|  |  |
| --- | --- |
|  | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
| LÊ THÀNH CÔNG | ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC TT&MMT |
| **THIẾT KẾ WEBSITE BÁN SÁCH TRỰC TUYẾN** |
|  | **CBHD: TỐNG VĂN LUYÊN** |
|  | **Sinh viên: LÊ THÀNH CÔNG** |
| TT&MMT1 | **Mã số sinh viên: 1141150068** |
| Hà Nội – 2021 |

# MỤC LỤC

[MỤC LỤC 1](#_bookmark0)

[DANH MỤC BẢNG BIỂU 3](#_bookmark1)

[DANH MỤC HÌNH ẢNH 4](#_bookmark2)

[DANH MỤC TỪ VIẾT TẮT 6](#_bookmark3)

[LỜI NÓI ĐẦU 7](#_bookmark4)

[CHƯƠNG 1. TÌM HIỂU NGÔN NGỮ PHP & CƠ SỞ DỮ LIỆU MYSQL 9](#_bookmark5)

* 1. [Ngôn ngữ PHP 9](#_bookmark6)
     1. [Khái niệm PHP 9](#_bookmark7)
     2. [Tại sao nên dùng PHP 10](#_bookmark8)
     3. [Hoạt động của PHP 10](#_bookmark9)
  2. [Cơ sở dữ liệu MySQL 11](#_bookmark10)
     1. [Giới thiệu cơ sở dữ liệu 11](#_bookmark11)
     2. [Mục đích sử dụng cơ sở dữ liệu 12](#_bookmark12)

[TÓM TẮT CHƯƠNG 1 13](#_bookmark13)

[CHƯƠNG 2. KHẢO SÁT CÁC YÊU CẦU CỦA BÀI TOÁN 14](#_bookmark14)

* 1. [Phân tích yêu cầu đề tài 14](#_bookmark15)
     1. [Tóm tắt hoạt động của hệ thống mà dự án sẽ được ứng dụng 14](#_bookmark16)
     2. [Phạm vi dự án được ứng dụng 16](#_bookmark17)
     3. [Đối tượng sử dụng 16](#_bookmark18)
     4. [Mục đích của dự án 16](#_bookmark19)
  2. [Xác định yêu cầu của khách hàng 17](#_bookmark20)
     1. [Hệ thống hiện hành của cửa hàng 17](#_bookmark21)
     2. [Hệ thống đề nghị 17](#_bookmark22)

[TÓM TẮT CHƯƠNG 2 21](#_bookmark23)

[CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 22](#_bookmark24)

* 1. [Giao diện người dùng 22](#_bookmark25)
  2. [Giao diện người quản trị 22](#_bookmark26)
  3. [Phân tích các chức năng của hệ thống 22](#_bookmark27)
     1. [Các chức năng của đối tượng Customer (Khách vãng lai) 24](#_bookmark29)
     2. [Các chức năng của đối tượng Member (thành viên) 26](#_bookmark30)
     3. [Chức năng của Administrator 27](#_bookmark31)
  4. [Biểu đồ phân tích thiết kế hệ thống 30](#_bookmark32)
     1. [Biểu đồ hoạt động 31](#_bookmark34)
     2. [Đặc tả Use-case 43](#_bookmark47)
  5. [Cách tổ chức dữ liệu và thiết kế chương trình cho trang Web 47](#_bookmark55)
  6. [Yêu cầu về bảo mật 55](#_bookmark71)

[TÓM TẮT CHƯƠNG 3 56](#_bookmark72)

[CHƯƠNG 4. THỰC NGHIỆM CHƯƠNG TRÌNH 57](#_bookmark73)

* 1. [Giao diện trang chủ 57](#_bookmark74)
  2. [Giao diện Form đăng nhập 59](#_bookmark76)
  3. [Giao diện trang giỏ hàng 59](#_bookmark78)
  4. [Giao diện Form đăng ký thành viên 60](#_bookmark80)
  5. [Giao diện trang chi tiết sản phẩm 61](#_bookmark82)
  6. [Giao diện trang quản trị Admin 61](#_bookmark84)
  7. [Giao diện trang đặt hàng 62](#_bookmark86)
  8. [Giao diện trang đặt hàng thành công 62](#_bookmark88)
  9. [Hệ thống sẽ gửi mail thông báo đơn hàng về cho khách hàng 63](#_bookmark90)
  10. [Hỗ trợ online trực tuyến trên Website 63](#_bookmark92)

[TÓM TẮT CHƯƠNG 4 64](#_bookmark94)

[KẾT LUẬN 65](#_bookmark95)

[TÀI LIỆU THAM KHẢO 66](#_bookmark96)

[PHỤ LỤC 67](#_bookmark97)

# DANH MỤC BẢNG BIỂU

[Bảng 1: Bảng db\_category 47](#_bookmark56)

[Bảng 2: Bảng db\_config (Cấu hình) 48](#_bookmark57)

[Bảng 3: Bảng db\_contact 48](#_bookmark58)

[Bảng 4: db\_content 49](#_bookmark59)

[Bảng 5: db\_customer 49](#_bookmark60)

[Bảng 6: db\_ discount 50](#_bookmark61)

[Bảng 7: db\_district 50](#_bookmark62)

[Bảng 8: db\_ order 51](#_bookmark63)

[Bảng 9: db\_ orderdetail 51](#_bookmark64)

[Bảng 10: db\_producer 52](#_bookmark65)

[Bảng 11: db\_product 52](#_bookmark66)

[Bảng 12: db\_province 53](#_bookmark67)

[Bảng 13: db\_slider 53](#_bookmark68)

[Bảng 14: db\_user 54](#_bookmark69)

