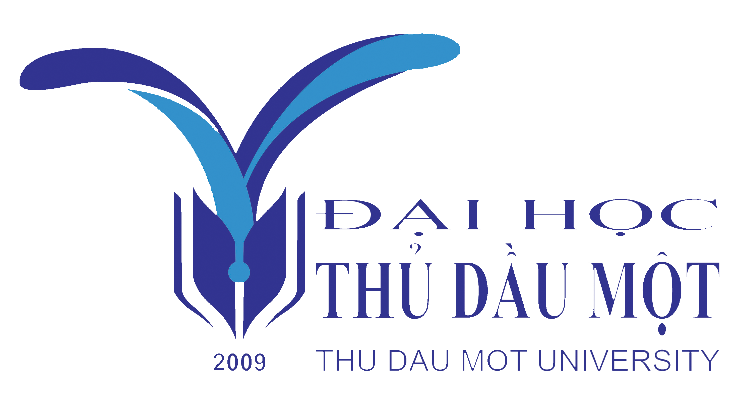
**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**KHOA KỸ THUẬT – CÔNG NGHỆ**



**TIỂU LUẬN**

**XÂY DỰNG HỆ THỐNG WEBSITE**

**BÁN ĐẶC SẢN BƯỞI VIỆT NAM**

**Giáo viên hướng dẫn : ThS. NGUYỄN HỮU VĨNH**

**Sinh viên thực hiện:**

**HỌ VÀ TÊN: ĐOÀN CÔNG CHÍNH**

**MSSV: 1724801040013**

**LỚP: D17HT01**

**Bình Dương, tháng 12 năm 2019**

**NHẬN XÉT VÀ CHẤM ĐIỂM CỦA GIẢNG VIÊN**

Họ và tên giảng viên: **Bùi Sỹ Vương**

Tên tiểu luận: **HỆ THỐNG QUẢN LÝ THÔNG TIN NHÂN SỰ VÀ TIỀN LƯƠNG**

Nội dung nhận xét:

            Điểm:

Bằng số:

Bằng chữ:

*Bình Dương, tháng năm 2020*

**GIẢNG VIÊN**

**ThS. Nguyễn Hữu Vĩnh**

MỤC LỤC

[MỤC LỤC ii](#_Toc28791872)

[MỤC LỤC HÌNH ẢNH v](#_Toc28791873)

[MỤC LỤC BẢNG BIỂU vi](#_Toc28791874)

[LỜI NÓI MỞ ĐẦU 1](#_Toc28791875)

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 2](#_Toc28791876)

[1.1. Mô hình MVC trong lập trình website 2](#_Toc28791877)

[1.1.1. MVC là gì ? 2](#_Toc28791878)

[1.1.2. Các thành phần trong MVC 2](#_Toc28791879)

[1.1.3. MVC làm việc như thế nào? 3](#_Toc28791880)

[1.1.4. Ưu điểm và nhược điểm của MVC 3](#_Toc28791881)

[1.2. Website tĩnh và website động 3](#_Toc28791882)

[1.2.1. Website tĩnh 3](#_Toc28791883)

[1.2.2. Website động 4](#_Toc28791884)

[1.3. Mô hình hướng đối tượng 5](#_Toc28791885)

[1.3.1. Khái niệm 5](#_Toc28791886)

[1.3.2. Các tính chất cơ bản 5](#_Toc28791887)

[1.4. Tổng quan về cơ sở dữ liệu quan hệ 6](#_Toc28791888)

[1.4.1. Khái niệm 6](#_Toc28791889)

[1.4.2. Các ràng buộc trong mô hình quan hệ 6](#_Toc28791890)

[1.4.3. Lịch sử ngắn gọn về cơ sở dữ liệu quan hệ 7](#_Toc28791891)

[1.4.4. Ưu điểm và nhược điểm của cơ sở dữ liệu quan hệ 7](#_Toc28791892)

[1.5. Giới thiệu về SQL Server 7](#_Toc28791893)

[1.5.1. SQL là gì 7](#_Toc28791894)

[1.5.2. Tại sao lại sử dụng SQL trong thiết kế web? 10](#_Toc28791895)

[1.6. Giới thiệu về ASP.NET 10](#_Toc28791896)

[1.6.1. Tìm hiểu về .Net Phatform 10](#_Toc28791897)

[1.6.2. Tìm hiểu về .Net Framework 11](#_Toc28791898)

[1.6.3. Tìm hiểu về ASP.Net 13](#_Toc28791899)

[1.6.4. Những ưu điểm của ASP.Net 14](#_Toc28791900)

[1.6.5. Web Server 14](#_Toc28791901)

[CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 18](#_Toc28791902)

[2.1. Mô tả CSDL Website bán hàng 18](#_Toc28791903)

[2.1.1. Khảo sát hiện trạng 18](#_Toc28791904)

[2.1.2. Yêu cầu đặt ra 18](#_Toc28791905)

[2.1.3. Phát biểu bài toán 18](#_Toc28791906)

[2.2. Các bảng trong CSDL 19](#_Toc28791907)

[2.2.1. Bảng phân tích xác định chức năng, tác nhân và hồ sơ 19](#_Toc28791908)

[2.2.2. Biểu đồ ngữ cảnh 20](#_Toc28791909)

[CHƯƠNG 3. THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG 21](#_Toc28791910)

[3.1. Thiết kế cơ sở dữ liệu 21](#_Toc28791911)

[3.1.1. Mô hình thực thể ER 21](#_Toc28791912)

[3.1.2. Mô hình quan hệ 24](#_Toc28791913)

[3.2. Thiết kế Usecase 28](#_Toc28791914)

[3.2.1. Danh sách các Actor 28](#_Toc28791915)

[3.2.2. Biểu đồ hệ thống 29](#_Toc28791916)

[3.2.3. Danh sách các Usecase 29](#_Toc28791917)

[3.2.4. Biểu đồ Usecase 31](#_Toc28791918)

[3.2.5. Biểu đồ hoạt động các chức năng hệ thống 33](#_Toc28791919)

[CHƯƠNG 4. GIAO DIỆN CHƯƠNG TRÌNH 38](#_Toc28791920)

[4.1. Giao diện trang quản lý 38](#_Toc28791921)

[4.1.1. Trang Index 38](#_Toc28791922)

[4.1.2. Trang quản lý đơn hàng 38](#_Toc28791923)

[4.1.3. Trang quản lý sản phẩm 39](#_Toc28791924)

[4.1.4. Trang thêm sản phẩm 39](#_Toc28791925)

[4.2. Giao diện trang người dùng 40](#_Toc28791926)

[4.2.1. Trang Index 40](#_Toc28791927)

[4.2.2. Trang giới thiệu 41](#_Toc28791928)

[4.2.3. Trang liên hệ 41](#_Toc28791929)

[4.2.4. Trang sản phẩm 42](#_Toc28791930)

[4.2.5. Giao diện trang giỏ hàng 43](#_Toc28791931)

[4.2.6. Giao diện trang tin tức 43](#_Toc28791932)

[4.2.7. Trang chi tiết sản phẩm 44](#_Toc28791933)

[4.2.8. Trang đăng nhập 45](#_Toc28791934)

[4.2.9. Trang đăng ký 45](#_Toc28791935)

[CHƯƠNG 5. KẾT LUẬN 46](#_Toc28791936)

[5.1. Ưu điểm của hệ thống 46](#_Toc28791937)

[5.2. Nhược điểm của hệ thống 46](#_Toc28791938)

[5.3. Hướng phát triển của hệ thống 46](#_Toc28791939)

[TÀI LIỆU THAM KHẢO 47](#_Toc28791940)

MỤC LỤC HÌNH ẢNH

[Hình 1. Sơ đồ mô hình MVC 2](#_Toc28791941)

[Hình 2. SQL Server 7](#_Toc28791942)

[Hình 3. .Net Phatform 11](#_Toc28791943)

[Hình 4. Kiến trúc .Net Famework 11](#_Toc28791944)

[Hình 5. Mô hình IIS with ASP Application 15](#_Toc28791945)

[Hình 6. Cấu hình IIS (1) 16](#_Toc28791946)

[Hình 7. Cấu hình IIS (2) 16](#_Toc28791947)

[Hình 8. Biểu đồ ngữ cảnh của hệ thống 20](#_Toc28791948)

[Hình 9. Các liên kết giữa các thực thể (ER) 23](#_Toc28791949)

[Hình 10. Mô hình thực thể ER (4 thực thể) 24](#_Toc28791950)

[Hình 11. Sơ đồ mối quan hệ dữ liệu 28](#_Toc28791951)

[Hình 12. Biểu đồ hệ thống 29](#_Toc28791952)

[Hình 13. Biểu đồ Usecase khách hàng tiềm năng 31](#_Toc28791953)

[Hình 14. Biêu đồ Usecase quản trị viên và khách hàng 32](#_Toc28791954)

[Hình 15. Biểu đồ hoạt động của Usecase đăng nhập 33](#_Toc28791955)

[Hình 16. Biểu đồ hoạt động Usecase xem sản phẩm 34](#_Toc28791956)

[Hình 17. Biểu đồ hoạt động Usecase quản lý giỏ hàng 35](#_Toc28791957)

[Hình 18. Biểu đồ hoạt động Usecase quản lý đơn hàng 36](#_Toc28791958)

[Hình 19. Biểu đồ hoạt động Usecase xem chi tiết sản phẩm 37](#_Toc28791959)

[Hình 20. Giao diện trang Index quản lý 38](#_Toc28791960)

[Hình 21. Giao diện trang quản lý đơn hàng 38](#_Toc28791961)

[Hình 22. Giao diện trang quản lý sản phẩm 39](#_Toc28791962)

[Hình 23. Giao diện thêm sản phẩm 39](#_Toc28791963)

[Hình 24. Giao diện trang Index người dùng 40](#_Toc28791964)

[Hình 25. Giao diện trang giới thiệu 41](#_Toc28791965)

[Hình 26. Giao diện trang liên hệ 41](#_Toc28791966)

[Hình 27. Giao diện trang sản phẩm 42](#_Toc28791967)

[Hình 28. Giao diện trang giỏ hàng 43](#_Toc28791968)

[Hình 29. Giao diện trang tin tức 43](#_Toc28791969)

[Hình 30. Giao diện trang chi tiết sản phẩm 44](#_Toc28791970)

[Hình 31. Giao diện trang đăng nhập 45](#_Toc28791971)

[Hình 32. Giao diện trang đăng ký 45](#_Toc28791972)

MỤC LỤC BẢNG BIỂU

[Bảng 1. Bảng phân tích chức năng, tác nhân, hồ sơ 19](#_Toc28791973)

[Bảng 2. Bảng thực thể, thuộc tính, khóa của thực thể (ER) 22](#_Toc28791974)

[Bảng 3. Mô tả bảng người dùng 25](#_Toc28791975)

[Bảng 4. Mô tả bảng khuyến mại 25](#_Toc28791976)

[Bảng 5. Mô tả bảng loại sản phẩm 25](#_Toc28791977)

[Bảng 6. Mô tả bảng sản phẩm 25](#_Toc28791978)

[Bảng 7. Mô tả bảng địa chỉ 26](#_Toc28791979)

[Bảng 8. Mô tả bảng giỏ hàng 26](#_Toc28791980)

[Bảng 9. Mô tả bàng chi tiết giỏ hàng 26](#_Toc28791981)

[Bảng 10. Mô tả bảng đơn đặt hàng 26](#_Toc28791982)

[Bảng 11. Mô tả bảng chi tiết đơn đặt hàng 26](#_Toc28791983)

[Bảng 12. Mô tả bảng hình ảnh 27](#_Toc28791984)

[Bảng 13. Mô tả bảng liên hệ 27](#_Toc28791985)

[Bảng 14. Mô tả bảng tin tức 27](#_Toc28791986)

[Bảng 15. Mô tả bảng khách hàng 27](#_Toc28791987)

[Bảng 16. Mô tả bảng địa chỉ nhận hàng 28](#_Toc28791988)

[Bảng 17. Mô tả Actor 28](#_Toc28791989)

[Bảng 18. Danh sách các Usecase 30](#_Toc28791990)

LỜI NÓI MỞ ĐẦU

Trong thời đại kỷ nguyên số ngày nay. Công nghệ thông tin không những là một phương tiện hữu ích mà còn giữ vai trò ngày càng quan trọng trong việc quyết định trình độ phát triển của từng khu vực, từng quốc gia, thông qua việc ngày càng làm tăng năng suất một cách tối đa. Việc áp dụng công nghệ tân tiến trong mặt đời sống là một điều tất yếu.

Với thời đại công nghệ thông tin bùng nổ như hiện nay thì tầm quan trọng website đối với bán hàng ngày càng được khẳng định. Các công ty, doanh nghiệp đã và đang tích cực sử dụng công cụ này để quảng bá thương hiệu của mình tới người tiêu dùng.

Website bán hàng là một bộ phận quan trọng trong cấu thành marketing bán hàng online của một công ty, doanh nghiệp. Đây được xem là xương sống của ngành thương mại điện tử hiện nay đồng thời website chính là nền tảng giúp các doanh nghiệp bắt đầu hoạt động bán hàng trên Internet và thu hút khách hàng.

Khác với cách thức bán hàng truyền thống, việc sử dụng website sẽ giúp cho các công ty, doanh nghiệp tối đa hóa hoạt động bán hàng 24/7. Bất cứ lúc nào ngay cả trong những ngày nghỉ lễ bạn vẫn có thể quảng bá sản phẩm của mình. Việc sử dụng website cũng tạo điều kiện thuận lợi hơn cho các công ty, doanh nghiệp trong việc chăm sóc khách hàng.

Để có thể phát huy hiệu quả của website đối với hoạt động bán hàng, các cửa hàng, doanh nghiệp cần thiết kế website có giao diện đẹp mắt, thân thiện với bố cục hợp lý, phù hợp với ngành nghề kinh doanh, tính năng đầy đủ và thân thiện với người dùng. Bên cạnh đó website còn phải cung cấp thông tin cụ thể, tốc độ tải trang nhanh, tối ưu hóa cho công cụ tìm kiếm và hiển thị tốt trên mọi màn hình thiết bị, trong đó có các thiết bị di động.

Có thể nói với sự phát triển mạnh mẽ của mạng Internet toàn cầu như hiện nay, tầm quan trọng website đối với bán hàng ngày càng được khẳng định, nó đã và đang trở thành một nhân tố tiên quyết giúp các công ty, doanh nghiệp tiến xa trên thị trường đầy cạnh tranh như hiện nay.

# CƠ SỞ LÝ THUYẾT

* 1. Mô hình MVC trong lập trình website
     1. MVC là gì ?

Mô hình MVC (Model – View – Controller) là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. Nó giúp cho các developer tách ứng dụng của họ ra 3 thành phần khác nhau Model, View và Controller. Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.

* + 1. Các thành phần trong MVC

Chúng ta khoan hãy tìm hiểu đến cách thức nó hoạt động mà hãy xem nó gồm những gì! Đây là mô hình MVC



1. Sơ đồ mô hình MVC
   * + 1. Model

Đây là thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý…

* + - 1. View

Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images…Hiểu một cách đơn giản, nó là tập hợp các form hoặc các file HTML.

* + - 1. Controller

Giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng… Chẳng hạn thành phần này sẽ nhận request từ url và form để thao tác trực tiếp với Model.

* + 1. MVC làm việc như thế nào?

Nhìn lại sơ đồ phía trên, ta thấy có mũi tên nét liền và những mũi tên nét đứt. Những mũi tên nét đứt được hình thành trên quan điểm của người dùng mà không phải là của những nhà thiết kế phần mềm thực sự. Do đó chúng ta chỉ quan tâm đến những mũi tên còn lại.

Đây là một cách đơn giản để mô tả lại luồng sự kiện được xử lý trong MVC:

* User tương tác với View, bằng cách click vào button, user gửi yêu cầu đi.
* Controller nhận và điều hướng chúng đến đúng phương thức xử lý ở Model.
* Model nhận thông tin và thực thi các yêu cầu.
* Khi Model hoàn tất việc xử lý, View sẽ nhận kết quả từ Model và hiển thị lại cho người dùng.
  + 1. Ưu điểm và nhược điểm của MVC
       1. 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 phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp, bảo trì..

