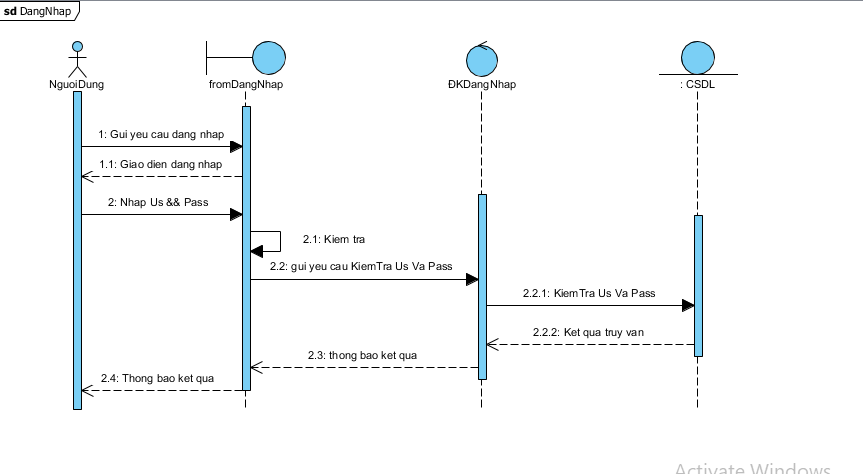
Hình 3- 13:Biểu đồ tuần tự cho chức năng sửa thông tin hàng hóa

1. **Biểu đồ tuần tự Chức năng xóa hàng hóa**



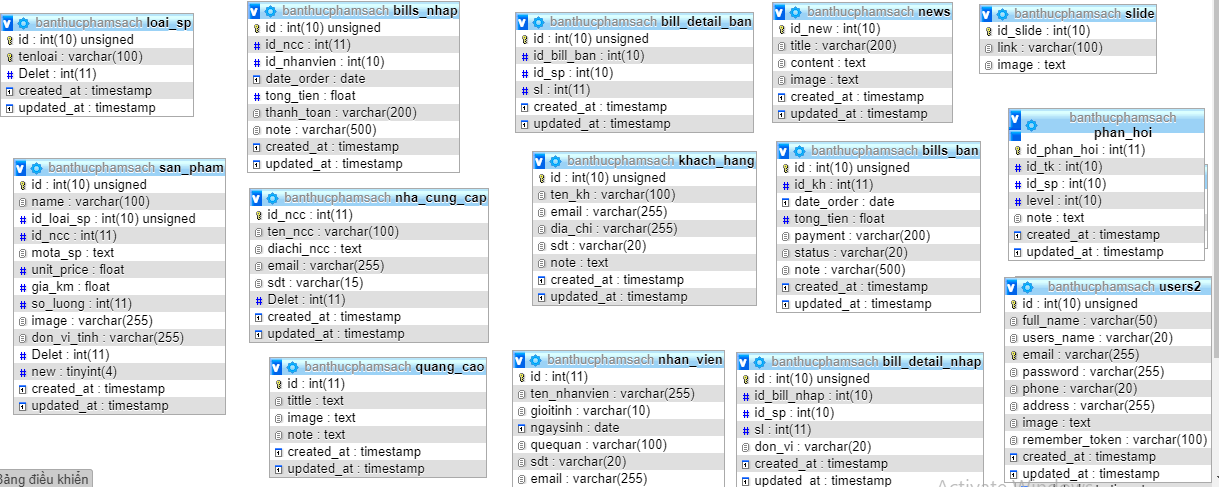
Hình 3- 14:Biểu đồ tuần tự chức năng xóa hàng hóa

1. **Biểu đồ tuần tự cho chức năng đăng nhập**



Hình 3- 15:Biểu đồ tuần tự chức năng đăng nhập

* 1. Thiết kế cơ sở dữ liệu của hệ thống.
     1. Thiết kế cơ sở dữ liệu quản lý nội dung



Hình 3- 16: Mô hình lưu trữ cấu trúc website trên Database

Trong đó các bảng dữ liệu được mô tả như sau:

Bảng 3- 2: Thiết kế bảng loại sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| tenloai | Varchar | 100 |  |  |