[Bảng 15: db\_usergroup 54](#_bookmark70)

# DANH MỤC HÌNH ẢNH

[Hình 3.1: Biểu đồ Use Case 23](#_bookmark28)

[Hình 3.2: Sơ đồ quan hệ 30](#_bookmark33)

[Hình 3.3: Biểu đồ hoạt động đăng nhập vào hệ thống 31](#_bookmark35)

[Hình 3.4: Biểu đổ hoạt động thoát ra khỏi hệ thống 32](#_bookmark36)

[Hình 3.5: Biểu đồ hoạt động đăng kí thành viên 33](#_bookmark37)

[Hình 3.6: Biểu đồ hoạt động xem thông tin về sản phẩm 34](#_bookmark38)

[Hình 3.7: Biểu đồ chức năng thêm SP vào giỏ hàng 35](#_bookmark39)

[Hình 3.8: Biểu đồ chức năng đặt hàng 36](#_bookmark40)

[Hình 3.9: Biểu đồ chức năng đổi mật khẩu 37](#_bookmark41)

[Hình 3.10: Biểu đồ chức năng xoá Member 38](#_bookmark42)

[Hình 3.11: Biểu đồ chức năng thêm sản phẩm 39](#_bookmark43)

[Hình 3.12: Biểu đồ chức năng sửa thông tin sản phẩm 40](#_bookmark44)

[Hình 3.13: Biểu đồ chức năng xoá sản phẩm 41](#_bookmark45)

[Hình 3.14: Biểu đồ chức năng xoá User 42](#_bookmark46)

[Hình 3.15: Use-case quản lý sản phẩm 43](#_bookmark48)

[Hình 3.16: Biểu đồ cộng tác chức năng cập nhật 43](#_bookmark49)

[Hình 3.17: Biểu đồ tuần tự chức năng thêm sản phẩm mới 44](#_bookmark50)

[Hình 3.18: Biểu đồ tuần tự chức năng sửa thông tin sản phẩm 44](#_bookmark51)

[Hình 3.19: Biểu đồ tuần tự chức năng xóa sản phẩm 45](#_bookmark52)

[Hình 3.20: Biểu đồ tuần tự chức năng đăng nhập 46](#_bookmark53)

[Hình 3.21: Biểu đồ tuần tự chức năng đăng ký thành viên 46](#_bookmark54)

[Hình 4.1: Giao diện trang chủ 57](#_bookmark75)

[Hình 4.2: Giao diện form đăng nhập 59](#_bookmark77)

[Hình 4.3: Giao diện giỏ hàng 59](#_bookmark79)

[Hình 4.4: Giao diện form đăng ký thành viên 60](#_bookmark81)

[Hình 4.5: Giao diện trang chi tiết sản phẩm 61](#_bookmark83)

[Hình 4.6: Giao diện trang quản trị Admin 61](#_bookmark85)

[Hình 4.7: Giao diện trang đặt hàng 62](#_bookmark87)

[Hình 4.8: Giao diện trang đặt hàng thành công 62](#_bookmark89)

[Hình 4.9: Kết quả trả về 63](#_bookmark91)

[Hình 4.10: Hỗ trợ online trực tuyến 63](#_bookmark93)

# DANH MỤC TỪ VIẾT TẮT

|  |  |
| --- | --- |
| Từ viết tắt | Ý nghĩa |
| CSDL | Cơ sở dữ liệu |
| SP | Sản phẩm |

# LỜI NÓI ĐẦU

Với sự phát triển nhảy vọt của công nghệ thông tin hiện nay, Internet ngày càng giữ vai trò quan trọng trong các lĩnh vực khoa học kĩ thuật và đời sống. Dĩ nhiên các bạn đã được nghe nói nhiều về Internet, nói một cách đơn giản, Internet là một tập hợp máy tính nối kết với nhau, là một mạng máy tính toàn cầu mà bất kì ai cũng có thể kết nối bằng máy PC của họ. Với mạng Internet, tin học thật sự tạo nên một cuộc cách mạng trao đổi thông tin trong mọi lĩnh vực văn hóa, xã hội, chính trị, kinh tế...

Trong thời đại ngày nay, thời đại mà “người người làm Web, nhà nhà làm Web” thì việc có một Website để quảng bá công ty hay một Website cá nhân không còn là điều gì xa xỉ nữa. Thông qua Website khách hàng có thể lựa chọn những sản phẩm mà mình cần một cách nhanh chóng và hiệu quả.

Với công nghệ Word Wide Web, hay còn gọi là Web sẽ giúp bạn đưa những thông tin mong muốn của mình lên mạng Internet cho mọi người cùng xem một cách dễ dàng với các công cụ và những ngôn ngữ lập trình khác nhau. Sự ra đời của các ngôn ngữ lập trình cho phép chúng ta xây dựng các trang Web đáp ứng được các yêu cầu của người sử dụng. PHP (Personal Home Page) là kịch bản trên phía trình chủ (Server Script) cho phép chúng ta xây dựng trang Web trên cơ sở dữ liệu. Với nhiều ưu điểm nổi bật mà PHP và MySQL được rất nhiều người sử dụng.

Với lí do đó, được sự hướng dẫn và giúp đỡ của thầy **Tống Văn Luyên**, em đã chọn: “**Thiết kế website bán sách trực tuyến**” làm đề tài cho đồ án đề tài tốt nghiệp của mình. Cụ thể em sẽ xây dựng website bán sách trực tuyến.