* + - 1. Nhược điểm:

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

* 1. Website tĩnh và website động
     1. Website tĩnh
        1. Khái niệm

Trang web tĩnh về cơ bản được định nghĩa là một trang web sử dụng hoàn toàn ngôn ngữ chính là HTML (có đuôi html hoặc htm), sau khi tải trang HTML từ máy chủ xuống, trình duyệt sẽ biên dịch mã và hiển thị nội dung trang web, người dùng hầu như sẽ không thể tương tác với trang web được.

* + - 1. Các ngôn ngữ để lập trình website tĩnh

Như đã nói ở trên, website tĩnh sẽ được xây dựng chủ yếu từ ngôn ngữ HTML, bên cạnh đó để thiết kế một website tĩnh đẹp hơn, người ta sẽ sử dụng thêm các ngôn ngữ phụ trợ như CSS và Javascript. Hiện nay khi công nghệ lập trình web ngày càng phát triển, HTML5 và CSS3 dần được sử dụng để thay thế cho HTML và CSS cũ.

Như vậy, một trang web tĩnh hiện nay sẽ được xây dựng bởi 3 ngôn ngữ cơ bản là HTML5, CSS3 và Javascript.

Một website tĩnh thường sẽ có thiết kế khá đẹp và bắt mắt, tuy nhiên do khó khăn về thay đổi nội dung nên nó ít được sử dụng. Website tĩnh chỉ thường được sử dụng ở các đơn vị chuyên về thiết kế website bởi họ đủ điều kiện và khả năng để thay đổi các thông

* + - 1. Ưu điểm của website tĩnh

Vậy website tĩnh có những ưu điểm gì mà dù khó khăn trong việc thay đổi thông tin mà các đơn vị thiết kế website bán hàng chuyên nghiệp cũng vẫn muốn sử dụng? Dưới đây là một số ưu điểm của website tĩnh:

* Tốc độ truy cập nhanh
* Giao diện thiết kế mới lạ
* Chi phí đầu tư thấp
* Thân thiện với bộ máy tìm kiếm
* Ít tốn tài nguyên máy chủ
* Gần như không thể hack
  + - 1. Nhược điểm của website tĩnh
* Khó quản lý nội dung
* Khó nâng cấp bảo trì
* Không có sự tương tác với người dùng
  + 1. Website động
       1. Khái niệm

Là một tập hợp các dữ liệu số hóa được tổ chức thành cơ sở dữ liệu, các dữ liệu số hóa được gọi ra trình diễn trên các trang web dưới dạng văn bản, âm thanh, hình ảnh.nó có thêm các phần xử lý thông tin và truy xuất dữ liệu còn website tĩnh thì không. Khác với web tĩnh, web động luôn luôn có thông tin mới do các thông tin này được cập nhật bởi phần mềm quản trị web do các công ty thiết kế website cung cấp. Các thông tin mới này được lưu vào cơ sở dữ liệu của website và đưa ra sử dụng dựa theo yêu cầu của người dùng. Trang web động được các chuyên gia lập trình, sử dụng các ngôn ngữ lập trình tạo ra mã nguồn dựa theo yêu cầu của trang web.

* + - 1. Lúc nào nên sử dụng web động?
* Bạn làm web tin tức, blog cá nhân
* Web bạn tầm cỡ lớn
* Bạn làm website thương mại điện tử bán hàng
* Bạn làm web giới thiệu sản phảm công ty
  + - 1. Ưu điểm của website động
* Dễ dàng nâng cấp và bảo trì
* Có thể xây dụng được web lớn
* Thường sử dụng tương tác với người dùng cao
* Dễ dàng quản lý nội dung
  + - 1. Nhược điểm của website động
* Nếu web lớn có thể cần thêm nhân sự chuyên ngành
* Chi phí xây dựng cao
  1. Mô hình hướng đối tượng
     1. Khái niệm

Lập trình hướng đối tượng (tiếng Anh: Object-oriented programming, viết tắt: OOP) là một mẫu hình lập trình dựa trên khái niệm "công nghệ đối tượng", mà trong đó, đối tượng chứa đựng các dữ liệu, trên các trường, thường được gọi là các thuộc tính; và mã nguồn, được tổ chức thành các phương thức. Phương thức giúp cho đối tượng có thể truy xuất và hiệu chỉnh các trường dữ liệu của đối tượng khác, mà đối tượng hiện tại có tương tác (đối tượng được hỗ trợ các phương thức "this" hoặc "self"). Trong lập trình hướng đối tượng, chương trình máy tính được thiết kế bằng cách tách nó ra khỏi phạm vi các đối tượng tương tác với nhau.[1][2] Ngôn ngữ lập trình hướng đối tượng khá đa dạng, phần lớn là các ngôn ngữ lập trình theo lớp, nghĩa là các đối tượng trong các ngôn ngữ này được xem như thực thể của một lớp, được dùng để định nghĩa một kiểu dữ liệu.

* + 1. Các tính chất cơ bản

Đối tượng (object): Các dữ liệu và chỉ thị được kết hợp vào một đơn vị đầy đủ tạo nên một đối tượng. Đơn vị này tương đương với một chương trình con và vì thế các đối tượng sẽ được chia thành hai bộ phận chính: phần các phương thức (method) và phần các thuộc tính (attribute / Properties). Trong thực tế, các phương thức của đối tượng là các hàm và các thuộc tính của nó là các biến, các tham số hay hằng nội tại của một đối tượng (hay nói cách khác tập hợp các dữ liệu nội tại tạo thành thuộc tính của đối tượng). Các phương thức là phương tiện để sử dụng một đối tượng trong khi các thuộc tính sẽ mô tả đối tượng có những tính chất gì.

Lập trình hướng đối tượng là một phương pháp lập trình có 4 tính chất chính sau:

1. **Tính trừu tượng (abstraction):** Đây là khả năng của chương trình bỏ qua hay không chú ý đến một số khía cạnh của thông tin mà nó đang trực tiếp làm việc lên, nghĩa là nó có khả năng tập trung vào những cốt lõi cần thiết. Mỗi đối tượng phục vụ như là một "động tử" có thể hoàn tất các công việc một cách nội bộ, báo cáo, thay đổi trạng thái của nó và liên lạc với các đối tượng khác mà không cần cho biết làm cách nào đối tượng tiến hành được các thao tác. Tính chất này thường được gọi là sự trừu tượng của dữ liệu.
2. **Tính đóng gói (encapsulation) và che giấu thông tin (information hiding):** Tính chất này không cho phép người sử dụng các đối tượng thay đổi trạng thái nội tại của một đối tượng. Chỉ có các phương thức nội tại của đối tượng cho phép thay đổi trạng thái của nó. Việc cho phép môi trường bên ngoài tác động lên các dữ liệu nội tại của một đối tượng theo cách nào là hoàn toàn tùy thuộc vào người viết mã. Đây là tính chất đảm bảo sự toàn vẹn của đối tượng.
3. **Tính đa hình (polymorphism):** Thể hiện thông qua việc gửi các thông điệp (message). Việc gửi các thông điệp này có thể so sánh như việc gọi các hàm bên trong của một đối tượng. Các phương thức dùng trả lời cho một thông điệp sẽ tùy theo đối tượng mà thông điệp đó được gửi tới sẽ có phản ứng khác nhau. Người lập trình có thể định nghĩa một đặc tính (chẳng hạn thông qua tên của các phương thức) cho một loạt các đối tượng gần nhau nhưng khi thi hành thì dùng cùng một tên gọi mà sự thi hành của mỗi đối tượng sẽ tự động xảy ra tương ứng theo đặc tính của từng đối tượng mà không bị nhầm lẫn.
4. **Tính kế thừa (inheritance):** Đặc tính này cho phép một đối tượng có thể có sẵn các đặc tính mà đối tượng khác đã có thông qua kế thừa. Điều này cho phép các đối tượng chia sẻ hay mở rộng các đặc tính sẵn có mà không phải tiến hành định nghĩa lại. Tuy nhiên, không phải ngôn ngữ định hướng đối tượng nào cũng có tính chất này.
   1. Tổng quan về cơ sở dữ liệu quan hệ
      1. Khái niệm

Cơ sở dữ liệu quan hệ dựa trên mô hình dữ liệu quan hệ. Trong mô hình này, dữ liệu được tổ chức thành một hoặc nhiều bảng (hoặc "quan hệ") có các cột và hàng. Mỗi hàng được xác định bằng cách sử dụng một khóa duy nhất.

* + 1. Các ràng buộc trong mô hình quan hệ

Các ràng buộc là các điều kiện phải giữ trên tất cả các trạng thái quan hệ hợp lệ. Có ba loại ràng buộc chính trong mô hình quan hệ: Ràng buộc khóa, ràng buộc miền và ràng buộc toàn vẹn.

Ràng buộc khóa là rằng buộc phải có ít nhất một tập hợp con nhỏ nhất của các thuộc tính trong quan hệ để xác định một bộ dữ liệu duy nhất; một thuộc tính khóa không thể có giá trị NULL. Ràng buộc khóa còn được gọi là ràng buộc thực thể.

Ràng buộc tên miền giới hạn phạm vi giá trị miền của một thuộc tính. Nó còn xác định tính cá nhân và liệu một thuộc tính có thể có một giá trị null hay không. Hơn nữa, nó còn có thể chỉ định một giá trị mặc định cho một thuộc tính khi không có giá trị nào được cung cấp.

Ràng buộc toàn vẹn quy định mối quan hệ bảng phải luôn luôn nhất quán. Bất kỳ thay đổi trường khóa chính nào phải được áp dụng cho tất cả các khoá ngoại.

* + 1. Lịch sử ngắn gọn về cơ sở dữ liệu quan hệ

Khái niệm cơ sở dữ liệu quan hệ lần đầu tiên được nhà khoa học máy tính người Anh E. F. Codd đề xuất vào năm 1970. Ông đã phát minh ra mô hình quan hệ để quản lý cơ sở dữ liệu và cơ sở lý thuyết cho cơ sở dữ liệu quan hệ cùng với hệ thống quản lý cơ sở dữ liệu quan hệ khi làm việc cho IBM.

Codd cũng đã định nghĩa 12 quy tắc cấu thành một cơ sở dữ liệu quan hệ cũng như mười hai luật về xử lý phân tích trực tuyến (một thuật ngữ chính ông đặt ra).

* + 1. Ưu điểm và nhược điểm của cơ sở dữ liệu quan hệ

Cơ sở dữ liệu quan hệ có một số ưu điểm so với cơ sở dữ liệu truyền thống.

Tính toàn vẹn dữ liệu - người dùng chỉ cần thay đổi dữ liệu một trong các bảng và sau đó nó sẽ tự cập nhật

Dư thừa dữ liệu - một cơ sở dữ liệu quan hệ sẽ đảm bảo không có thuộc tính nào được lặp lại.

Tính nhất quán - một thuộc tính không thể được lưu trữ dưới định dạng khác trong một tập tin khác.

Tuy nhiên, điều quan trọng cần nhớ là cơ sở dữ liệu quan hệ có thể chậm và không thể mở rộng quá nhiều được.

* 1. Giới thiệu về SQL Server
     1. SQL là gì



1. SQL Server
   * + 1. Khái niệm

SQL Server chính là một hệ quản trị 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, datase engine và các chương trình ứng dụng dùng để quản lý các bộ phận trong RDBMS và những dữ liệu khác.

* + - 1. Lịch sử ra đời và các ấn bản của SQL server
* Lịch sử ra đời SQL server:

Năm 1989, phiên bản đầu tiên của SQL Server 1.0 ra đời được dùng cho các hệ điều hành 16 bit và được phát triển cho tới ngày nay.

Cho tới khi SQL Server ra phiên bản 6.5 thì được thị trường chấp nhận rộng rãi. Một đột phá cải tiến cho SQL Server 7.0 khi được Microsoft viết lại một engine hoàn toàn mới. Đến khi SQL Server từ phiên bản 7.0 cải tiến lên 8.0 chủ yếu phát triển về tính năng thiết kế web.

Cho đến ngày nay thì phiên bản mới nhất đó là SQL Server 2016 hỗ trợ bộ vi xử lý 64 bit ra đời vào ngày 1 tháng 6 năm 2016.

* Một vài ấn bản SQL Server

**Enterprise:** là một ấn bản chứa tất cả các đặc điểm nổ bật của SQL Server như: các công cụ cho tạo và quản lý phân cụm SQL Server, nhân bộ máy cơ sở dữ liệu và một số dịch vụ đi kèm. Nó có thể đánh địa chỉ 12 terabytes và quản lý cơ sở dữ liệu lên tới 524 petabytes.

**Standard:** Ấn bản này có thể chạy tốt trên hệ thống lên tới 4 CPU và 2 GB RAM rất thích hợp cho các dịch vụ thiết kế web vừa và nhỏ.

**Developer:** Ấn bản này giới hạn số lượng người kết nối với server nhưng có đầy đủ các tính năng của Enterprise Edition. Đây là phiên bản được sử dụng cho kiểm tra và phát triển ứng dụng phù hợp cho các cá nhân trong lĩnh vực web như: freelancer Việt Nam.

**Workgroup:** ấn bản SQL Server này có các chức năng lõi cơ sở dữ liệu nhưng không đi kèm các dịch vụ. Ở phiên bản 2012 không có ấn bản này.

**Express:** Ấn bản này dễ dàng sử dụng và quản trị cơ sở dữ liệu đơn giản.

* + - 1. Các thành phần cơ bản trong SQL Server

Các thành cơ bản trong SQL Server gồm có: Reporting Services, Database Engine, Integration Services, Notification Services, Full Text Search Service,… Tất cả kết hợp với nhau tạo thành một giải pháp hoàn chỉnh giúp cho việc phân tích và lưu trữ dữ liệu trở nên dễ dàng hơn.

* **Database Engine:** Đây là một engine có khả năng chứa dữ liệu ở các quy mô dưới dạng support và table. Ngoài ra, nó còn có khả năng tự điều chỉnh ví dụ: trả lại tài nguyên cho ệ điều hành khi một user log off và sử dụng thêm các tài nguyên của máy khi cần.
* **Integration Services:** là tập hợp các đối tượng lập trình và các công cụ đồ họa cho việc sao chép, di chuyển và chuyển đổi dữ liệu. Khi bạn làm việc trong một công ty lớn thì dữ liệu được lưu trữ ở nhiều nơi khác nhau như được chứa trong: Oracle, SQL Server, DB2, Microsoft Access,… và bạn chắc chắn sẽ có nhu cầu di chuyển dữ liệu giữa các server này. Ngoài ra, bạn còn muốn định dạng dữ liệu trước khi lưu vào database. Chắc chắn Integration Services sẽ giúp bạn giải quyết được công việc này dễ dàng.
* **Analysis Services:** Đây là một dịch vụ phân tích dữ liệu rất hay của Microsoft. Dữ liệu khi được lưu trữ vào trong database mà bạn không thể lấy được những thông tin bổ ích thì coi như không có ý nghĩa gì. Chính vì thế, công cụ này ra đời giúp bạn trong việc phân tích dữ liệu một cách hiệu quả và dễ dàng bằng cách dùng kỹ thuật khai thác dữ liệu – datamining và khái niệm hình khối nhiều chiều – multi dimendion cubes.
* **Notification Services:** Dịch vụ thông báo này là nền tảng cho sự phát triển và triển khai các ứng dụng soạn và gửi thông báo. Ngoài ra, dịch vụ này còn có chức năng gửi thông báo theo dịch thời đến hàng ngàn người dăng ký sử dụng trên nhiều loại thiết bị khác nhau.
* **Reporting Services:** là một công cụ tạo, quản lý và triển khai báo cáo bao gồm: server và client. Ngoài ra, nó còn là nền tảng cho việc phát triển và xây dựng các ứng dụng báo cáo.
* **Full Text Search Service:** là một thành phần đặc biệt trong việc truy vấn và đánh chỉ mục dữ liệu văn bản không cấu trúc được lưu trữ trong các cơ sở dữ liệu SQL Server.
* **Service Broker:** là một môi trường lập trình cho việc tạo ra các ứng dụng trong việc nhảy qua các Instance.
  + - 1. Cách cài đặt SQL Server

Muốn cài đặt SQL Server thì bạn cần phải có phiên bản Developer Edition và ít nhất là 500MB ổ cứng cùng với 64 MB Ram và có thể thể cài đặt trên hầu hết các hệ điều hành Windows.