Bảng 3- 3: Thiết kế bảng sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| name | Varchar | 100 |  |  |
| id\_loai\_sp | int | 10 | Foreign key |  |
| id\_ncc | int | 10 | Foreign key |  |
| mota\_sp | Text |  |  |  |
| unit\_price | float |  |  |  |
| gia\_km | floai |  |  |  |
| image | Varchar | 255 |  |  |
| don\_vi\_tinh | Varchar | 255 |  |  |
| note | text |  |  |  |
| new | tinyint | 4 |  |  |

Bảng 3- 4:Thiết kế bảng nhà cung cấp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| ten\_ncc | Varchar | 100 |  |  |
| diachi\_ncc | Text |  |  |  |
| email | Varchar | 255 |  |  |
| sdt | Varchar | 50 |  |  |

Bảng 3- 5:Thiết kế bảng nhân viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| ten\_nhan\_vien | Varchar | 255 |  |  |
| gioitinh | Varchar | 10 |  |  |
| ngaysinh | date |  |  |  |
| quequan | varchar | 100 |  |  |
| sdt | varchar | 20 |  |  |
| email | varchar | 255 |  |  |
| capbac | varchar | 10 |  |  |

Bảng 3- 6:Thiết kế bảng khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| ten\_kh | varchar | 100 |  |  |
| email | varchar | 255 |  |  |
| dia\_chi | varchar | 255 |  |  |
| sdt | varchar | 20 |  |  |
| note | text |  |  |  |

Bảng 3- 7:Thiết kế bảng User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| full\_name | varchar | 50 |  |  |
| users\_name | varchar | 50 |  | Unique |
| email | varchar | 255 |  | Unique |
| password | varchar | 255 |  |  |
| phone | varchar | 20 |  | Unique |
| address | varchar | 255 |  |  |
| image | text |  |  |  |
| Remember\_token |  |  |  |  |

Bảng 3- 8:Thiết kế bảng hóa đơn nhập

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| id\_ncc | int | 10 | Foreign key |  |
| id\_nhanvien | int | 10 | Foreign key |  |
| date\_order | date | 255 |  |  |
| tong\_tien | float | 255 |  |  |
| phone | varchar | 20 |  |  |
| thanh\_toan | varchar | 200 |  |  |
| note | varchar | 200 |  |  |

Bảng 3- 9:Thiết kế bảng chi tiết hóa đơn nhập

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| id\_bill\_nhap | int | 10 | Foreign key |  |
| id\_sp | int | 10 | Foreign key |  |
| Sl | int | 11 |  |  |
| don\_vi | varchar | 20 |  |  |

Bảng 3- 10:Thiết kế bảng hóa đơn bán

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| id\_kh | int | 10 | Foreign key |  |
| date\_order | date | 255 |  |  |
| tong\_tien | float | 255 |  |  |
| payment | varchar | 200 |  |  |
| note | varchar | 500 |  |  |

Bảng 3- 11:Thiết kế bảng chi tiết hóa đơn bán

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id | int | 10 | Primary key | Unique,  Auto\_increment |
| id\_bill\_ban | int | 10 | Foreign key |  |
| id\_sp | int | 10 | Foreign key |  |
| Sl | int | 11 |  |  |
| don\_vi | varchar | 20 |  |  |

Bảng 3- 12:Thiết kế bảng phản hồi của khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id\_phan\_hoi | int | 11 | Primary key | Unique,  Auto\_increment |
| id\_tk | int | 10 | Foreign key |  |
| id\_sp | int | 10 | Foreign key |  |
| level | int | 10 |  |  |
| note | text |  |  |  |