Tuy nhiên, do trình độ có hạn, thời gian nghiên cứu ngắn, đề tài đồ án khá phức tạp, đồ án em viết không tránh khỏi những thiếu sót và hạn chế. Em rất mong nhận được sự đóng góp ý kiến bổ sung của giảng viên hướng dẫn để nội dung đồ án được hoàn thiện hơn, chặt chẽ hơn về cả lý luận và tính thực tiễn.

Em xin chân thành cảm ơn thầy **Tống Văn Luyên** đã tận tình chỉ bảo cho em

### Em xin chân thành cảm ơn!

|  |
| --- |
| **Sinh viên thực hiện** |
| **Lê Thành Công** |

# CHƯƠNG 1. TỔNG QUAN VỀ WEBSITE, QUY TRÌNH VÀ CÔNG CỤ XÂY DỰNG WEBSITE

## Tổng quan về Website

### Website là gì

Website là một tập hợp các trang web (web pages) bao gồm văn bản, hình ảnh, video, flash v.v... thường chỉ nằm trong một tên miền (domain name) hoặc tên miền phụ (subdomain) trên World Wide Web của Internet. Trang web được lưu trữ (web hosting) trên máy chủ web (web server) có thể truy cập thông qua Internet. Một trang web là tập tin HTML hoặc XHTML có thể truy nhập dùng giao thức HTTP hoặc HTTPS. Website có thể được xây dựng từ các tệp tin HTML (website tĩnh) hoặc vận hành bằng các CMS chạy trên máy chủ (website động). Website có thể được xây dựng bằng nhiều ngôn ngữ lập trình khác nhau (PHP,.NET, Java, Ruby on Rails...)

### Website tĩnh

Web tĩnh là những website không có hệ thống quản lý nội dung (xem bài cms) và người dùng không thể chỉnh sửa được hoặc có thể chỉnh sửa, thay đổi dữ liệu được. Hiện nay web tĩnh hầu như ít tồn tại hoặc chỉ tồn tại với những công ty chuyên về thiết kế website, họ sử dụng web tĩnh bởi họ có kiến thức và dễ dàng chỉnh sửa nội dung khi cần.

**Web tĩnh được làm từ gì?**

Về kiến thức cơ bản thì web tĩnh thường được xây dựng từ CSS, HTML, JAVASCRIPT (DHTML), hiện nay có thêm công nghệ HTML5 & CSS3. Có thể bạn nhìn một web nào đó nhìn rất lung lay, đẹp và nội dung có thể thay đổi nhưng chưa chắc nó là web động bởi vì người ta có thể sử dụng DHTML để thay đổi nội dung (tại client). Nếu bạn chạy trên máy tính của bạn thì với web tĩnh bạn có thể đặt tại đâu cũng chạy được vì bản chất nó chỉ là một file bình thường.

**Ưu điểm của web tĩnh:**

Về giao diện Designer có thẻ thiết kế theo kiểu mới lạ Tốc độ truy cập nhanh bởi nó chỉ là những file HTML Chi phí đầu tư thấp bởi bạn không phải trả tiền nhiều cho Coder Thân thiện với bộ máy tìm kiếm bởi bạn có thể đặt tên file tùy ý (ten-file.html, tieu-de-tin-tuc.html)

**Nhược điểm của web tĩnh:**

Khó quản lý nội dung Khó nâng cấp bảo trì Mỗi khi thay đổi phải vào file HTML, CSS hoặc JAVASCRIPT để chỉnh sửa

**Khi nào nên sử dụng web tĩnh?**

Nếu bạn là doanh nghiệp muốn tự mình làm website thì bạn có thể học các kiến thức căn bản và tự làm một Web tĩnh cho mình Nội dung website ít khi cập nhật và ít nên bạn muốn tiết kiệm chi phí Website bạn nhỏ và bạn thuê luôn người chuyên về web để quản trị

### Website động

Ngược lại với web tĩnh, web động là những website có hệ thống quản lý nội dung và người dùng có thể chỉnh sửa nội dung được. Ví dụ như trang vnexpress.net hay 24h.com.vn thì đó là những web động bởi vì họ có thể đăng tin mới, chỉnh sửa tin mới và chỉnh sửa danh mục menu, ...

**Web động được làm từ gì?**

Web động sử dụng các công nghệ như HTML, CSS, HTML5, CSS3, JAVASCRIPT, ... và điều đặc biệt là có sử dụng một ngôn ngữ lập trình server như Java, một hệ quản trị cơ sở dữ liệu như MySQL

**Ưu điểm của web động:**

Dễ dàng quản lý nội dung Dễ dàng nâng cấp và bảo trì Có thể xây dụng được web lớn Thường sử dụng tương tác với người dùng cao Hiện nay vẫn SEO tốt bởi ta có thể sử dụng chức năng Rewrite URL để chuyên URL thân thiện

**Nhược điểm của web động:**

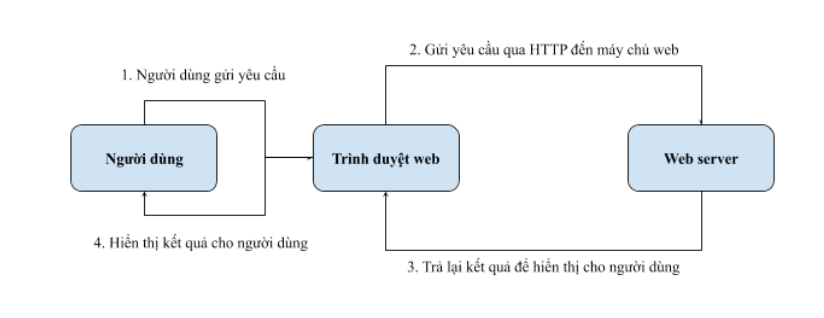
Chi phí xây dựng cao Nếu web lớn có thể cần thêm nhân sự chuyên ngành

**Khi nào nên sử dụng web động?**

Bạn làm website thương mại điện tử bán hàng Bạn làm web giới thiệu sản phảm công ty Bạn làm web tin tức, blog cá nhân Web bạn tầm cỡ lớn Nhưng chung quy lại thì hiện nay đa số bạn nên chọn giải pháp làm web động sẽ tốt hơn.