Các bước cài đặt không có gì khó khăn cũng tương tự như các ứng dụng khác. Tuy nhiên trong quá trình cài đặt bạn cần lữu ý những điều sau:

* Sau khi lựa chọn Install Database Server và chọn cài đặt SQL Server thì bạn có thể cài đặt thêm Analysis Service nếu bạn thích.
* Ở màn hình Installation Definition thì bạn nên chọn Server and Client Tools.
* Sau đó bạn nên chọn chọn tất cả các bộ phận của SQL Server và chọn kiểu Custom. Ngoài ra, bạn còn nên chọn các giá trị mặc định – default.
* Sau khi cài đặt thành công, bạn sẽ thấy một icon nằm ở góc phải bên dưới của màn hình, đây chính là Service Manager. Bạn nên lưu ý rằng SQL Server có thể dùng chế độ bảo mật riêng của nó cũng có thể dùng chế độ security của hệ điều hành Windows.
  + 1. Tại sao lại sử dụng SQL trong thiết kế web?

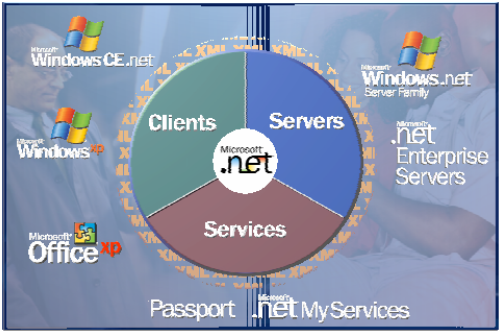
SQL Server không phải là một hệ quản trị cơ sở dữ liệu độc lập mà nó chỉ là một thành phần với vai trò ngôn ngữ là công cụ giao tiếp giữa hệ cơ sở dữ liệu và người dùng. Chính vì thế nó được sử dụng trong các dịch vụ thiết kế web đẹp với chức năng giao tiếp với người dùng với các vai trò sau:

* **SQL là một ngôn ngữ đòi hỏi có tính tương tác cao:** Người dùng có thể dễ dàng trao đổi với các tiện ích thông qua các câu lệnh của SQL đến cơ sở dữ liệu và nhận kết quả từ cơ sở dữ liệu.
* **SQL là một ngôn ngữ lập trình cơ sở dữ liệu:** Các lập trình viên có thể xây dựng các chương trình ứng dụng giao tiếp với cơ sở dữ liệu bằng cách nhúng các câu lệnh SQL vào trong ngôn ngữ lập trình.
* **SQL là một ngôn ngữ lập trình quản trị cơ sở dữ liệu:** Người quản trị cơ sở dữ liệu có thề quản lý, định nghĩa và điều khiển truy cập cơ sở dữ liệu thông qua SQL. SQL là một ngôn ngữ lập trình cho các hệ thống chủ khách:
* **SQL được sử dụng như là một công cụ giao tiếp** với các trình ứng dụng trong hệ thống cơ sở dữ liệu khách chủ.
* **SQL là ngôn ngữ truy cập dữ liệu trên Internet:** SQL được sử dụng với vai trò tương tác với dữ liệu trong hầu hết các máy chủ web và máy chủ Internet.
* **SQL là ngôn ngữ cơ sở dữ liệu phân tán:** Với vai trò giao tiếp với các hệ thống trên mạng, gửi và nhận các yêu cầu truy xuất dữ liệu với nhau.

Nhìn chung SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ được sử dụng trong các chức năng tương tác giữa người dùng và dữ liệu. Chính vì thế, nó được sử dụng trong các dịch vụ thiết kế web rẻ và chuyên nghiệp.

* 1. Giới thiệu về ASP.NET
     1. Tìm hiểu về .Net Phatform

.Net Phatform bao gồm .Net Framework và những công cụ được dùng để xây dựng, phát triển ứng dụng và dịch vụ. ASP.Net.

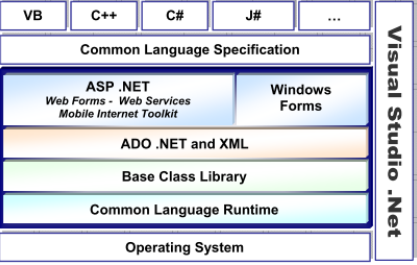


1. .Net Phatform

Những sản phẩm công nghệ .Net của Microsoft bao gồm: MSN.Net, Office.Net, Visual Studio.Net và Windows Server 2003 được biết đến với tên gọi Windows .Net Server.

Visual Studio .Net là bộ phần mềm được dùng để xây dựng và phát triển các ứng dụng bao gồm các ngôn ngữ lập trình: C++.Net, Visual Basic.Net, ASP.Net, C# và J#. Tất cả các ngôn ngữ này được xây dựng dựa trên nền .Net Framework, vì thế, bạn sẽ nhận thấy cú pháp cơ bản của những ngôn ngữ này tương tự nhau.

* + 1. Tìm hiểu về .Net Framework



1. Kiến trúc .Net Famework
   * + 1. Hệ điều hành

Cung cấp các chức năng xây dựng ứng dụng

Với vai trò quản lý việc xây dựng và thi hành ứng dụng, .NET Framework cung cấp các lớp đối tượng (Class) để bạn có thể gọi thi hành các chức năng mà đối tượng đó cung cấp. Tuy nhiên, lời kêu gọi của bạn có được "hưởng ứng" hay không còn tùy thuộc vào khả năng của hệ điều hành đang chạy ứng dụng của bạn.

Các chức năng đơn giản như hiển thị một hộp thông báo (Messagebox) sẽ được .NET framework sử dụng các hàm API của Windows. Chức năng phức tạp hơn như sử dụng các COMponent sẽ yêu cầu Windows phải cài đặt Microsoft Transaction Server (MTS) hay các chức năng trên Web cần Windows phải cài đặt Internet Information Server (IIS).

Như vậy, bạn cần biết rằng lựa chọn một hệ điều hành để cài đặt và sử dụng .NET Framework cũng không kém phần quan trọng. Cài đặt .NET Framework trên các hệ điều hành Windows 2000, 2000 Server, XP, XP.NET, 2003 Server sẽ đơn giản và tiện dụng hơn trong khi lập trình.

* + - 1. Common Language Runtime

Là thành phần "kết nối" giữa các phần khác trong .NET Framework với hệ điều hành. Common Language Runtime (CLR) giữ vai trò quản lý việc thi hành các ứng dụng viết bằng .NET trên Windows. CLR sẽ thông dịch các lời gọi từ chương trình cho Windows thi hành, đảm bảo ứng dụng không chiếm dụng và sử dụng tràn lan tài nguyên của hệ thống. Nó cũng không cho phép các lệnh "nguy hiểm" được thi hành. Các chức năng này được thực thi bởi các thành phần bên trong CLR như Class loader, Just In Time compiler, Garbage collector, Exception handler, COM marshaller, Security engine,…

Trong các phiên bản hệ điều hành Windows mới như XP.NET và Windows 2003, CLR được gắn kèm với hệ điều hành. Điều này đảm bảo ứng dụng viết ra trên máy tính của chúng ta sẽ chạy trên máy tính khác mà không cần cài đặt, các bước thực hiện chỉ đơn giản là một lệnh xcopy của DOS!

* + - 1. Bộ thư viện các lớp đối tượng

Nếu phải giải nghĩa từ "Framework" trong thuật ngữ .NET Framework thì đây là lúc thích hợp nhất.

Framework chính là một tập hợp hay thư viện các lớp đối tượng hỗ trợ người lập trình khi xây dựng ứng dụng. Có thể một số người trong chúng ta đã nghe qua về MFC và JFC. Microsoft Foundation Class là bộ thư viện mà lập trình viên Visual C++ sử dụng trong khi Java Foundation Class là bộ thư viện dành cho các lập trình viên Java. Giờ đây, có thể coi .NET Framework là bộ thư viện dành cho các lập trình viên .NET.

Với hơn 5000 lớp đối tượng để gọi thực hiện đủ các loại dịch vụ từ hệ điều hành, chúng ta có thể bắt đầu xây dựng ứng dụng bằng Notepad.exe!!!… Nhiều người lầm tưởng rằng các môi trường phát triển phần mềm như Visual Studio 98 hay Visual Studio.NET là tất cả những gì cần để viết chương trình. Thực ra, chúng là những phần mềm dùng làm "vỏ bọc" bên ngoài. Với chúng, chúng ta sẽ viết được các đoạn lệnh đủ các màu xanh, đỏ; lỗi cú pháp báo ngay khi đang gõ lệnh; thuộc tính của các đối tượng được đặt ngay trên cửa sổ properties, giao diện được thiết kế theo phong cách trực quan… Như vậy, chúng ta có thể hình dung được tầm quan trọng của .NET Framework.

Nếu không có cái cốt lõi .NET Framework, Visual Studio.NET cũng chỉ là cái vỏ bọc! Nhưng nếu không có Visual Studio.NET, công việc của lập trình viên .NET cũng lắm bước gian nan!

* + - 1. ADO.NET và XML

Bộ thư viện này gồm các lớp dùng để xử lý dữ liệu. ADO.NET thay thế ADO để trong việc thao tác với các dữ liệu thông thường. Các lớp đối tượng XML được cung cấp để bạn xử lý các dữ liệu theo định dạng mới: XML. Các ví dụ cho bộ thư viện này là SqlDataAdapter, SqlCommand, DataSet, XMLReader, XMLWriter,…

* + - 1. ASP.NET

Bộ thư viện các lớp đối tượng dùng trong việc xây dựng các ứng dụng Web. ASP.NET không phải là phiên bản mới của ASP 3.0. Ứng dụng web xây dựng bằng ASP.NET tận dụng được toàn bộ khả năng của .NET Framework. Bên cạnh đó là một "phong cách" lập trình mới mà Microsoft đặt cho nó một tên gọi rất kêu: code behind. Đây là cách mà lập trình viên xây dựng các ứng dụng Windows based thường sử dụng – giao diện và lệnh được tách riêng. Tuy nhiên, nếu bạn đã từng quen với việc lập trình ứng dụng web, đây đúng là một sự "đổi đời" vì bạn đã được giải phóng khỏi mớ lệnh HTML lộn xộn tới hoa cả mắt.

Sự xuất hiện của ASP.NET làm cân xứng giữa quá trình xây dựng ứng dụng trên Windows và Web.

ASP.NET cung cấp một bộ các Server Control để lập trình viên bắt sự kiện và xử lý dữ liệu của ứng dụng như đang làm việc với ứng dụng Windows. Nó cũng cho phép chúng ta chuyển một ứng dụng trước đây viết chỉ để chạy trên Windows thành một ứng dụng Web khá dễ dàng. Ví dụ cho các lớp trong thư viện này là WebControl, HTMLControl, …

* + 1. Tìm hiểu về ASP.Net

Từ khoảng cuối thập niên 90, ASP (Active Server Page) đã được nhiều lập trình viên lựa chọn để xây dựng và phát triển ứng dụng web động trên máy chủ sử dụng hệ điều hành Windows. ASP đã thể hiện được những ưu điểm của mình với mô hình lập trình thủ tục đơn giản, sử dụng hiệu quả các đối tượng COM: ADO (ActiveX Data Object) - xử lý dữ liệu, FSO (File System Object) - làm việc với hệ thống tập tin…, đồng thời, ASP cũng hỗ trợ nhiều ngôn ngữ: VBScript, JavaScript. Chính những ưu điểm đó, ASP đã được yêu thích trong một thời gian dài.

Tuy nhiên, ASP vẫn còn tồn đọng một số khó khăn như Code ASP và HTML lẫn lộn, điều này làm cho quá trình viết code khó khăn, thể hiện và trình bày code không trong sáng, hạn chế khả năng sử dụng lại code. Bên cạnh đó, khi triển khai cài đặt, do không được biên dịch trước nên dễ bị mất source code. Thêm vào đó, ASP không có hỗ trợ cache, không được biên dịch trước nên phần nào hạn chế về mặt tốc độ thực hiện. Quá trình xử lý Postback khó khăn, …

Đầu năm 2002, Microsoft giới thiệu một kỹ thuật lập trình Web khá mới mẻ với tên gọi ban đầu là ASP+, tên chính thức sau này là ASP.Net. Với ASP.Net, không những không cần đòi hỏi bạn phải biết các tag HTML, thiết kế web, mà nó còn hỗ trợ mạnh lập trình hướng đối tượng trong quá trình xây dựng và phát triển ứng dụng Web.

ASP.Net là kỹ thuật lập trình và phát triển ứng dụng web ở phía Server (Server-side) dựa trên nền tảng của Microsoft .Net Framework.

Hầu hết, những người mới đến với lập trình web đều bắt đầu tìm hiểu những kỹ thuật ở phía Client (Client-side) như: HTML, Java Script, CSS (Cascading Style Sheets). Khi Web browser yêu cầu một trang web (trang web sử dụng kỹ thuật client-side), Web server tìm trang web mà Client yêu cầu, sau đó gởi về cho Client. Client nhận kết quả trả về từ Server và hiển thị lên màn hình.

ASP.Net sử dụng kỹ thuật lập trình ở phía server thì hoàn toàn khác, mã lệnh ở phía server (ví dụ: mã lệnh trong trang ASP) sẽ được biên dịch và thi hành tại Web Server. Sau khi được Server đọc, biên dịch và thi hành, kết quả tự động được chuyển sang HTML/JavaScript/CSS và trả về cho Client.

Tất cả các xử lý lệnh ASP.Net đều được thực hiện tại Server và do đó, gọi là kỹ thuật lập trình ở phía server.

* + 1. Những ưu điểm của ASP.Net

ASP.Net cho phép bạn lựa chọn một trong các ngôn ngữ lập trình mà bạn yêu thích: Visual Basic.Net, J#, C#,…

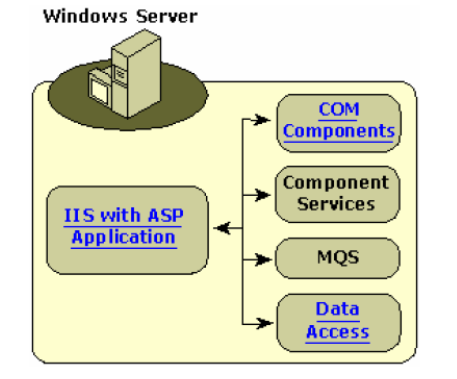
Trang ASP.Net được biên dịch trước. Thay vì phải đọc và thông dịch mỗi khi trang web được yêu cầu, ASP.Net biên dịch những trang web động thành những tập tin DLL mà Server có thể thi hành nhanh chóng và hiệu quả. Yếu tố này là một bước nhảy vọt đáng kể so với kỹ thuật thông dịch của ASP.

* + 1. Web Server

Trong phần này chúng tôi giới thiệu cho bạn về IIS (phần mềm Web Server của Microsot dành cho Windows), đồng thời hướng dẫn bạn cài đặt, cấu hình và kiểm tra Web Server trên các hệ thống sử dụng Windows XP, Windows 2000, Windows Server 2003.

* + - 1. Internet Information Services

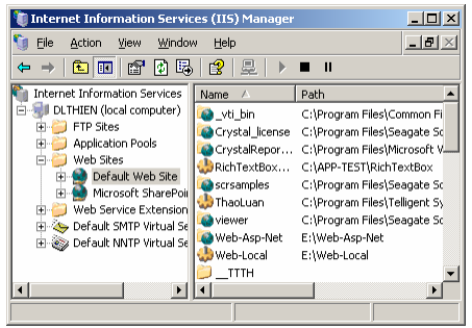
IIS có thể được sử dụng như một Web server, kết hợp với ASP để xây dựng các ứng dụng Web tận dụng các điểm mạnh của Server-side Script, COM component, … theo mô hình Client/Server.



1. Mô hình IIS with ASP Application
   * + 1. Cấu hình Internet Information Services

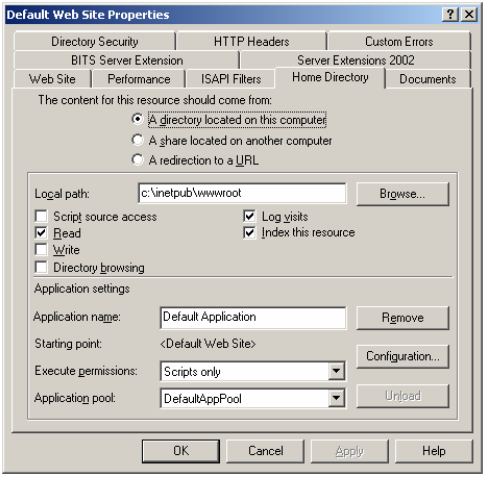
Để cấu hình IIS, vào Control Panel| Administrative Tools|Internet Services Manager.