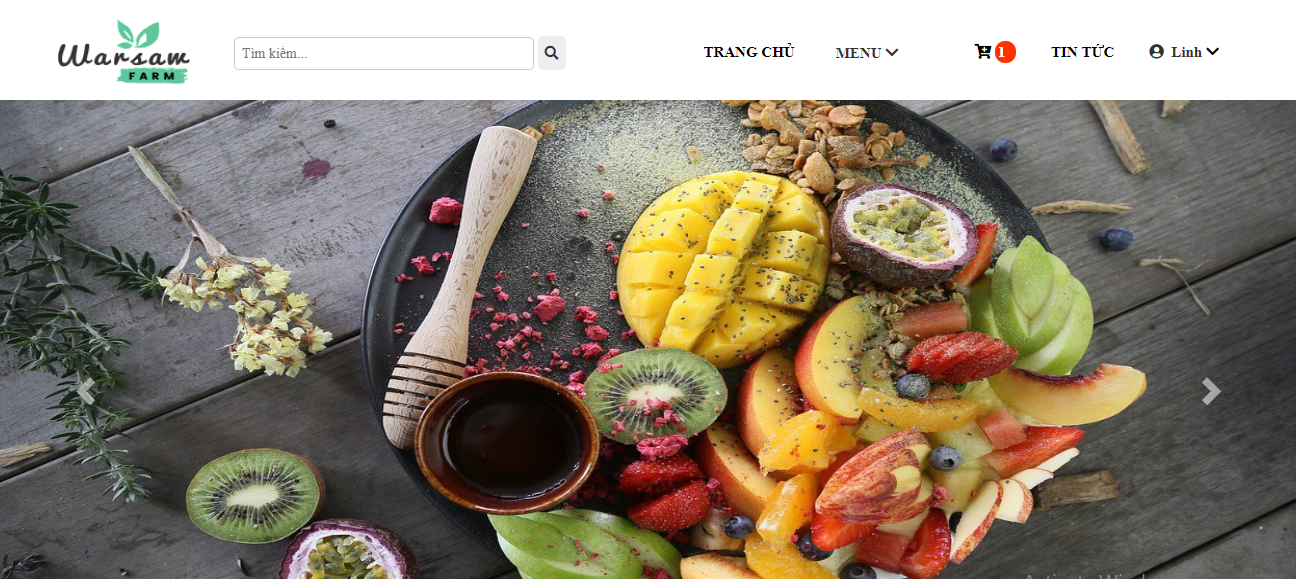
Bảng 3- 13:Thiết kế bảng tin tức

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id\_new | int | 10 | Primary key | Unique,  Auto\_increment |
| title | varchar | 200 |  |  |
| content | text |  |  |  |
| image | text |  |  |  |

Bảng 3- 14:Thiết kế bảng Slide

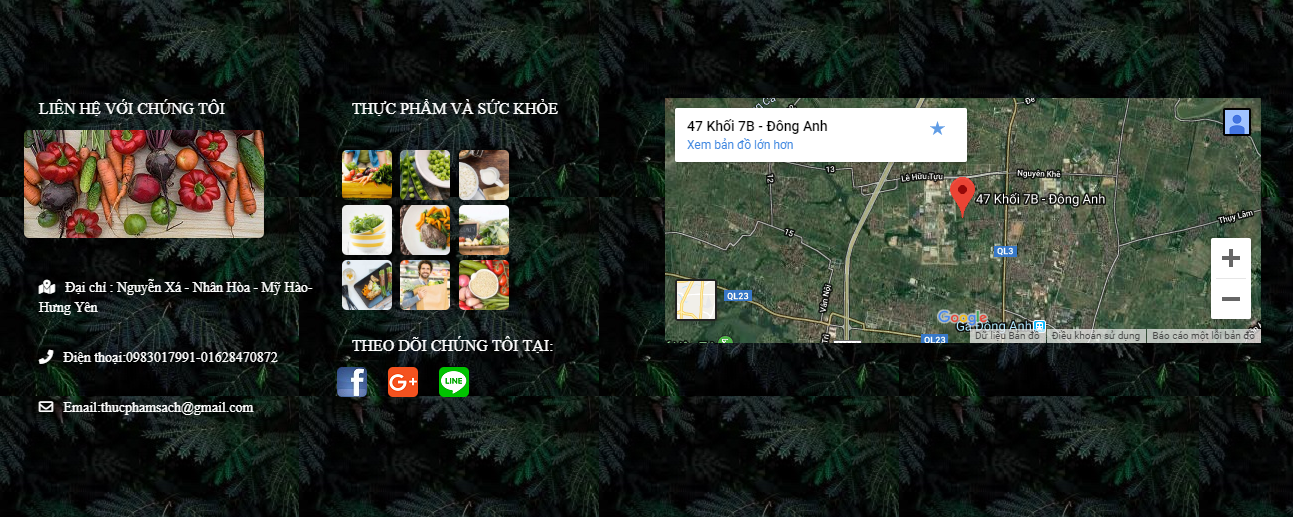
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| id\_slide | int | 10 | Primary key | Unique,  Auto\_increment |
| image | text |  |  |  |