### Cách thức hoạt động của trang web động

Web động sẽ gồm có ba thành phần chính: Web server; Trình duyệt web và Giao thức HTTP. Ở phần tiếp theo này chúng ta sẽ làm rõ hơn cách thức vận hành của cả ba thành phần này nhé.



**Bước 1:**

Người dùng gửi yêu cầu Người dùng Internet sẽ truy cập một website bất kỳ thông qua một trình duyêt web được cài trên máy tính hoặc thiết bị di động.

**Bước 2:**

Trình duyệt web gửi yêu cầu tới Web server để xử lý Lúc này, trình duyệt web mà bạn đang sử dụng (Chrome, Cốc Cốc, Firefox …) sẽ nhận yêu cầu đó và chuyển đổi từ địa chỉ tên miền sang địa chỉ IP kèm theo tên miền đó. Việc truy xuất thông tin IP này sẽ thông qua các máy chủ DNS. Sau đó trình duyệt sẽ thông qua giao thức HTTP gửi yêu cầu đến Web server báo là có một người dùng đang cần truy xuất thông tin tại địa chỉ này. Và nó yêu cầu máy chủ hãy trả về kết quả cho người dùng.

**Bước 3:**

Máy chủ web kiểm tra, trả về kết quả và trình duyệt hiển thị kết quả cho người dùng Khi nhận được yêu cầu từ trình duyệt, máy chủ web sẽ kiểm tra lại trong hệ thống xem có tài nguyên nào liên quan đến địa chỉ mà người dùng đang cần tìm hay không. Trường hợp có nó sẽ trả lại thông tin qua giao thức HTTP đến trình duyệt web để hiển thị cho người dùng. Còn nếu không thì nó sẽ xuất hiện các thông báo lỗi hoặc nội dung không tìm thấy. Cứ như vậy quy trình này được lặp đi lặp lại.

## Quy trình xây dựng Website

### Thu thập, khảo sát thông tin thiết kế Website

Điều đầu tiên cần làm trong mọi quy trình thiết kế website đó là thu thập thông tin đầu vào. Để không chệch hướng khi thiết kế, mọi thông tin càng chi tiết, cụ thể thì càng tốt. Theo đó, bạn cần hiểu rõ về những yếu tố: mục tiêu của website, khách hàng mục tiêu của website, nội dung cốt lõi mà website muốn hướng tới. Khi bạn thiết kế website cho khách hàng, việc ghi nhận các yêu cầu về tính năng, hình thức, tên miền – máy chủ lưu trữ của website cũng rất quan trọng; cùng với đó là các tư liệu, thông tin, hình ảnh phục vụ cho việc thiết kế mà khách hàng cung cấp

### Phân tích các yêu cầu của Website

* Xác định các tính năng của website:

Đây là yêu cầu cơ bản khi thiết kế website tiếp theo. Bạn hãy liệt kê các nội dung mà bạn muốn đưa lên website như: thông tin giới thiệu về công ty, thông tin - hình ảnh dịch vụ, sản phẩm, thông tin liên hệ, các hình ảnh hoạt động của công ty,... Điều này sẽ giúp đơn vị tư vấn xây dựng website dễ dàng thống nhất yêu cầu xây dựng website với bạn hơn.

* Xác định các yêu cầu về mặt giao diện website:

Đây là phần quan trọng trong yêu cầu để xây dựng 1 website. Bạn là người hiểu rõ nhất sở thích của mình, các yêu cầu của khách hàng mục tiêu. Bạn có thể định hình trước các tiêu chí như: màu sắc đặc trưng của website là màu gì? bố cục tương tự như website www.? kiểu bố cục mà bạn thích là kiểu bố cục ngắn hay dài? bạn thích phong cách thiết kế đơn giản hay nhiều đường nét?... Khi bạn trả lời các câu hỏi này, bạn sẽ định hình được một phần yêu cầu của website. Chi phí mà bạn dự định dành cho việc thiết kế/xây dựng website - yêu cầu của 1 website quyết định mức chi phí xây dựng website.

### Thiết kế Website

Sau khi đã có một kế hoạch hoàn thiện, bạn có thể tiến đến giai đoạn hiện thực hóa trang web bằng cách thiết kế giao diện. Khi thiết kế trang web, bạn nên cân đối nhiều yếu tố để cho ra những lựa chọn phù hợp nhất. Dựa trên lĩnh vực hoạt động kinh doanh, thì một website ở lĩnh vực tài chính không thể trông giống như một website giải trí. Hay dựa theo những yêu cầu về ý nghĩa mà website cần thể hiện từ khách hàng, dựa theo bộ nhận diện thương hiệu của công ty để đồng bộ theo. Và dù bạn lên kế hoạch như thế nào thì cũng đừng quên rằng trải nghiệm của người dùng mới là ưu tiên hàng đầu – trang web nên đủ thuận tiện, tối giản và dễ hiểu cho người dùng truy cập và sử dụng.

### Triển khai Website

Deploy/ deployment được hiểu là triển khai, sắp đặt một thứ gì đó. Trong lĩnh vực phần mềm, deploy được hiểu là triển khai tiến hành sử dụng phần mềm hoàn thiện trong môi trường ứng dụng thực tế. Nó có nghĩa gần giống với install, thế nhưng deploy mang một sắc thái ý nghĩa riêng đó là có thể biến những component được tham chiếu từ phần mềm trên máy chủ hoặc được sử dụng thông qua internet từ bên ngoài hay những component được tham chiếu từ các phần mềm khác thành trạng thái có thể sử dụng và tiếp cận được.