Trên các hệ điều hành Windows 2000/XP, Microsoft sử dụng công cụ Microsoft Management Console (MMC) để làm công cụ quản lý, do đó tất cả các thao tác đều sử dụng menu ngữ cảnh bằng cách nhắp chuột phải trên mục muốn chọn. Chọn Properties của mục Default Web Site, bạn có thể xem và cấu hình lại các thông tin dành cho trang web mặc định của mình.



1. Cấu hình IIS (1)

Trên tab Home Directory, bạn có thể thay đổi đường dẫn đến một thư mục khác trên ổ cứng nếu muốn



1. Cấu hình IIS (2)

Trên tab Documents, bạn có thể đặt trang web mặc định sẽ hiển thị khi Web Browser không chỉ định trang web cụ thể. Bạn sẽ thấy index.htm và default.htm được liệt kê trong phần này. Đây là lý do tại sao khi bạn gõ //localhost thì Web browser lại hiển thị được trang hompage. Thực ra, //localhost tương đương với //localhost/index.htm hay //localhost/default.htm. Trong tab Directory Security, bạn có thể định lại các chế độ kiểm tra người dùng truy cập vào web site.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

* 1. Mô tả CSDL Website bán hàng
     1. Khảo sát hiện trạng

Hiện tại, công ty đang sử dụng các hình thức giới thiệu sản phẩm trực tiếp như: tờ rơi, aphic, tiếp thị qua điện thoại, mua quảng cáo trên báo, truyền hình, ...

Các hình thức trên đều có nhược điểm là mắt nhiều thời gian, cần nhiều nhân viên và thiết bị để giới thiệu sản phẩm nên hiệu quả mang lại cũng chưa cao. Trừ phương pháp mua quảng cáo rất tiện lợi, nhanh chóng và hiệu quả nhưng chỉ phí lại rất cao.

Từ việc khảo sát và đánh giá hệ thống cũ của công ty. ta thấy việc xây dựng hệ thống mới đẻ giới thiệu sản phẩm của công ty là rất cần thiết. Giải pháp đẻ ra là xây dựng một website có chức năng quản lý và giới thiệu các sản phẩm công nghệ của công ty.

* + 1. Yêu cầu đặt ra

Công ty Đặc Sản Bưởi Việt Nam cần xây dựng một website giới thiệu các sản phẩm công nghệ. Công ty yêu cầu website có các chức năng sau:

* Quảng bá sản phẩm
* Quản lý thông tin các sản phẩm
* Đăng tin tức liên quan đến hoạt động của công ty, tin công nghệ. ...
* Đăng tin tuyên dụng
* Chức năng gửi liên hệ khách với công ty
* Quản trị các nội dung thêm trên trang website (phẩn giới thiệu công ty, logo công Ly, quảng cáo, các đối tác, ... )
  + 1. Phát biểu bài toán

Người quản trị viên xem bảng thông tin sản phẩm đẻ cập nhật thông tin sản phẩm, xem các quyết định tuyển dụng đẻ cập nhật tin tuyển dụng, xem bảng tin để cập nhật tin tức lên hệ thống. Người quản trị sẽ nhận được thông báo xác nhận thành công khi đăng tin hoặc cập nhật thành công.

Hằng ngày, khách sẽ truy cập vào website của công ty đẻ xem thông tin vẻ công ty, về các sản phẩm, tin tức của công ty.

Khách cũng có thể tìm kiếm thông tin trên website để nhanh chóng nhận được các thông tin về sản phẩm, tin tức mình cần.

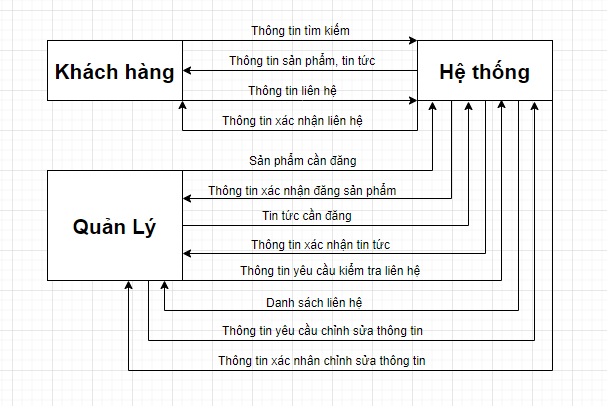
Khi cần liên hệ (cần tư vấn, mua bán) với công ty, khách sẽ đùng chức năng liên hệ có sẵn trên website để gửi các thông tin liên hệ. Khách sẽ nhận được thông báo xác nhân khi gửi thành công. Người quản lý sẽ kiểm tra danh sách các liên hệ đã nhận.

Người quản lý có thẻ chỉnh sửa các thông tin thêm như phần giới thiệu công ty, logo công ty, quảng cáo, các đối tác, ... Người quản lý sẽ nhân được thông báo chỉnh sửa thành công.

* 1. Các bảng trong CSDL
     1. Bảng phân tích xác định chức năng, tác nhân và hồ sơ

|  |  |  |
| --- | --- | --- |
| **Động từ + Bổ ngữ** | **Tác nhân** | **Nhận xét** |
| Xem bảng thông tin sản phẩm | Người quản trị | HSDL |
| Cập nhập thông tin sản phẩm | Người quản trị | Tác nhân |
| Xem bảng tin | Người quản trị | HSDL |
| Cập nhập tin tức | Người quản trị | Tác nhân |
| Xem thông tin về công ty, các sản phẩm của công ty | Khách | Tác nhân |
| Tìm kiếm thông tin | Khách | Tác nhân |
| Nhận các thông tin của sản phẩm, tin tức | Khách | Tác nhân |
| Gửi thông tin liên hệ | Khách | Tác nhân |
| Nhận được thông báo xác nhận liên hệ | Khách | Tác nhân |
| Kiểm tra danh sách liên hệ | Người quản trị | HSDL |
| Chỉnh sửa các thông tin thêm | Người quản trị | Tác nhân |

1. Bảng phân tích chức năng, tác nhân, hồ sơ
   * 1. Biểu đồ ngữ cảnh



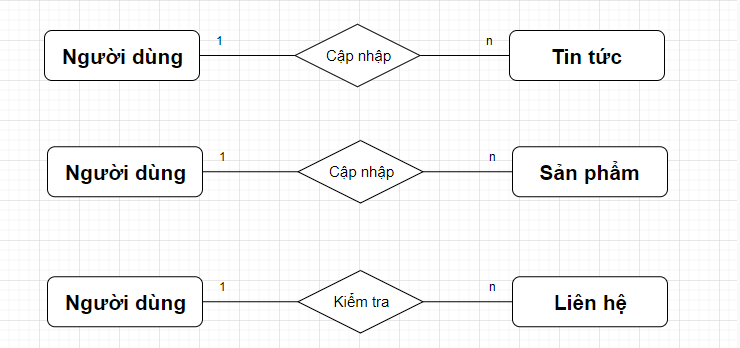
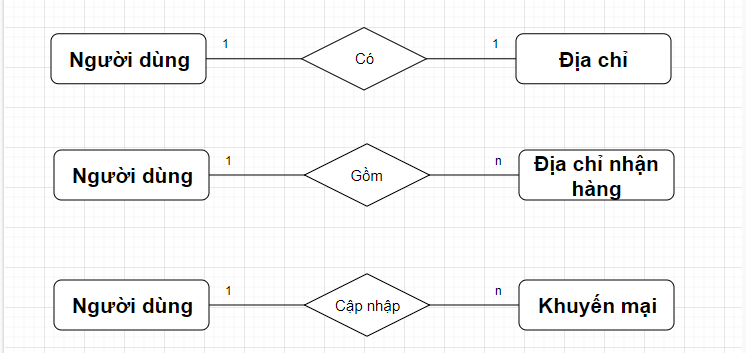
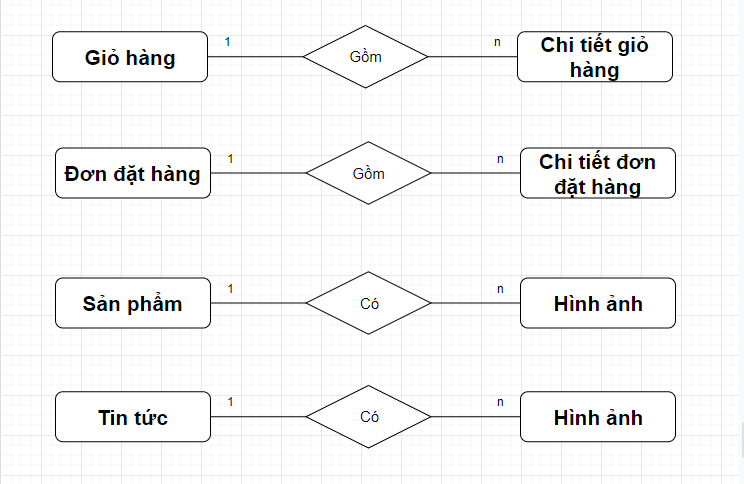
1. Biểu đồ ngữ cảnh của hệ thống

# THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

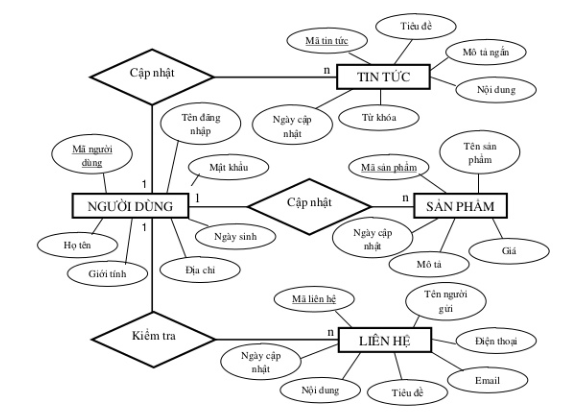
* 1. Thiết kế cơ sở dữ liệu
     1. Mô hình thực thể ER
        1. Xác định các kiểu thực thể, các thuộc tính khóa của thực thể

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Kiểu thực thể** | **Thuộc tính** | **Thuộc tính khóa** |
| 1 | Người dùng | Mã người dùng, tên đăng nhập, Mật khẩu, Tên họ, tên, địa chỉ email, Số điện thoại, quyền | Mã người dùng, tên đăng nhập |
| 2 | Tin tức | Mã tin tức, nội dung, ngày viết, tiêu đề, mã hình ảnh, trạng thái | Mã tin tức |
| 3 | Sản phẩm | Mã sản phẩm, tên sản phẩm, mô tả, mã loại sản phẩm, mã địa chỉ, đơn giá | Mã sản phẩm |
| 4 | Loại sản phẩm | Mã loại sản phẩm, tên loại sản phẩm | Mã loại sản phẩm |
| 5 | Liên hệ | Mã liên hệ, họ tên, email, sđt, lời nhắn, trạng thái | Mã liên hệ |
| 6 | Khách hàng | Mã khách hàng, tên, địa chỉ, sđt, email | Mã khách hàng |
| 7 | Hình ảnh | Mã hình ảnh, đường link hình ảnh | Mã hình ảnh |
| 8 | Giỏ hàng | Mã giỏ hàng, mã người dùng | Mã giỏ hàng |
| 9 | Chi tiết giỏ hàng | Mã chi tiết giỏ hàng, mã giỏ hàng, mã sản phẩm, số lương, đơn giá, ngày thêm | Mã chi tiết giỏ hàng |
| 10 | Đơn đặt hàng | Mã đơn đặt hàng, mã khách hàng, loại khách hàng | Mã đơn đặt hàng |
| 11 | Chi tiết đơn đặt hàng | Mã chi tiết đơn đặt hàng, mã sản phẩm, số lương, đơn giá, ngày đặt hàng, mã đơn đặt hàng | Mã chi tiết đơn đặt hàng |
| 12 | Địa chỉ | Mã địa chỉ, tên tỉnh | Mã địa chỉ |
| 13 | Địa chỉ nhận hàng | Mã địa chỉ nhận hàng, huyện, tỉnh, mã người dùng | Mã địa chỉ nhận hàng |
| 14 | Khuyến mại | Mã thông tin khuyến mại, mã sản phẩm, mô tả, giá giảm, ngày bắt đầu, ngày kết thúc | Mã thông tin khuyến mại |

1. Bảng thực thể, thuộc tính, khóa của thực thể (ER)
   * + 1. Xác định các liên kết

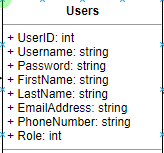
1. Các liên kết giữa các thực thể (ER)
   * + 1. Mô hình ER (4 thực thể)



1. Mô hình thực thể ER (4 thực thể)
   * 1. Mô hình quan hệ
        1. Xác định quan hệ

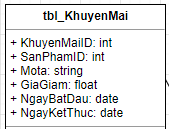
* Người dùng (**Mã người dùng**, **tên đăng nhập**, Mật khẩu, Tên họ, tên, địa chỉ email, Số điện thoại, quyền)
* Tin tức(**Mã tin tức**, nội dung, ngày viết, tiêu đề, mã hình ảnh, trạng thái)
* Sản phẩm(**Mã sản phẩm**, tên sản phẩm, mô tả, mã loại sản phẩm, mã địa chỉ, đơn giá)
* Loại sản phẩm(**Mã loại sản phẩm**, tên loại sản phẩm)
* Liên hệ(**Mã liên hệ**, họ tên, email, sđt, lời nhắn, trạng thái)
* Khách hàng(**Mã khách hàng**, tên, địa chỉ, sđt, email)
* Hình ảnh(**Mã hình ảnh**, đường link hình ảnh)
* Giỏ hàng(**Mã giỏ hàng,** mã người dùng)
* Chi tiết giỏ hàng(**Mã chi tiết giỏ hàng**, mã giỏ hàng, mã sản phẩm, số lương, đơn giá, ngày thêm)
* Đơn đặt hàng(**Mã đơn đặt hàng**, mã khách hàng, loại khách hàng)
* Mã chi tiết đơn đặt hàng(**Mã chi tiết đơn đặt hàng**, mã sản phẩm, số lương, đơn giá, ngày đặt hàng, mã đơn đặt hàng)
  + - 1. Mô tả các bảng

Bảng người dùng (tbl\_users) dùng để lưu trữ thông tin người dùng



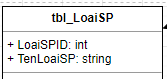
1. Mô tả bảng người dùng

Bảng khuyễn mại (tbl\_KhuyenMai) dùng để lưu trữ thông tin khuyến mại



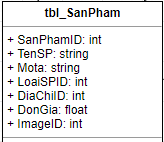
1. Mô tả bảng khuyến mại

Bảng loại sản phẩm (tbl\_loaiSP) dùng để lưu trữ thông tin loại sản phẩm



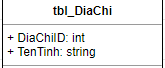
1. Mô tả bảng loại sản phẩm

Bảng Sản phẩm (tbl\_SanPham) dùng để lưu trữ thông tin sản phẩm



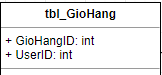
1. Mô tả bảng sản phẩm

Bảng địa chỉ (tbl\_DiaChi) dùng để lưu trữ thông tin địa chỉ xuất xứ của sản phẩm



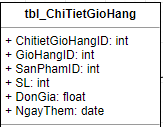
1. Mô tả bảng địa chỉ

Bảng giỏ hàng (tbl\_GioHang) dùng để lưu trữ thông tin giỏ hàng của khách hàng



1. Mô tả bảng giỏ hàng

Bảng chi tiết giỏ hàng (tbl\_ChiTietGioHang) dùng đẻ lưu trữ thông tin của giỏ hàng



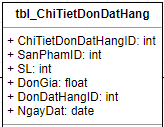
1. Mô tả bàng chi tiết giỏ hàng

Bảng Đơn đặt hàng (tbl\_DonDatHang) dùng để lưu trữ đơn đặt hàng của khách hàng



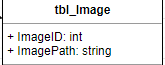
1. Mô tả bảng đơn đặt hàng

Bảng Chi tiết đơn đặt hàng (tbl\_ChiTietDonDatHang) dùng để lưu trữ chi tiết sản phẩm của đơn đặt hàng