* 1. Giao diện của hệ thống.
     1. Giao diện trang chủ



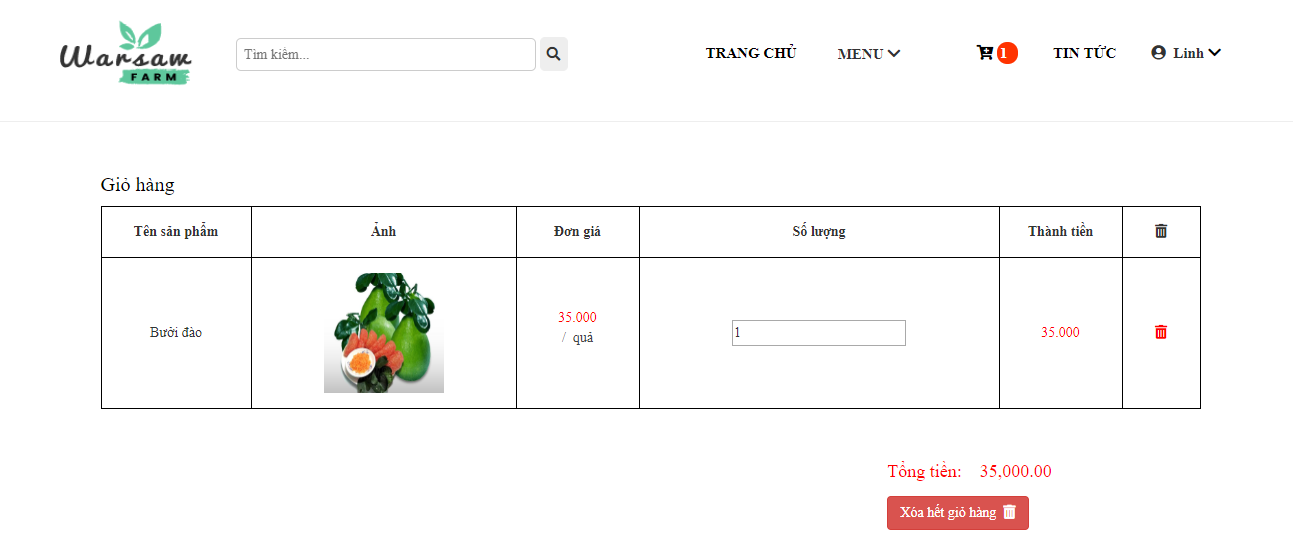


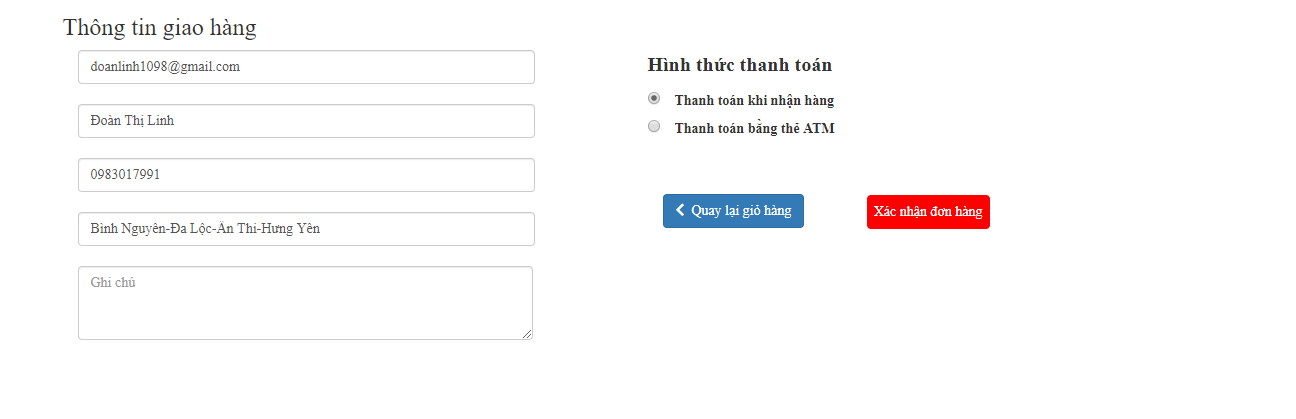




Hình 3- 17:Giao diện trang chủ

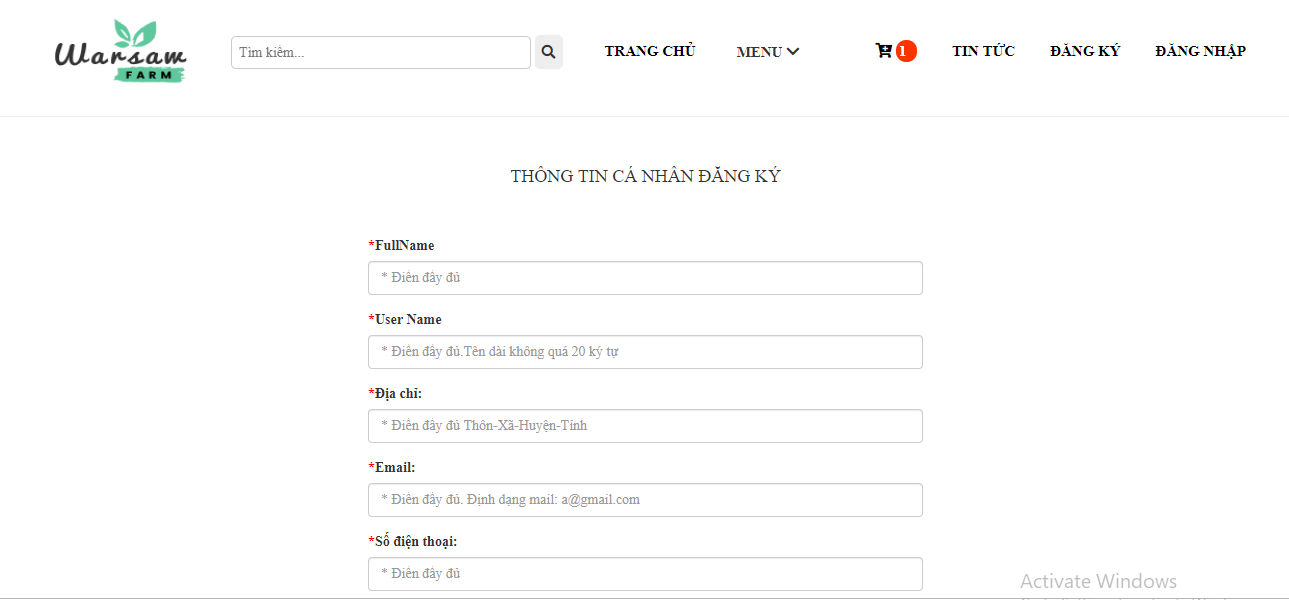
* + 1. Giao diện trang giỏ hàng





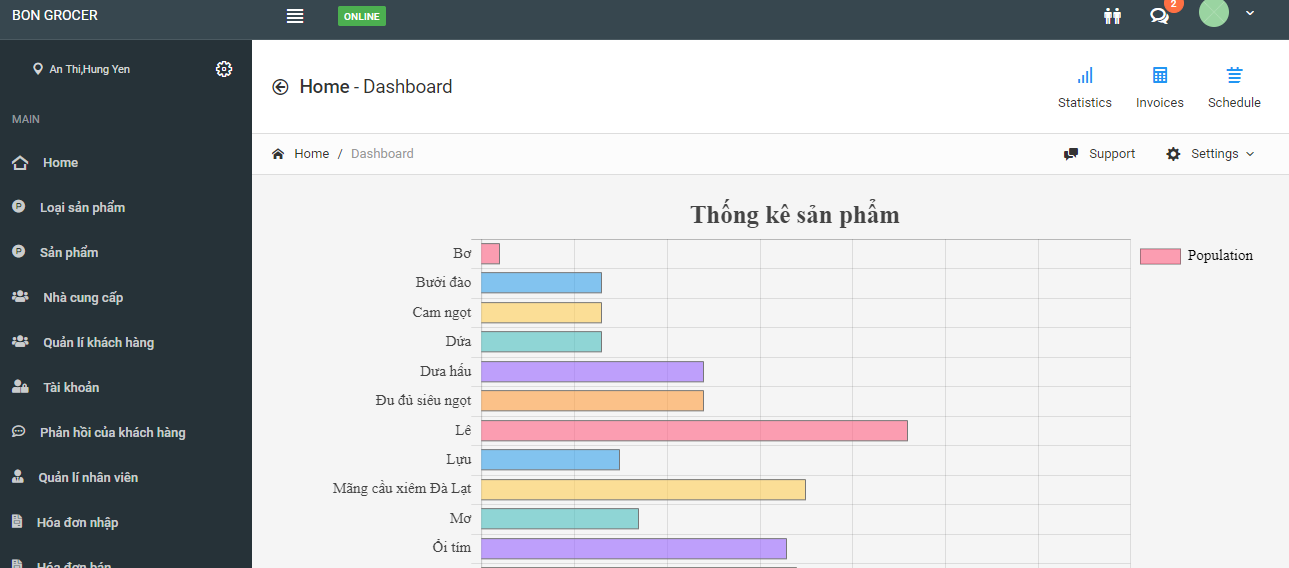
Hình 3- 18:Giao diện trang giỏ hàng

* + 1. Giao diện trang đăng kí



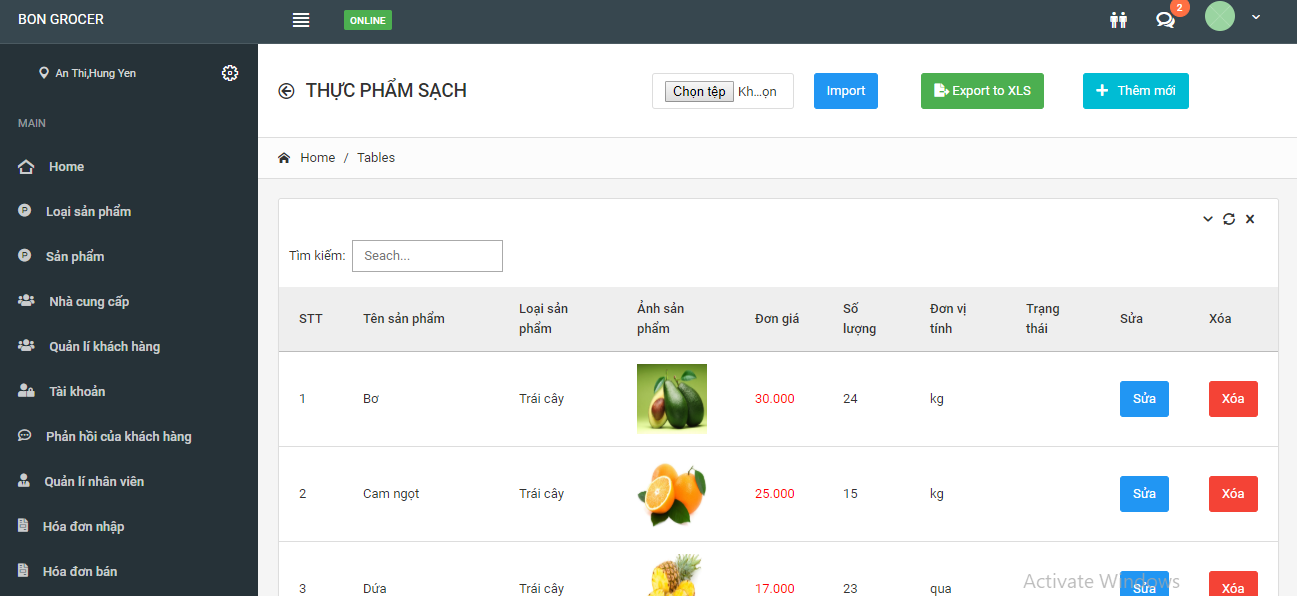
Hình 3- 19:Giao diện trang đăng kí

* + 1. Giao diện trang admin



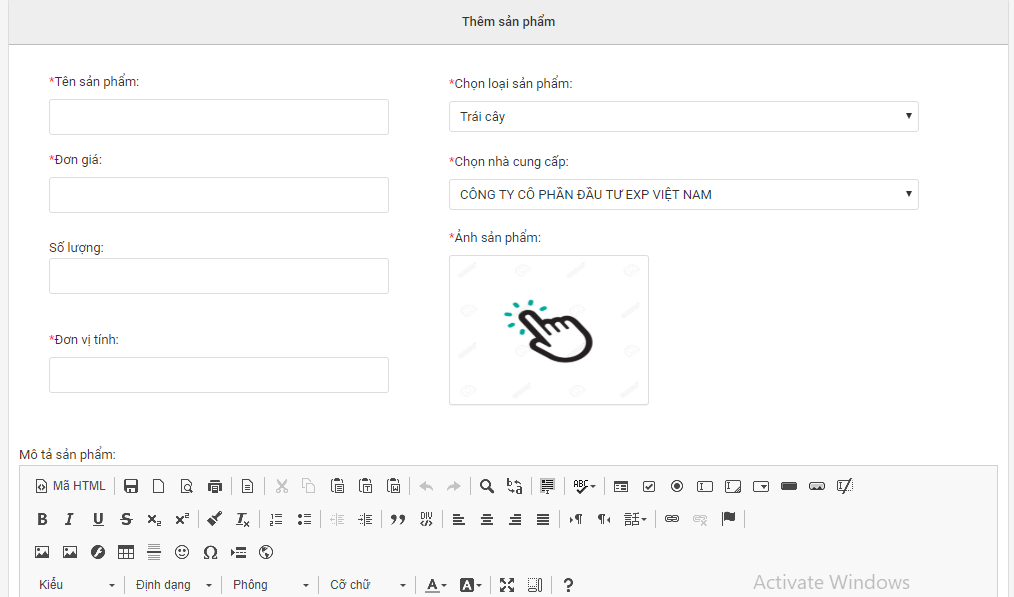
Hình 3- 20:Giao diện trang admin

* + 1. Giao diện trang admin quản lí sản phẩm



Hình 3- 21:Giao diện trang quản lý sản phẩm

* + 1. Giao diện trang admin thêm sản phẩm



Hình 3- 22:Giao diện trang thêm sản phẩm

# **KẾT LUẬN**

* 1. Kết quả đạt được của đề tài