## Các công cụ để xây dựng Website

### Ngôn ngữ lập trình hệ thống

#### Giới thiệu về ngôn ngữ lập trình Java

Java là một trong những ngôn ngữ lập trình hướng đối tượng. Nó được sử dụng trong phát triển phần mềm, trang web, game hay ứng dụng trên các thiết bị di động. Java được khởi đầu bởi James Gosling và bạn đồng nghiệp ở Sun MicroSystem năm 1991. Ban đầu Java được tạo ra nhằm mục đích viết phần mềm cho các sản phẩm gia dụng, và có tên là Oak.

Java được phát hành năm 1994, đến năm 2010 được Oracle mua lại từ Sun MicroSystem. Java được tạo ra với tiêu chí “Viết (code) một lần, thực thi khắp nơi” (Write Once, Run Anywhere – WORA). Chương trình phần mềm viết bằng Java có thể chạy trên mọi nền tảng (platform) khác nhau thông qua một môi trường thực thi với điều kiện có môi trường thực thi thích hợp hỗ trợ nền tảng đó.

**Ứng dụng của Java:**

* Phát triển ứng dụng cho các thiết bị điện tử thông minh, các ứng dụng cho doanh nghiệp với quy mô lớn.
* Tạo các trang web có nội dung động (web applet), nâng cao chức năng của server.
* Phát triển nhiều loại ứng dụng khác nhau: Cơ sở dữ liệu, mạng, Internet, viễn thông, giải trí,...

**Những đặc điểm cơ bản của Java:**

* Đơn giản và quen thuộc: Vì Java kế thừa trực tiếp từ C/C++ nên nó có những đặc điểm của ngôn ngữ này, Java đơn giản vì mặc dù dựa trên cơ sở C++ nhưng Sun đã cẩn thận lược bỏ các tính năng khó nhất của của C++ để làm cho ngôn ngữ này dễ sử dụng hơn.
* Hướng đối tượng và quen thuộc.
* Mạnh mẽ (thể hiện ở cơ chế tự động thu gom rác - Garbage Collection) và an toàn.
* Kiến trúc trung lập, độc lập nền tảng và có tính khả chuyển (Portability).
* Hiệu suất cao.
* Đa nhiệm: Ngôn ngữ Java cho phép xâ dựng trình ứng dụng, trong đó nhiều quá trình có thể xảy ra đồng thời. Tính đa nhiệm cho phép các nhà lập trình có thể biên soạn phần mềm đáp ứng tốt hơn, tương tác tốt hơn và thực hiện theo thời gian thực.

#### Tại sao nên sử dụng Java

Trong hơn 24 năm hình thành và phát triển, Java đã trở thành nền tảng của hầu hết các ứng dụng ngày nay. Nó đã chứng minh được giá trị của mình trong việc phát triển các phần mềm doanh nghiệp, ứng dụng đi động, phát triển website, game và cả hệ thống nhúng. Java đã được sử dụng rất nhiều trong các ứng dụng ngân hàng / tài chính trên khắp thế giới. Nó cũng được các doanh nghiệp lớn như Amazon, Google và Facebook coi là một trong những ngôn ngữ tốt nhất để duy trì và phát triển website dành cho doanh nghiệp. Do đó, có thể nói rằng bạn hoàn toàn yên tâm chọn Java để Lập trình Web. Dưới đây là những lý do tại sao nên sử dụng Java để lập trình web:

* JAVA độc lập với nền tảng giúp giảm chi phí lập trình và phát triển web tối đa
* Khả năng bảo mật của Java rất cao
* Java có bộ API phong phú cho lập trình web
* JAVA được sử dụng rất nhiều trong các ứng dụng thực tế
* Java có các công cụ hỗ trợ lập trình cực kỳ tốt
* Java hỗ trợ lập trình đa luồng giúp tăng tối đa khả năng xử lý của các trang web. Biến chúng trở thành ứng dụng web tiên tiến.
* Trang web lập trình bằng Java dễ Scale
* Tính hướng đối tượng của Java là đầy đủ nhất, đáp ứng yêu cầu khắt khe nhất
* Java có một cộng đồng khổng lồ và tuyệt vời giúp bạn giải quyết mọi khó khăn trong học tập và lập trình web

#### Giới thiệu về HTML, CSS,JavaScript

**HTML là gì?**

HTML hay HyperText Markup Language – Ngôn ngữ đánh dấu siêu văn bản, là ngôn ngữ được sử dụng cho các tài liệu web. Nhưng HTML không phải là một ngôn ngữ lập trình, HTML giống như một ngôn ngữ xác định đâu là ý nghĩa, mục đích và cấu trúc của một tài liệu. Cùng với CSS và JavaScript, HTML tạo ra bộ ba nền tảng kỹ thuật cho các website.