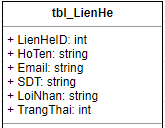
1. Mô tả bảng chi tiết đơn đặt hàng

Bảng Hình ảnh (tbl\_Image) đùng để lưu trữ thông tin của hình ảnh



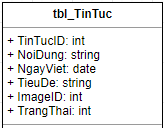
1. Mô tả bảng hình ảnh

Bảng Liên hệ (tbl\_LienHe) dùng để lưu trữ thông tin liên hệ



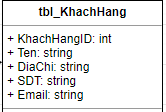
1. Mô tả bảng liên hệ

Bảng Tin tức (tbl\_TinTuc) dùng để lưu trữ thông tin tin tức



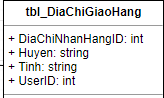
1. Mô tả bảng tin tức

Bảng Khách hàng (tbl\_KhachHang) dùng để lưu trữ thông tin mua hàng của khách hàng lãng vai

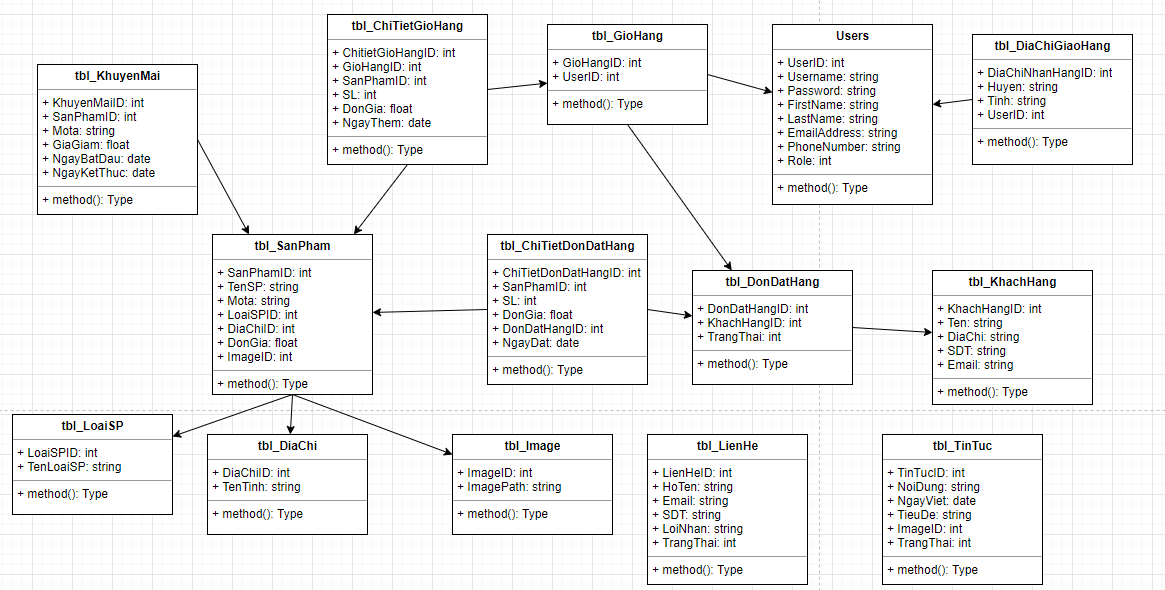


1. Mô tả bảng khách hàng

Bảng địa chỉ nhận hàng (tbl\_DiaChiGiaoHang) dùng để lưu trữ thông tin địa chỉ giao hàng



1. Mô tả bảng địa chỉ nhận hàng
   * + 1. Mô hình quan hệ

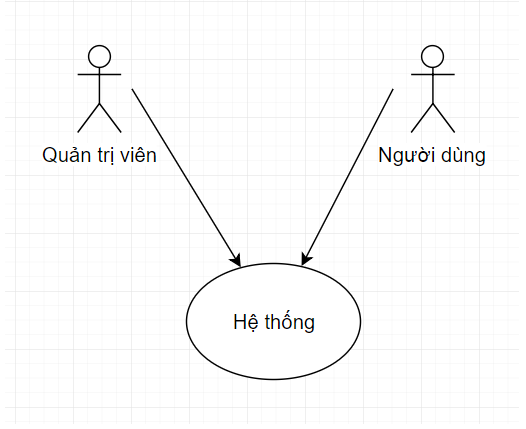


1. Sơ đồ mối quan hệ dữ liệu
   1. Thiết kế Usecase
      1. Danh sách các Actor

Một actor hay một tác nhân ngoài là một vai trò của một hay nhiều người hay vật thể trong sự tương tác với hệ thống

|  |  |  |
| --- | --- | --- |
| **STT** | **Tác nhân** | **Mô tả** |
| 1 | Quản trị viên | Là người quản trị hệ thống, có toàn quyền trên hệ thống |
| 2 | Người dùng | Là người sử dụng hệ thống để mua, đọc tin tức |

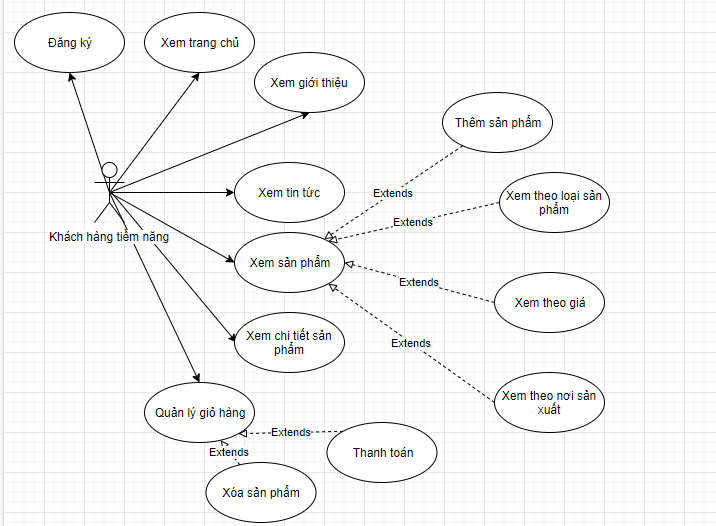
1. Mô tả Actor
   * 1. Biểu đồ hệ thống



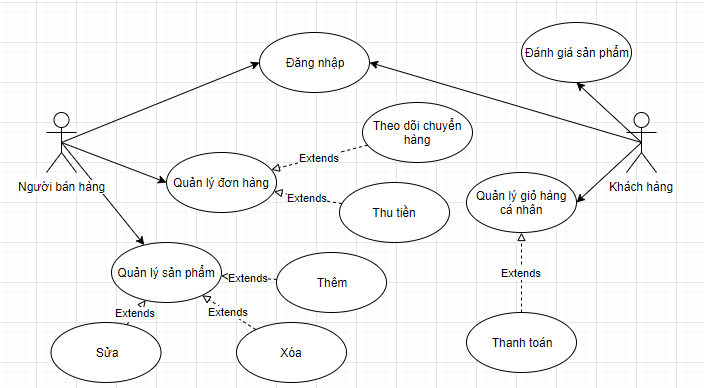
1. Biểu đồ hệ thống
   * 1. Danh sách các Usecase

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Use case** | **Ý nghĩa** | **Actor** |
| 1 | Đăng nhập | Cho phép người dùng và quản lý truy cập vào hệ thống | Quản trị viên, người dùng |
| 2 | Đăng ký | Cho phép người dùng đăng ký thành khách hàng thân thiết | Quản trị viên, khách lãng vai |
| 3 | Xem trang chủ | Cho phép người dùng xem trang chủ website | Quản trị viên, người dùng, khách lãng vai |
| 4 | Xem giới thiệu | Cho phép người dùng xem trang giới thiệu công ty | Quản trị viên, người dùng, khách lãng vai |
| 5 | Xem tin tức | Cho phép người dùng xem trang tin tức của website | Quản trị viên, người dùng, khách lãng vai |
| 6 | Xem sản phẩm | Cho phép người dùng xem danh sách sản phẩm website | Quản trị viên, người dùng, khách lãng vai |
| 7 | Xem chi tiết sản phẩm | Cho phép người dùng xem chi tiết sản phẩm website | Quản trị viên, người dùng, khách lãng vai |
| 8 | Quản lý giỏ hàng | Cho phép người dùng xem danh sách sản phẩm đã thêm | Quản trị viên, người dùng, khách lãng vai |
| 9 | Thanh toán | Cho phép người dùng thanh toán | Quản trị viên, người dùng, khách lãng vai |
| 10 | Quản lý đơn hàng | Cho phép người dùng xem đơn hàng đã đặt | Quản trị viên |
| 11 | Quản lý sản phẩm | Cho phép người dùng xem sản phảm đã thêm | Quản trị viên |
| 12 | Quản lý tin tức | Cho phép người dùng xem tin tức đã thêm | Quản trị viên |
| 13 | Quản lý khuyến mại | Cho phép người dùng xem khuyến mại đã thêm | Quản trị viên |
| 14 | Thêm, sửa sản phẩm | Cho phép người dùng thêm / sửa sản phẩm | Quản trị viên |
| 15 | Xóa sản phẩm | Cho phép người dùng xóa sản phẩm | Quản trị viên |
| 16 | Thêm sản phẩm vào giỏ hàng | Cho phép người dùng thêm sản phẩm vào giỏ hàng | Quản trị viên, người dùng, khách lãng vai |
| 17 | Xóa sản phẩm khỏi giỏ hàng | Cho phép người dùng xóa sản phẩm khỏi giỏ hàng | Quản trị viên, người dùng, khách lãng vai |

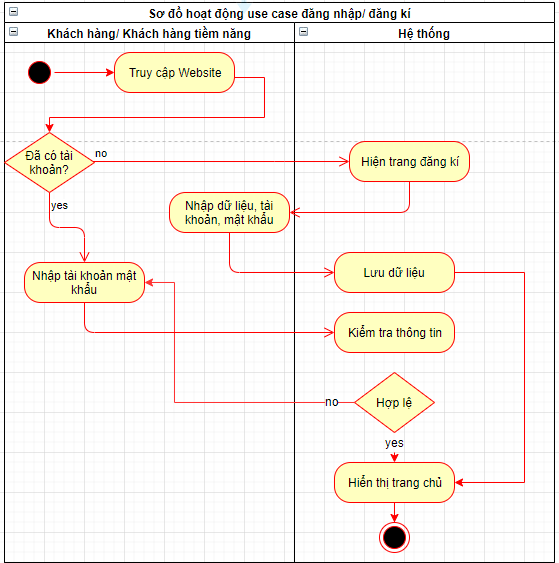
1. Danh sách các Usecase
   * 1. Biểu đồ Usecase
        1. Biểu đồ Usecase khách hàng tiềm năng (khách lãng vai)



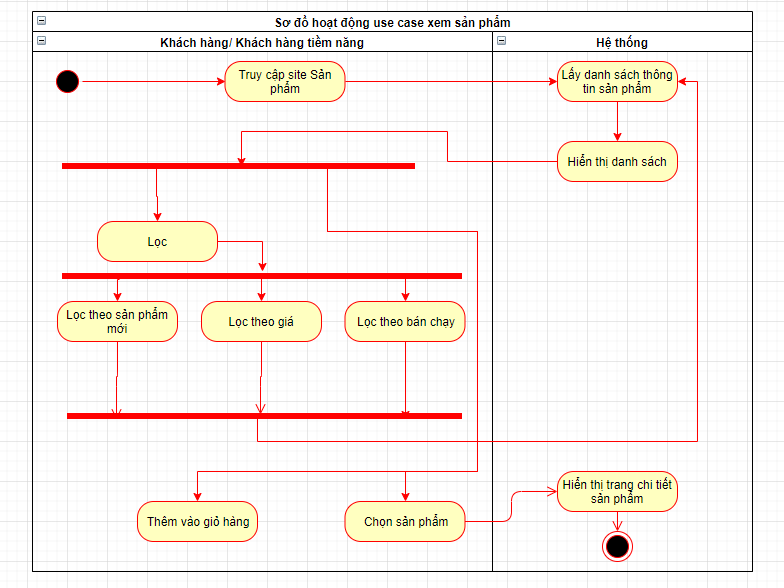
1. Biểu đồ Usecase khách hàng tiềm năng
   * + 1. Biểu đồ Usecase quản trị viên và khách hàng



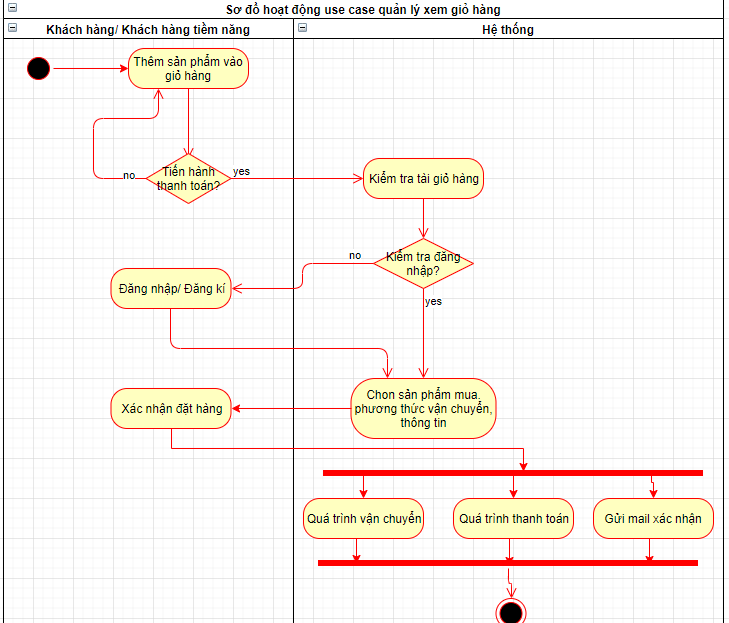
1. Biêu đồ Usecase quản trị viên và khách hàng
   * 1. Biểu đồ hoạt động các chức năng hệ thống
        1. Usecase đăng nhập



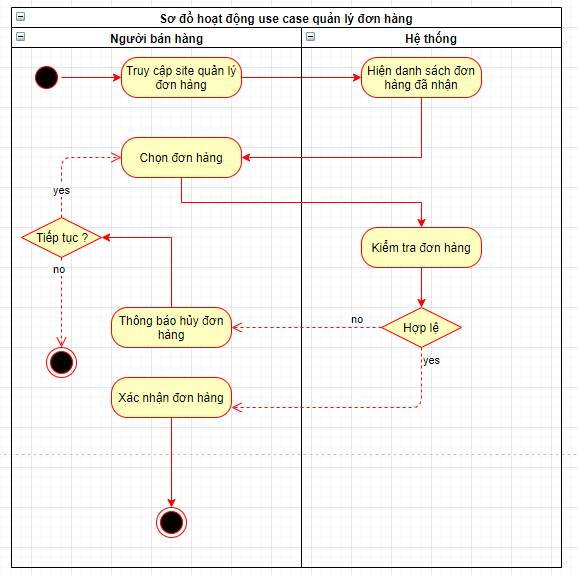
1. Biểu đồ hoạt động của Usecase đăng nhập
   * + 1. Usecase xem sản phẩm



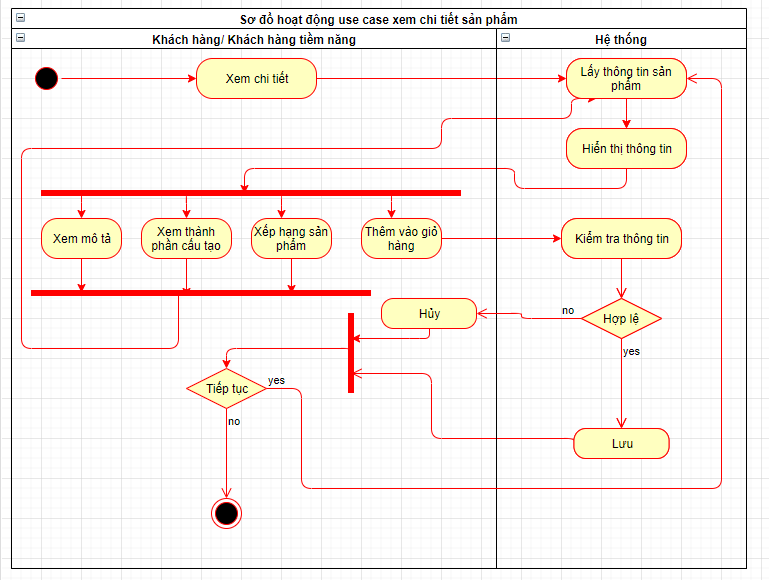
1. Biểu đồ hoạt động Usecase xem sản phẩm
   * + 1. Usecase quản lý giỏ hàng