Sau thời gian nghiên cứu và phát triển, em đã hoàn thành cơ bản các mục tiêu ban đầu đề ra của đề tài. Em đã thực hiện nghiên cứu và đề ra giải pháp thiết kế web trên nền tảng PHP, Framework Laravel bao gồm giải pháp về thiết kế cơ sở dữ liệu, quản lý, tổ chức và cấu hình WebSite trên mạng Internet.

* Nắm được quy trình bán hàng trực tuyến.
* Nắm rõ về ngôn ngữ lập trình web , hệ quản trị cơ sở dữ liệu MySql.
* Cơ bản thực hiện tốt chức năng đề ra.
* Giao diện chương trình thiết kế với phong cách hiện đại,chuyên nghiệp.
* Đầy đủ các tính năng cần thiết của website bán hàng.
* Tích hợp bộ lọc và tìm kiếm chính xác.
* Nội dung website được trình bày hợp lý.
* Về mặt giao diện
* Thiết kế được trang web có giao diện thân thiện với người sử dụng.
* Bố cục giao diện WebSite cân đối, hài hòa, phù hợp với nội dung hiển thị của WebSite.
* Màu sắc hài hòa, trang nhã giúp cho người dùng không mỏi mắt khi truy cập vào WebSitetrong thời gian dài.
* Về mặt chức năng WebSite
* Do được xây dựng từ các module đơn lẻ, WebSite có thể dễ dàng nâng cấp các chức năng cho mình trong một thời gian ngắn.
* Nội dung các sản phẩm, tin tức được phân theo loại danh mục rõ ràng.
* Nội dung các tin tức hiển thị ngắn gọn, xúc tích mà vẫn giúp cho người dùng hiểu được đầy đủ được nội dung của tin tức đó.
* Tìm kiếm sản phẩm nhanh chóng qua chức năng tìm kiếm nhanh.
* Về mặt dữ liệu
  + WebSite có khả năng lưu trữ lượng dữ liệu tương đối lớn mà vẫn duy trì được độ ổn định trong quá trình hoạt động.
  + Thông tin mật khẩu của quản trị viên đã được mã hóa.
  1. Hạn chế của đề tài

Hệ thống còn một số hạn chế như:

* Một số chức năng còn chưa hoàn hảo.
* CSDL chưa thực sự tối ưu.
* Chưa tương thích với một số trình duyệt website.

Chưa được chưa có các trợ giúp ngữ cảnh cho người dùng cấu hình tạo mới website.

* 1. Hướng phát triển của đề tài
* Website sẽ chạy ổn định trên nhiều trình duyệt khác nhau.
* Tương thích sử dụng cho các dòng điện thoại, máy tính bảng.
* Xây dựng thêm các chức năng hỗ trợ chat trực tiếp
* Nâng cấp website lên . Có thể thêm nhiều chức năng như luyện thi, trò chuyện trực tiếp với giáo viên.
* Tối ưu hóa hệ thống để có tốc độ nhanh nhất

TÀI LIỆU THAM KHẢO

x

|  |  |
| --- | --- |
| [1] | Website <https://laravel.com/>. |
| [2] | Website <https://vietpro.net.vn/lap-trinh-laravel> |

x

PHỤ LỤC

…..