**Vai trò của HTML**

Một website thường chứa nhiều trang con. Mỗi trang con sẽ có một tập tin HTML riêng. Dù bạn lập trình bằng bất cứ ngôn ngữ nào, trên bất cứ Framework nào, khi chạy trên nền website, chúng đều được biên dịch ra ngôn ngữ HTML. Đa số các trình soạn thảo văn bản trên website đều có 2 chế độ xem: HTML và văn bản thường. Chức năng chính của HTML là xây dựng cấu trúc siêu văn bản trên một website, hoặc khai báo các tập tin kỹ thuật số (media) như hình ảnh, video, nhạc. HTML thường được dùng để phân chia các đoạn văn, heading, links, blockquotes,… Với HTML, bạn sẽ có thể: Thêm tiêu đề, định dạng đoạn văn, ngắt dòng điều khiển. Tạo danh sách, nhấn mạnh văn bản, tạo ký tự đặc biệt, chèn hình ảnh, tạo liên kết. Xây dựng bảng, điều khiển một số kiểu mẫu Như đã đề cập ở trên, HTML không phải là ngôn ngữ lập trình. Do đó, HTML cũng không thể tạo ra các chức năng “động” được. Hiểu đơn giản, HTML giống như Microsoft Word, dùng để bố cục và định dạng trang web. Tóm lại, HTML là ngôn ngữ markup, dễ học, dễ hiểu, dễ áp dụng. Tuy nhiên, một website được viết bằng HTML rất đơn giản, nếu không muốn nói là nhàm chán. Để gây hứng thú với người truy cập, website cần có sự hỗ trợ của CSS và JavaScript. Nếu HTML cung cấp cấu trúc cơ bản của các trang web thì CSS và JavaScript sẽ cải tiến và sửa đổi website sinh động hơn.

**CSS là gì?**

CSS là viết tắt của cụm từ “Cascading Style Sheets”, tạm dịch: ngôn ngữ tạo phong cách cho trang web. Ngôn ngữ lập trình này quy định cách các thành phần HTML của trang web thực sự sẽ xuất hiện trên frontend như thế nào.

Hiểu đơn giản, CSS sẽ giúp webmaster xác định styles và định nghĩa nhiều loại nội dung của website. CSS được tạo ra để kết hợp với ngôn ngữ markup HTML để tạo phong cách cho trang web. Có 3 loại style CSS chính:

* Style CSS Internal: là style được tải lên mỗi khi trang web được refresh.
* Style CSS Inline: bạn có thể chỉnh sửa một yếu tố nào đó mà không cần truy cập trực tiếp vào file CSS.
* External style: bạn có thể tạo phong cách ở file khác áp dụng CSS vào trang bạn muốn. External style sẽ cải thiện thời gian tải trang rất nhiều. Đặc biệt, bạn có thể sử dụng External Style CSS để tạo phong cách cho nhiều trang cùng lúc.

**Vai trò của CSS**

Do đó, sử dụng CSS, bạn có thể:

* Tạo phong cách và định kiểu cho những yếu tố được viết dưới dạng ngôn ngữ đánh dấu, như HTML.
* Tiết kiệm công sức của lập trình viên nhờ điều khiển định dạng của nhiều trang web
* Phân biệt cách hiển thị của trang web với nội dung chính của trang bằng cách điều khiển bố cục, màu sắc và font chữ. Nếu HTML cung cấp các công cụ thô cần thiết để cấu trúc nội dung trên một trang web thì CSS sẽ giúp định hình kiểu nội dung này để trang web xuất hiện trước người dùng theo một cách đẹp hơn. Ví dụ: HTML sẽ đánh dấu từng phần văn bản để biết được đó là yếu tố gì. CSS sẽ sử dụng cấu trúc tiếng Anh đơn giản để tạo ra một bộ các quy tắc tạo phong cách cho các yếu tố. Có thể nói, CSS gần như tạo nên bộ mặt của một website. Và CSS cũng không phải là tất cả. Để có một trang web đẹp bạn không chỉ dựa vào CSS mà phải kết hợp với nhiều ngôn ngữ khác. Các ngôn ngữ phải được thực hiện dựa trên các bản thiết kế đã thống nhất.

**JavaScript là gì?**

Thường được viết tắt là “JS”, JavaScript là một ngôn ngữ lập trình được Brendan Eich (đồng sáng lập dự án Mozilla, quỹ Mozilla và tập đoàn Mozilla) cho ra mắt vào năm 1995 với tên LiveScript.

JavaScript được biết đến đầu tiên với tên Mocha, và sau đó là LiveScript, nhưng công ty Netscape đã đổi tên của nó thành JavaScript, bởi vì sự phổ biến như là một hiện tượng của Java lúc bấy giờ. Các slideshow, pop-up quảng cáo và tính năng autocomplete của Google đều được viết bằng JavaScript. JS có tác dụng giúp chuyển website từ trạng thái tĩnh sang động, tạo tương tác để cải thiện hiệu suất máy chủ và nâng cao trải nghiệm người dùng. Hiểu đơn giản, JavaScript là ngôn ngữ được sử dụng rộng rãi khi kết hợp với HTML/CSS để thiết kế web động.

**Vai trò của JavaScript**

Sử dụng JavaScript, bạn sẽ: Dễ dàng bắt đầu với các bước nhỏ, với thư viện ảnh, bố cục có tính thay đổi … nhờ sự linh hoạt của JavaScript. Có thể tạo ra các trò chơi, hoạt họa 2D hoặc 3D, ứng dụng cơ sở dữ liệu toàn diện, … Tăng cường các hành vi và kiểm soát mặc định của trình duyệt. JavaScript là ngôn ngữ lập trình cho phép các nhà phát triển web thiết kế các trang web tương tác. Hầu hết các hành vi động mà bạn sẽ thấy trên một trang web đều có được là nhờ JavaScript. Vì thế, đây là một ngôn ngữ phức tạp và khó học.