1. Biểu đồ hoạt động Usecase quản lý giỏ hàng
   * + 1. Usecase quản lý đơn hàng



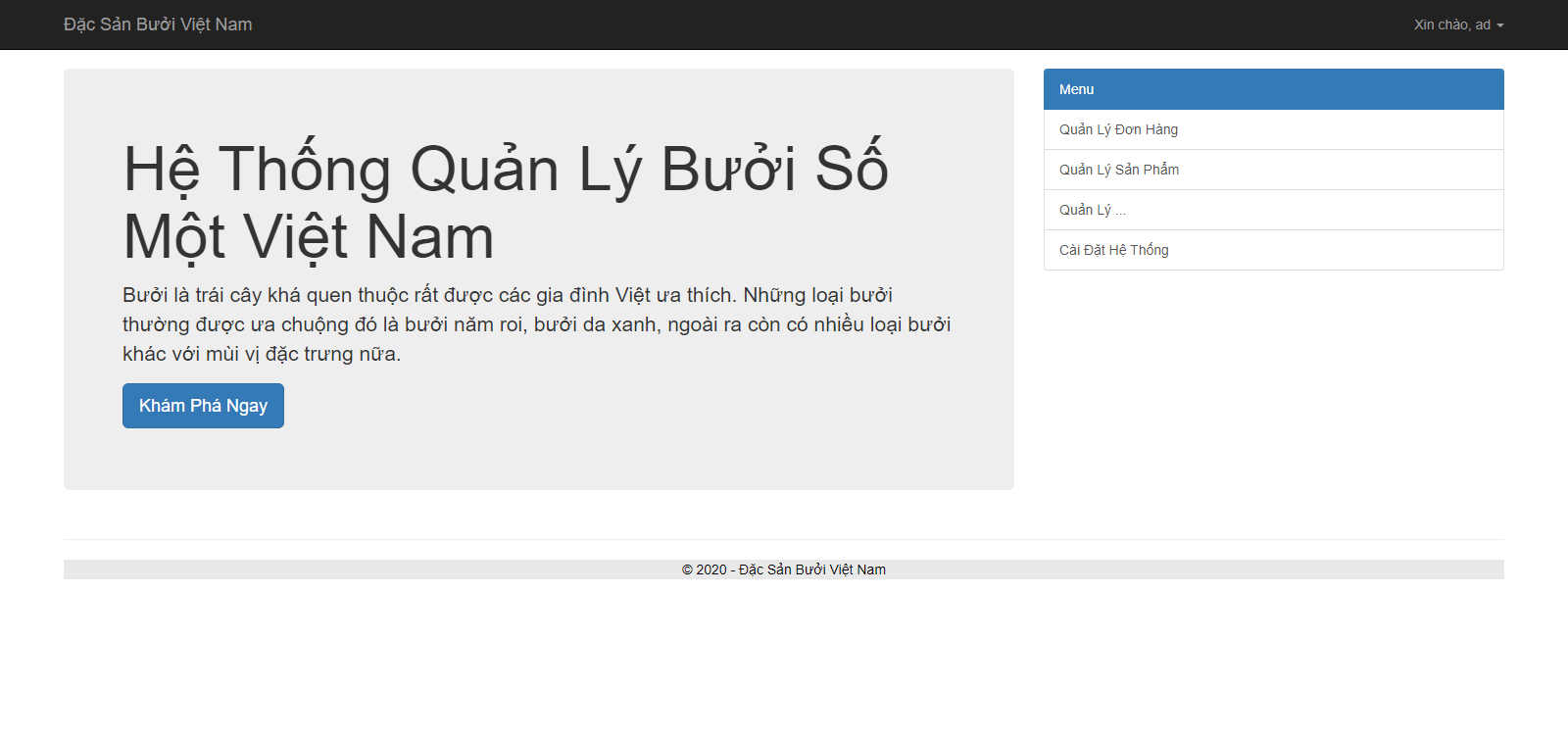
1. Biểu đồ hoạt động Usecase quản lý đơn hàng
   * + 1. Usecase xem chi tiết sản phẩm



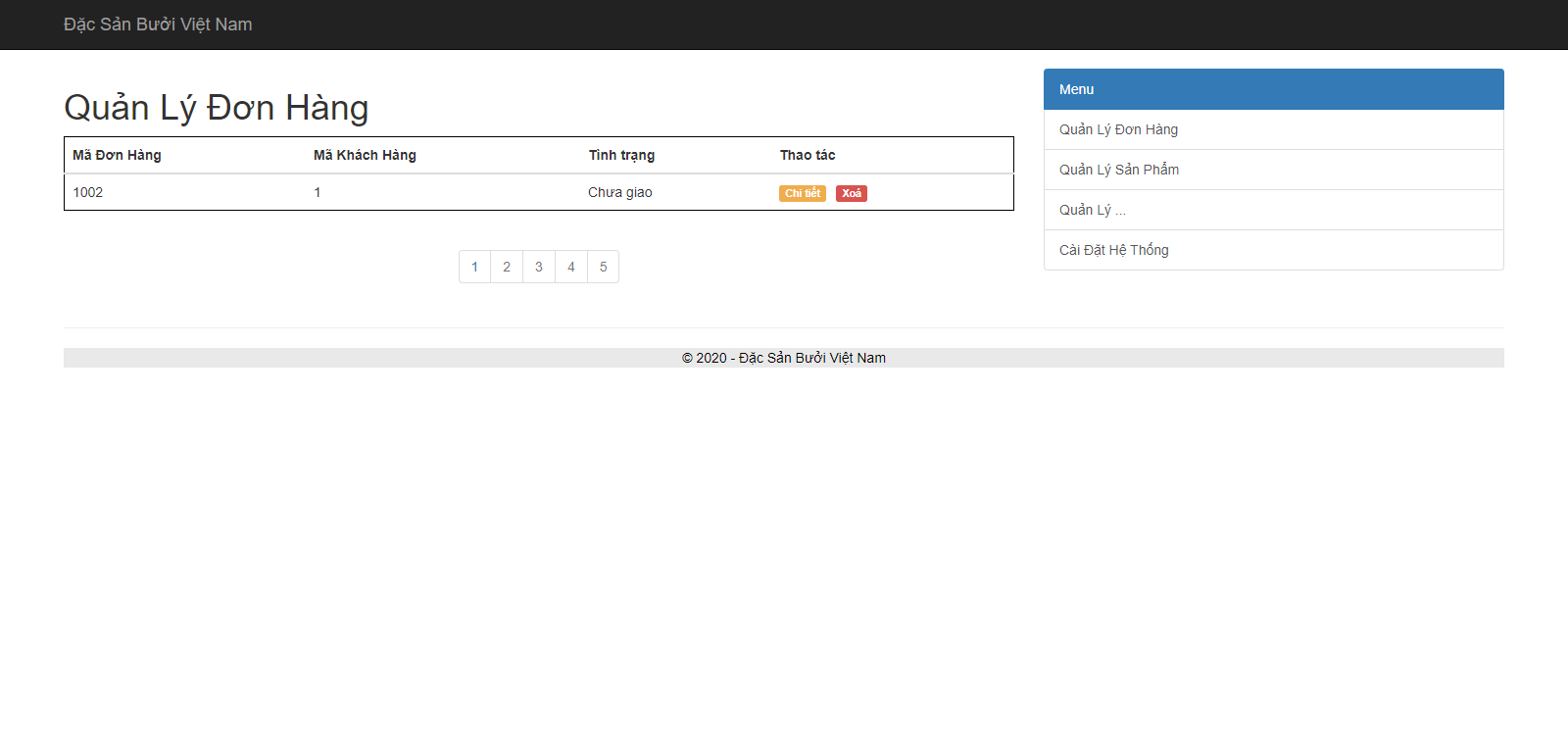
1. Biểu đồ hoạt động Usecase xem chi tiết sản phẩm

# GIAO DIỆN CHƯƠNG TRÌNH

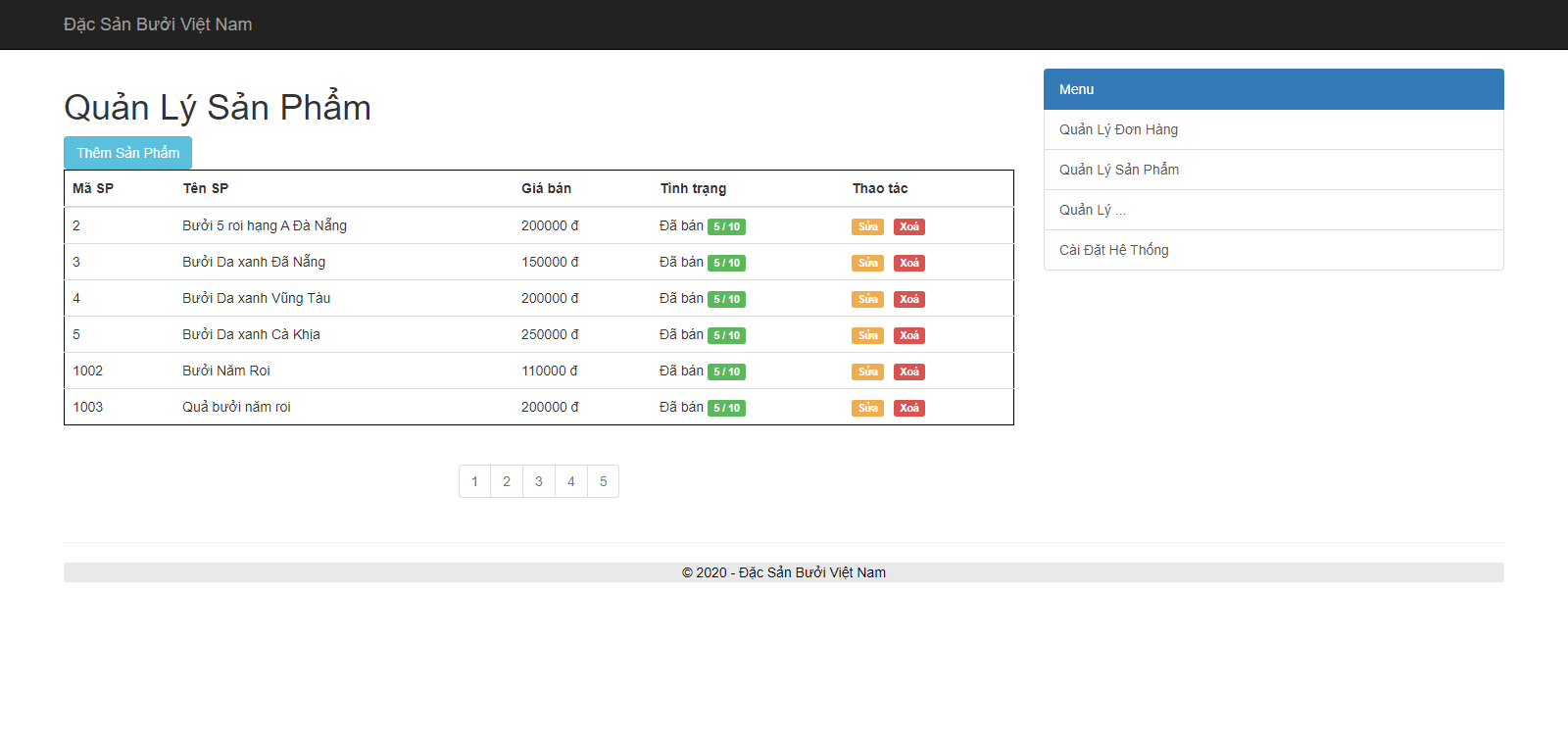
* 1. Giao diện trang quản lý
     1. Trang Index



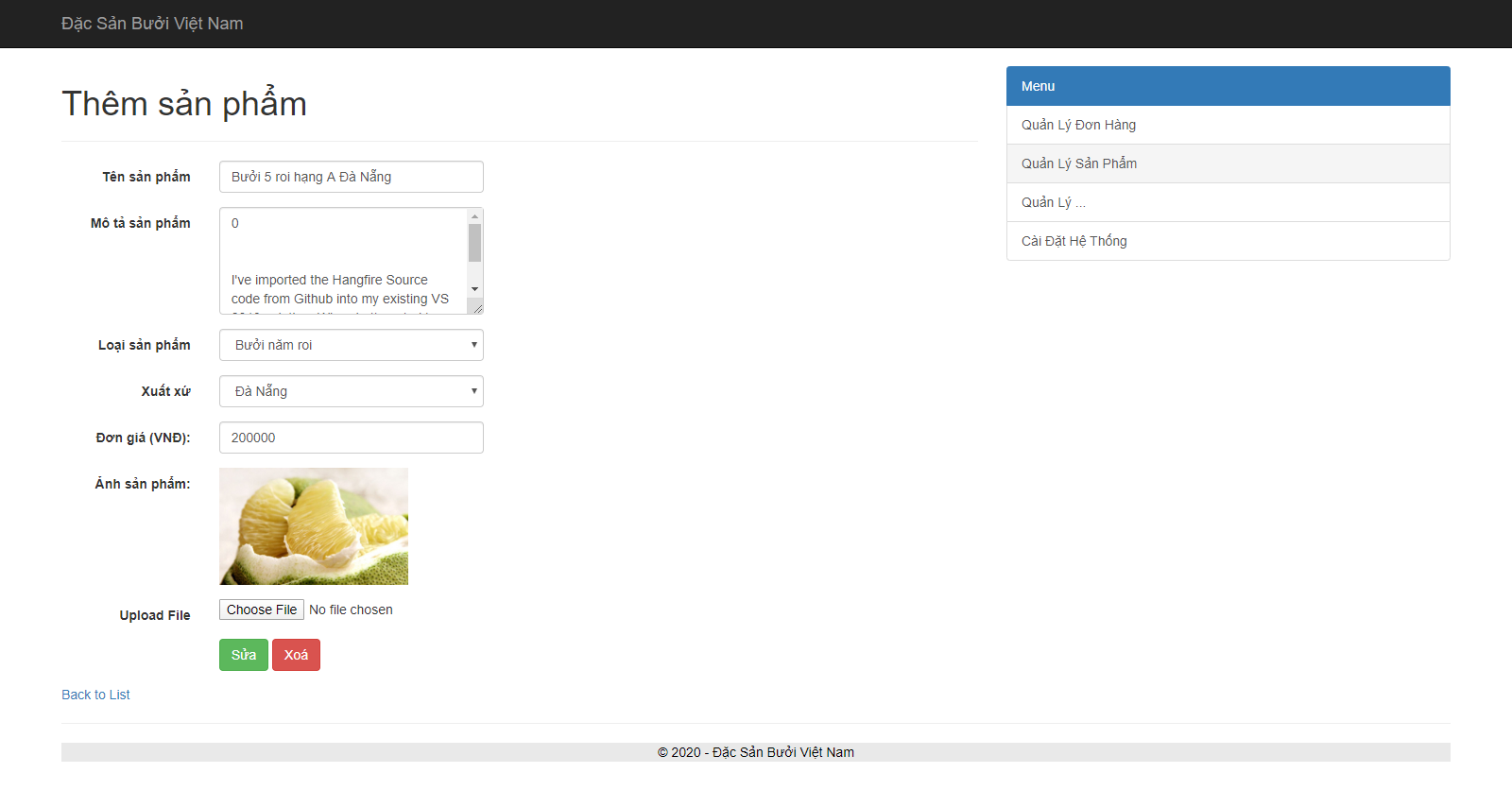
1. Giao diện trang Index quản lý
   * 1. Trang quản lý đơn hàng