#### Tạo sao nên sử dụng TypeScript thay vì JavaScript

TypeScript là ngôn ngữ lập trình hướng đối tượng được phát triển và duy trì bởi Tập đoàn Microsoft. Nó là một siêu ký tự của JavaScript và chứa tất cả các yếu tố của nó. TypeScript hoàn toàn tuân theo khái niệm OOPS và với sự trợ giúp của TSC (Trình biên dịch TypeScript), chúng ta có thể chuyển đổi mã typecript (tệp .ts) thành JavaScript (tệp .js), dưới đây là những lý do chúng ta nên sử dụng TypeScript để lập trình giao diện website thay vì JavaScript:

* TypeScript đơn giản hóa mã JavaScript, giúp đọc và gỡ lỗi dễ dàng hơn.
* TypeScript là mã nguồn mở.
* TypeScript cung cấp các công cụ phát triển năng suất cao cho các IDE và thực tiễn JavaScript, như kiểm tra tĩnh.
* TypeScript làm cho mã dễ đọc và dễ hiểu hơn.
* Với TypeScript, chúng ta có thể cải thiện rất nhiều so với JavaScript đơn giản.
* TypeScript cung cấp cho chúng ta tất cả các lợi ích của ES6 (ECMAScript 6), cộng với năng suất cao hơn.
* TypeScript có thể giúp chúng ta tránh các lỗi đau đớn mà các nhà phát triển thường gặp phải khi viết JavaScript bằng cách kiểm tra mã.
* Hệ thống loại mạnh mẽ, bao gồm cả thuốc generic.
* TypeScript không có gì ngoài JavaScript với một số tính năng bổ sung.
* Cấu trúc, hơn là danh nghĩa.
* Mã TypeScript có thể được biên dịch theo tiêu chuẩn ES5 và ES6 để hỗ trợ trình duyệt mới nhất.
* Căn chỉnh với ECMAScript để tương thích.
* Bắt đầu và kết thúc bằng JavaScript.
* Hỗ trợ gõ tĩnh.
* TypeScript sẽ tiết kiệm thời gian cho các nhà phát triển.
* TypeScript là siêu bộ của ES3, ES5 và ES6.

### Cơ sở dữ liệu

#### Giới thiệu về cơ sở dữ liệu

**Cơ sở dữ liệu là gì? Database là gì?**

Cơ sở dữ liệu có tên tiếng anh là Database, viết tắt là CSDL. Là một tập hợp các dữ liệu có tổ chức, được lưu trữ và truy cập điện tử từ hệ thống máy tính. Khi cơ sở dữ liệu phức tạp hơn, chúng sẽ được phát triển bằng cách sử dụng các thiết kế và mô hình hóa hình thức. Khái niệm CSDL là gì còn được định nghĩa là một tập hợp thông tin có cấu trúc. Tuy nhiên, định nghĩa này chỉ được sử dụng trong lĩnh vực công nghệ thông tin và nó được hiểu rõ dưới dạng là một tập hợp các liên kết dữ liệu để lưu trữ trên thiết bị như đĩa hoặc băng. Dữ liệu này được duy trì dưới dạng tập hợp các thông tin trong hệ điều hành hay lưu trữ hóa trong hệ quản trị cơ sở dữ liệu. Tóm lại, có sở dữ liệu là hệ thống gồm rất nhiều thông tin, dữ liệu được xây dựng theo một cấu trúc nhất định để đáp ứng nhu cầu khai thác, sử dụng của nhiều người hay chạy nhiều chương trình cùng một lúc. Khi áp dụng hình thức lưu trữ này, sẽ khắc phục được những điểm yếu của việc lưu file thông thường trên máy tính. Các thông tin dữ liệu đảm bảo được sự nhất quán, hạn chế được việc trùng lặp thông tin.

**Hệ quản trị cơ sở dữ liệu là gì?**

Là chương trình phần mềm, thực hiện việc lưu trữ cơ sở dữ liệu. Hệ quản trị CSDL khi lưu trữ dữ liệu cần phải đảm bảo được tính cấu trúc trong cơ sở dữ liệu và cần phải hỗ trợ việc đọc, chỉnh sửa, thêm và xóa dữ liệu trên CSDL một cách dễ dàng.

**Cơ sở dữ liệu quan hệ là gì?**

Cơ sở dữ liệu quan hệ có tên tiếng anh là relational database. Là cơ sở dữ liệu số dựa trên mô hình quan hệ dữ liệu. Một hệ thống phần mềm sử dụng để duy trì CSDL là một hệ quản trị cơ sở dữ liệu quan hệ RDBMS. Nhiều hệ thống CSDL quan hệ có tùy chọn sử dụng SQL (ngôn ngữ truy vấn có cấu trúc) tiêu chuẩn để truy vấn và duy trì CSDL.

**Sự cần thiết của các hệ cơ sở dữ liệu**

Hệ cơ sở dữ liệu giữ một vai trò vô cùng quan trọng, đó là:

* Giảm bớt sự dư thừa dữ liệu trong lưu trữ :Trong các ứng dụng lập trình truyền thống, phương pháp tổ chức dữ liệu vừa tốt kém, lãng phí bộ nhớ, các thiết bị lưu trữ. Nhưng nếu tổ chức dữ liệu theo lý thuyết thì CSDL có thể hợp nhất các tệp dữ liệu lưu trữ của các bài toán, chương trình ứng dụng có thể cùng chia sẻ nguồn tài nguyên trên mạng cùng với hệ CSDL.
* Tổ chức lưu trữ dữ liệu theo lý thuyết CSDL có thể triển khai đồng thời trên nhiều ứng dụng cùng CSDL: Các ứng dụng không chỉ được sử dụng để chia sẻ chung tài nguyên dữ liệu mà còn trên cùng một CSDL để có thể triển khai đồng thời nhiều ứng dụng khác nhau trên các thiết bị sau cuối. Tổ chức dữ liệu theo cơ sở dữ liệu sẽ thống nhất các tiêu chuẩn, thủ tục và các biện pháp an toàn dữ liệu. Các hệ CSDL sẽ được quản trị bởi một hoặc một nhóm người quản trị CSDL. Người quản trị CSDL có thể áp dụng thống nhất các tiêu chuẩn, thủ tục,….giúp cho công việc bảo trì dữ liệu dễ dàng hơn.