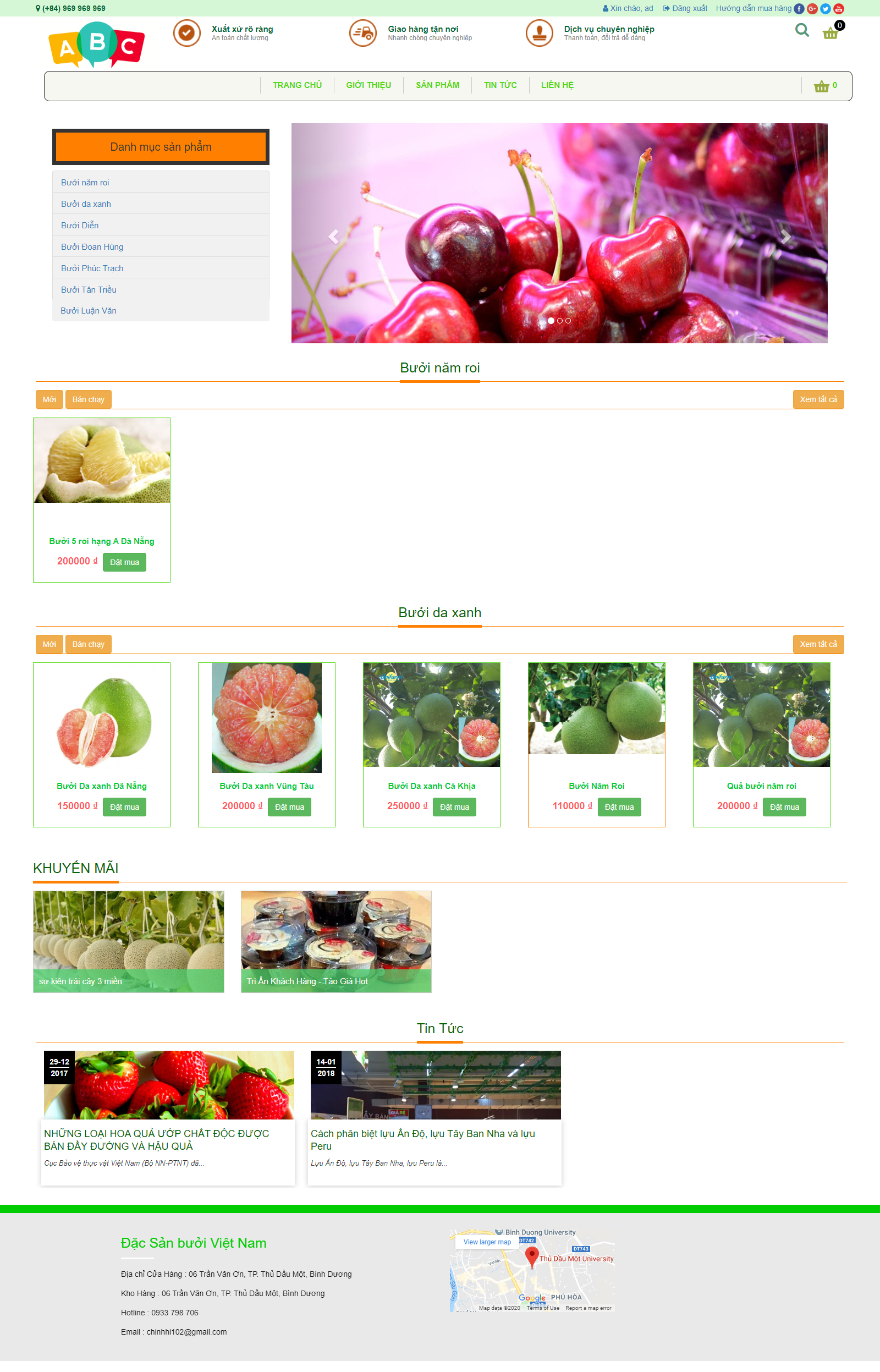
1. Giao diện trang quản lý đơn hàng
   * 1. Trang quản lý sản phẩm



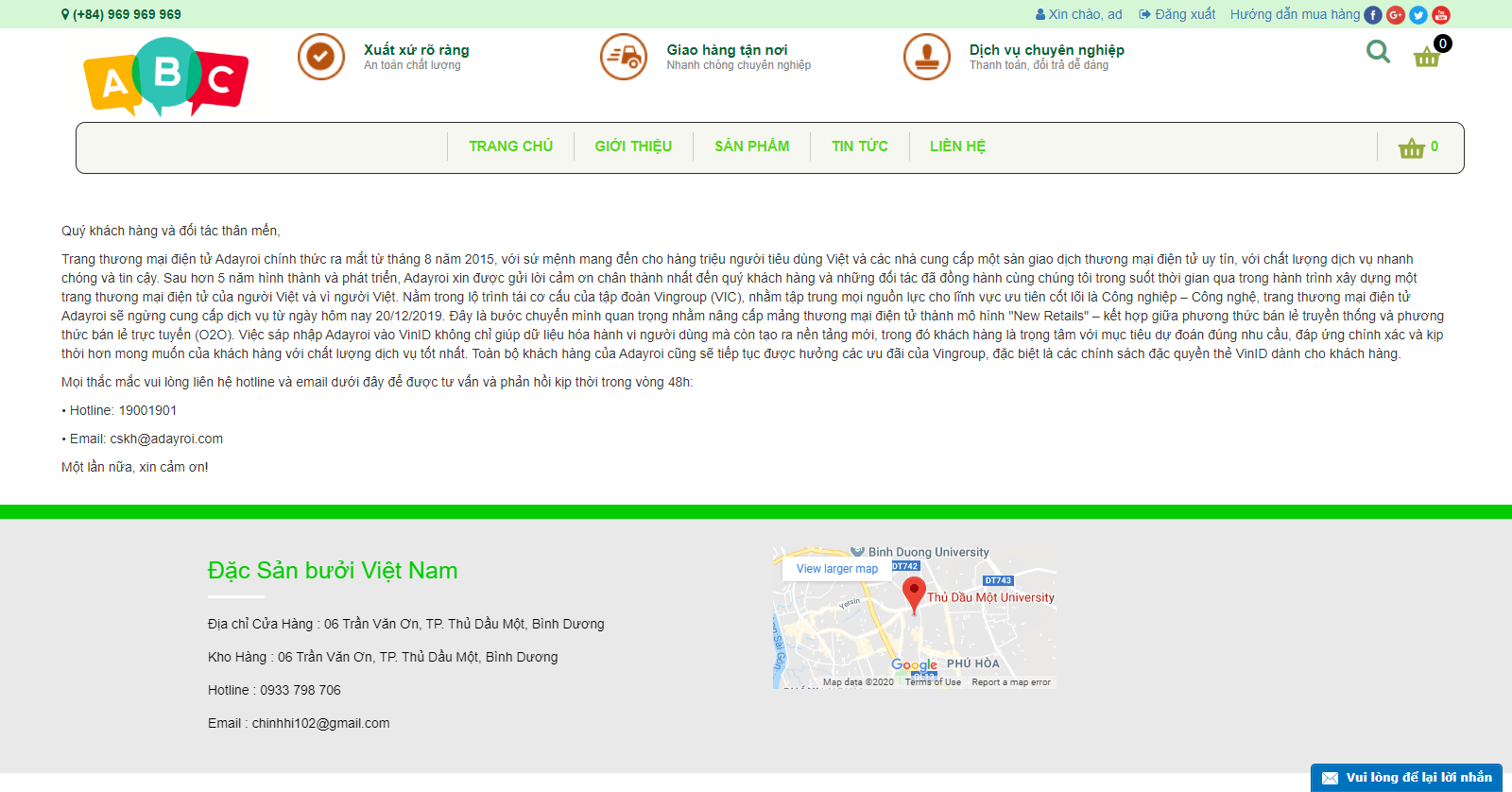
1. Giao diện trang quản lý sản phẩm
   * 1. Trang thêm sản phẩm



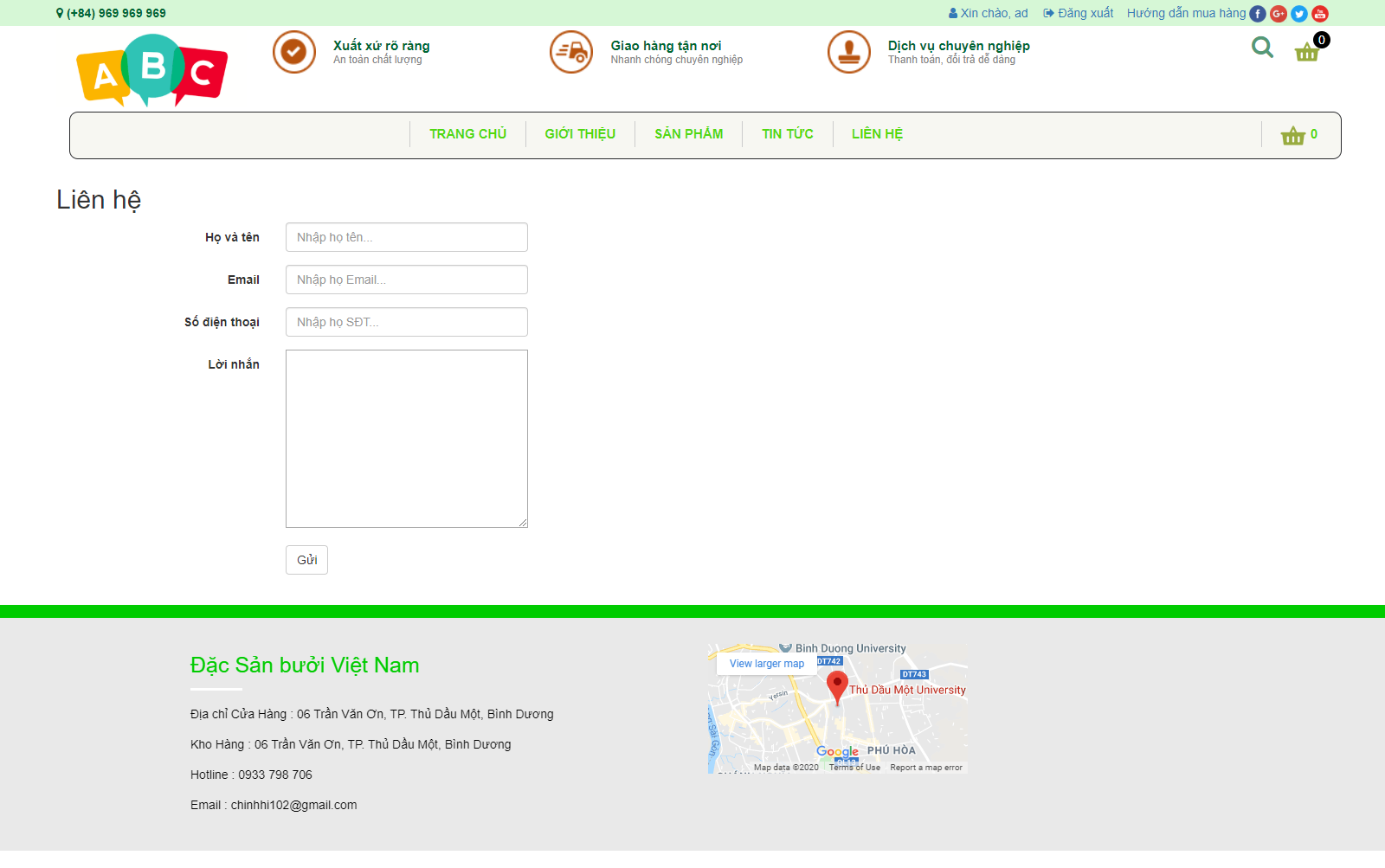
1. Giao diện thêm sản phẩm
   1. Giao diện trang người dùng
      1. Trang Index



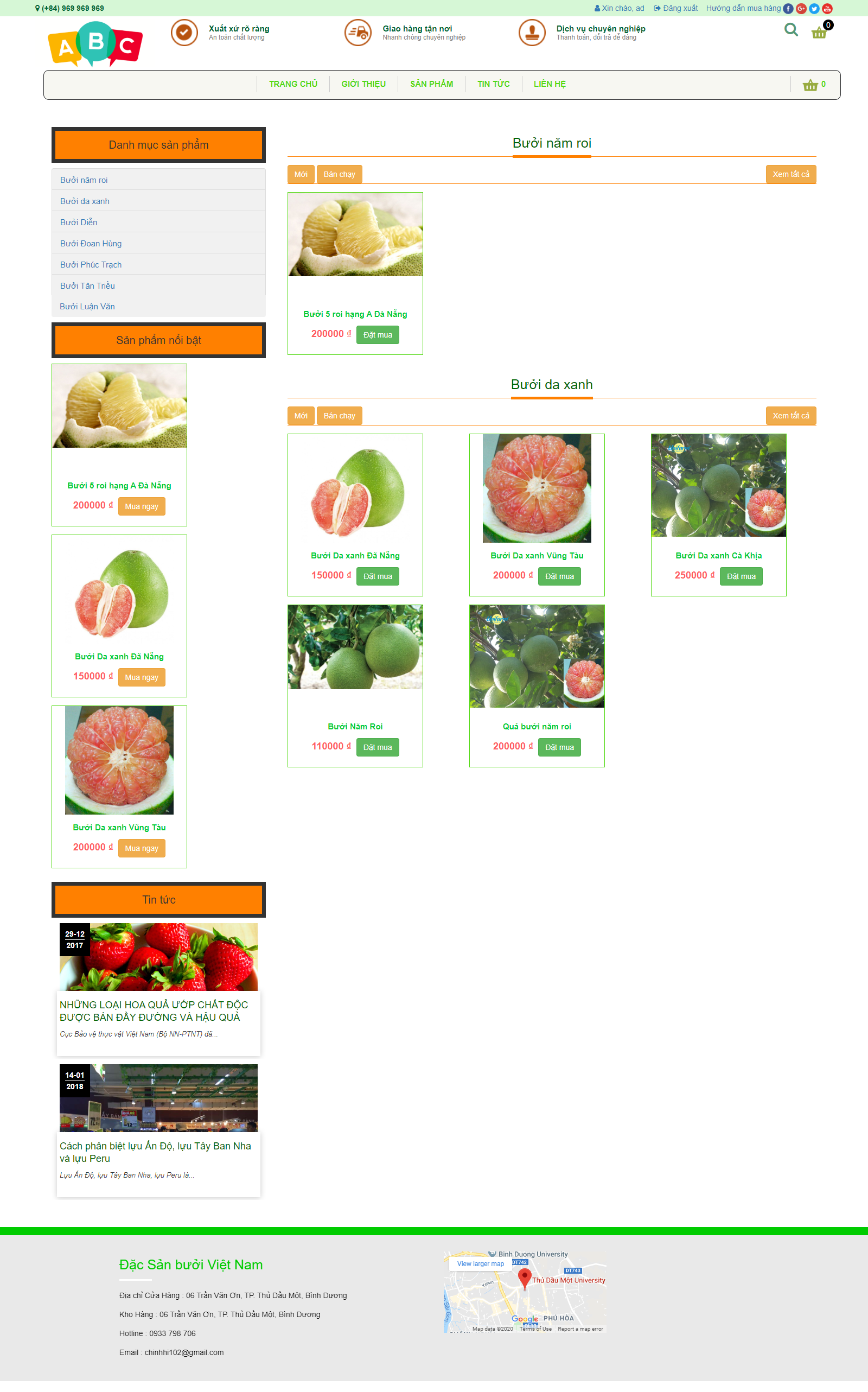
1. Giao diện trang Index người dùng
   * 1. Trang giới thiệu



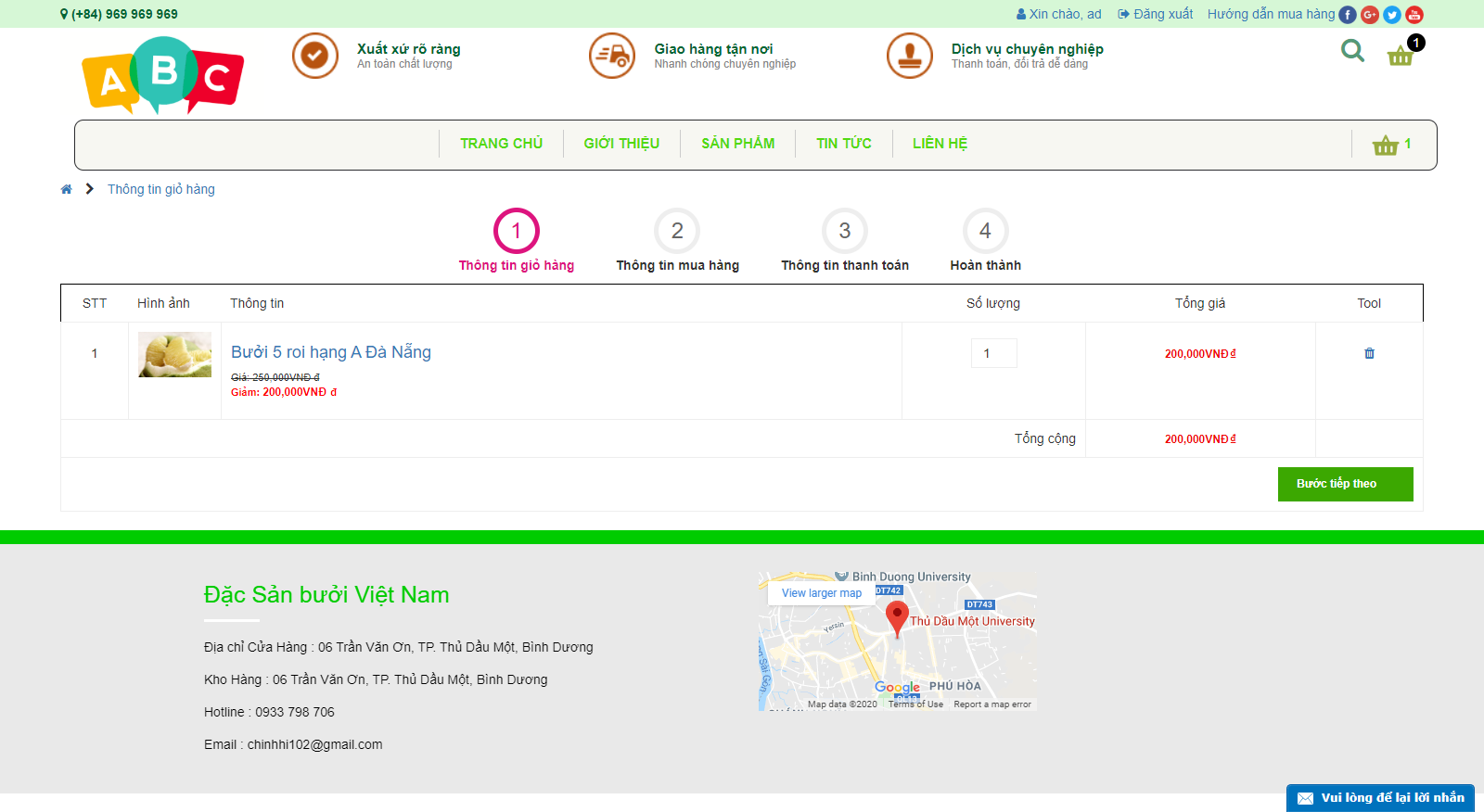
1. Giao diện trang giới thiệu
   * 1. Trang liên hệ



1. Giao diện trang liên hệ
   * 1. Trang sản phẩm



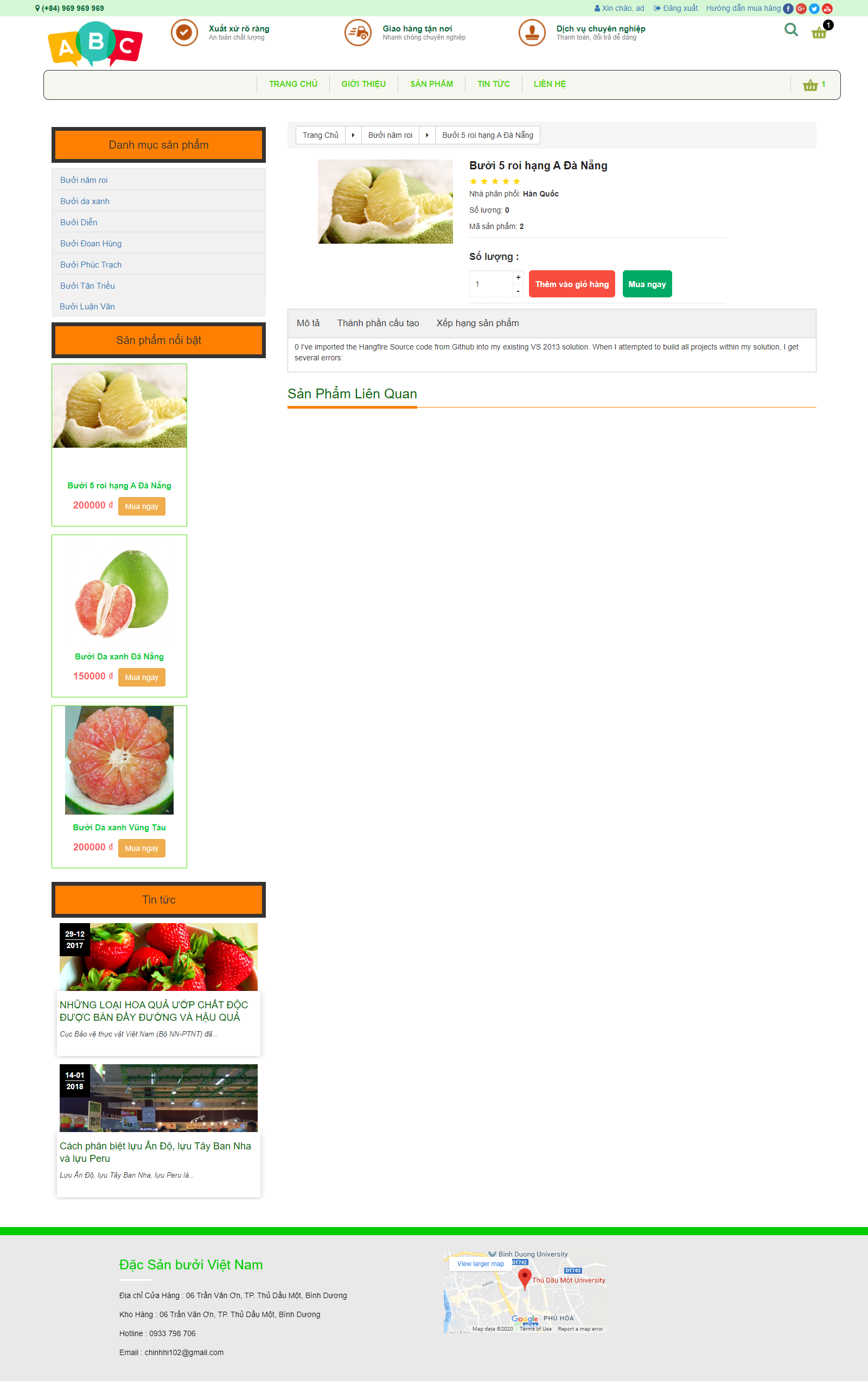
1. Giao diện trang sản phẩm
   * 1. Giao diện trang giỏ hàng



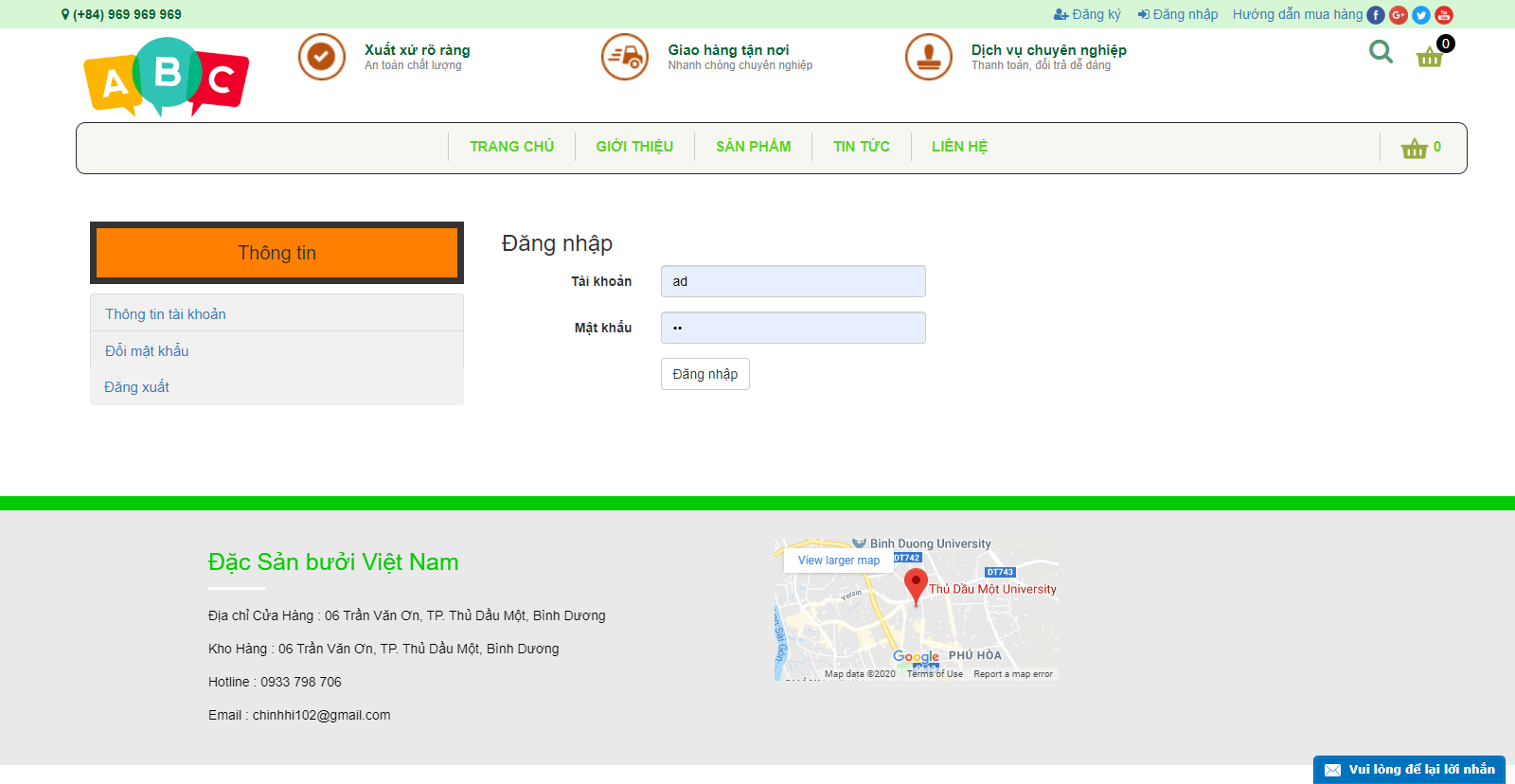
1. Giao diện trang giỏ hàng
   * 1. Giao diện trang tin tức



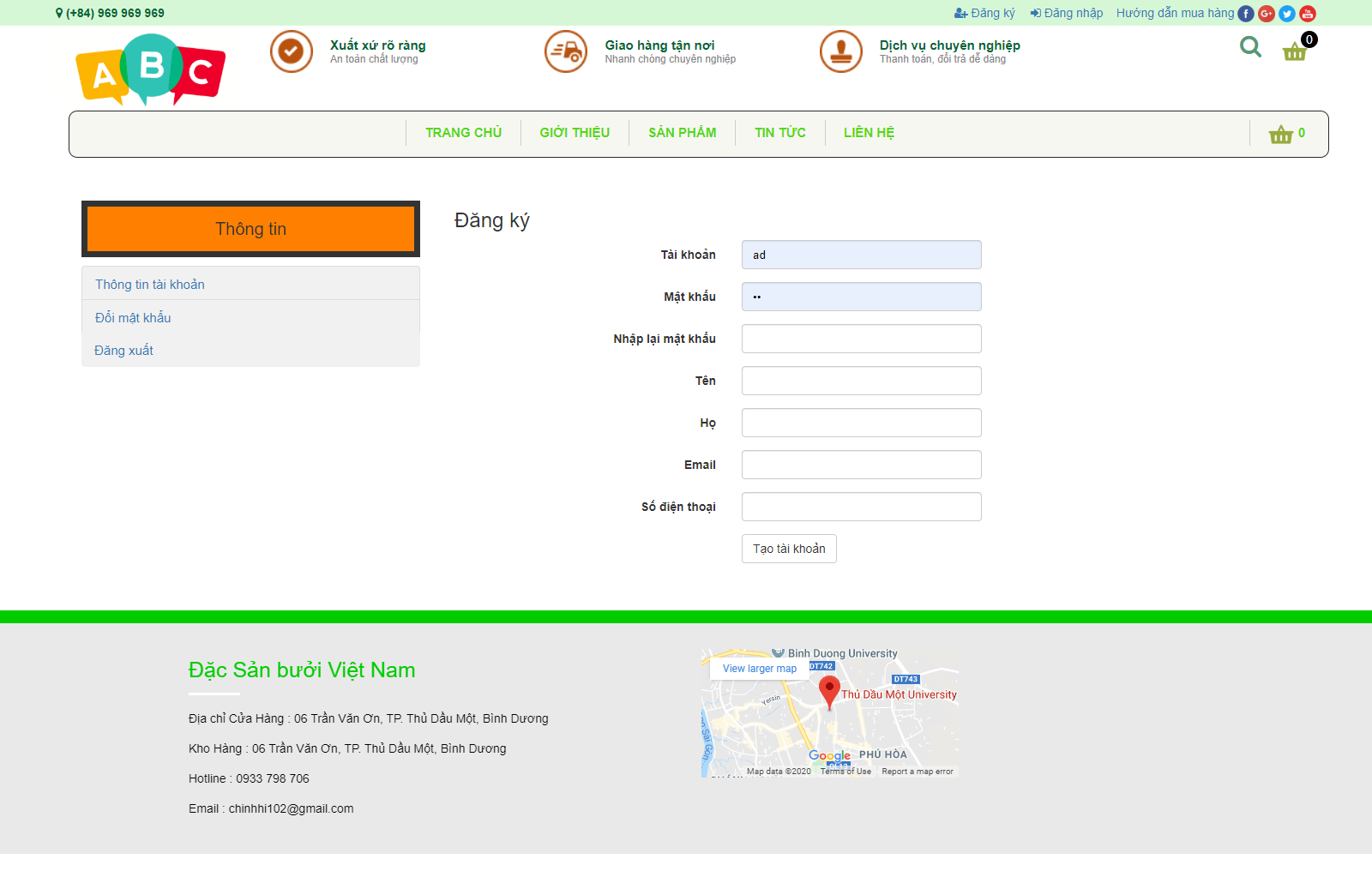
1. Giao diện trang tin tức
   * 1. Trang chi tiết sản phẩm



1. Giao diện trang chi tiết sản phẩm
   * 1. Trang đăng nhập



1. Giao diện trang đăng nhập
   * 1. Trang đăng ký



1. Giao diện trang đăng ký

# KẾT LUẬN

* 1. Ưu điểm của hệ thống
* Ứng dụng gọn nhẹ, dễ dàng sử dụng.
* Tra cứu thông tin và tìm hiểu sản phẩm một cách đơn giản và nhanh chóng.
* Có khả năng chứa nguồn dữ liệu lớn và truy xuất cực kì nhanh chóng.
* Ứng dụng có khả năng mở rộng để thêm một số tính năng khác.
  1. Nhược điểm của hệ thống
* Giao diện chưa thân thiện với người sử dụng
* Cơ sở dữ liệu sẵn có chưa mở rộng đủ để phát triển đề tài.
* Chưa có khả năng tích hợp các cơ sở dữ liệu bên ngoài
  1. Hướng phát triển của hệ thống
* Khắc phục các nhược điểm trên.
* Hoàn thành một số tính năng mới

TÀI LIỆU THAM KHẢO

## Tài liệu thư viện

1. Matthew MacDonald, Adam Freeman, and Mario Szpuszta, Pro ASP.NET 4 in C# 2010, Fourth Edition, Apress.
2. Nguyễn Hữu Vĩnh, Võ Quốc Lương (2017), Bài tập thực hành Thiết kế và Lập trình web, Đại học Thủ Dầu Một.
3. Nguyễn Văn Lân (2008), Kỹ thuật xây dựng ứng dụng ASP.net, Lao động xã hội.
4. Belkheraz, Stephane; Polat, Engin (2018), ASP.NET Core MVC 2.0 cookbook : effective ways to build modern, interactive web applications with ASP.NET Core MVC 2.0, PACKT Publishing.

## Tài liệu Internet

1. <https://lmt.com.vn/lap-trinh/tim-hieu-php-mysql/360-tim-hieu-mo-hinh-mvc-la-gi.html>
2. <https://webso.vn/web-tinh-va-web-dong-la-gi>
3. <https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng>
4. <http://ictvietnam.vn/danh-gia-va-trao-doi/doi-moi-sang-tao/co-so-du-lieu-quan-he-la-gi.htm>
5. <http://sqladvice.com/tong-quan-ve-sql-server/>
6. <https://voer.edu.vn/m/tong-quan-ve-aspnet/55e40ed5>
7. <https://voer.edu.vn/m/tong-quan-ve-aspnet/55e40ed5>