#### Tạo sao nên sử dụng MySQL

### Các Framework và thư viện hỗ trợ

#### Giới thiệu về SpringBoot

Spring là một Framework phát triển các ứng dụng Java được sử dụng bởi hàng triệu lập trình viên. Nó giúp tạo các ứng dụng có hiệu năng cao, dễ kiểm thử, sử dụng lại code… Spring nhẹ và trong suốt (nhẹ: kích thước nhỏ, version cơ bản chỉ khoảng 2MB; trong suốt: hoạt động một cách trong suốt với lập trình viên) Spring là một mã nguồn mở, được phát triển, chia sẻ và có cộng đồng người dùng rất lơn. Spring Framework được xây dựng dựa trên 2 nguyên tắc design chính là: Dependency Injection và Aspect Oriented Programming. Những tính năng core (cốt lõi) của Spring có thể được sử dụng để phát triển Java Desktop, ứng dụng mobile, Java Web. Mục tiêu chính của Spring là giúp phát triển các ứng dụng J2EE một cách dễ dàng hơn dựa trên mô hình sử dụng POJO (Plain Old Java Object).Spring Boot là gì thì thực chất đây chính là một phần quan trọng của Spring framework. Đây chính là ứng dụng giúp làm giảm được sự phức tạp khi lập trình Spring cho các thành viên. Theo đó các thành viên chỉ phải tập trung cho việc nâng cao business cho ứng dụng. Spring được xem là một Framework mang đến khá nhiều ưu điểm cho việc phát triển phần mềm. The nhưng không thể phủ nhận rằng nó vẫn tồn tại nhiều hạn chế, đặc biệt là quá nhiều cấu hình được sử dụng trong phần mềm này. Chính vì thế, Spring Boot được ra đời đời nhằm khắc phục những điểm yếu và xây dựng một hướng phát triển mới, tốt hơn. Sử dụng Spring boot được xem là một tiêu chuẩn cho sự cho cấu hình giúp nâng cao năng suất cho developer. Được xem là cấu hình thiết kế phần mềm tối ưu nhất, tạo ra ứng dụng độc lập được ứng dụng trong tương lai. Đối với Spring Boot thì có thể thực hiện một dự án Spring được một cách nhanh chóng và tiến hành đơn giản nhất. Khi đó sẽ sử dụng cấu hình Sublime Text để phát triển luôn mà không cần phải thực hiện lại bước cài đặt netbean hay eclipse nữa.

#### Tại sao nên sử dụng SpringBoot

Như đã giới thiệu ở trên thì bạn có thể viết được nghĩa của Spring Boot là gì ? Sử dụng Spring Boot sẽ mang đến những lợi ích nổi bật như sau:

* Giúp tạo được ứng dụng độc lập dựa trên Spring, có thể tự chạy được java – jar
* Có ít cấu hình, có khả năng tự động cấu hình lại Spring khi cần, từ đó giúp các thành viên có thể tiết kiệm thời gian viết code và tăng thêm năng suất.
* Không cần phải triển khai file WAR mà thực hiện nhúng trực tiếp các ứng dụng server Giúp cung cấp nhiều plugin
* Spring Boot được phát triển tối ưu sao cho việc cấu hình XML trở nên đơn giản nhất trong Spring.
* Spring Boot được phát triển nhằm giúp người không có nhiều kiến thức lập trình vẫn có thể xây dựng ứng dụng.

Như vậy qua thông tin bài viết có thể giúp người đọc hiểu được nghĩa của Spring boot là gì. Theo đó có thể thấy, ứng dụng Spring boot này mang đến nhiều lợi ích cho các thành viên tăng năng suất hơn để tạo một web project. Chỉ cần ứng dụng theo đúng quy trình là sẽ giúp bạn đơn giản hóa các công đoạn.

#### Giới thiệu về Angular 2

Angular 2 là 1 framework UI để xây dựng ứng dụng web trên desktop và mobile. Nó được xây dựng dựa trên Javascript. Chúng ta có thể dùng nó để xây dựng 1 ứng dụng client side thú vị dùng HTML, CSS và Javascript. Angular 2 có rất nhiều cải tiến so với Angular 1 để dễ dàng học và phát triển các ứng dụng quy mô doanh nghiệp. Với angular 2 thì chúng ta dễ dàng xây dựng được 1 ứng dụng có thể dễ dàng mở rộng, bảo trì, kiểm nghiệm và chuẩn hóa ứng dụng của mình. Một số lợi ích của AngularJs 2:

* Nhanh hơn AngularJs 1
* Hỗ trợ đa nền tảng và đa trình duyệt.
* Cấu trúc code được tổ chức đơn giản hơn.
* Sử dụng dependency injection để maintane ứng dụng.
* Tất cả mọi thứ dựa vào component.

AngularJs 2 sử dụng TypeScript để phục vụ cho quá trình phát triển ứng dụng Angular. TypeScript là một ngôn ngữ có thể gọi là bao hàm Javascript, sử dụng TypeScript để viết các ứng dụng Angular và một trình compiler cho phép chúng ta biên dịch về ra các file javascript thuần nếu muốn. TypeScript cũng giúp giảm thiểu các lỗi khi viết các ứng dụng Angular

#### Tại sao nên sử dụng Angular 2

* Dưới đây là những tính năng nổi bật và cũng là những lý dó tại sao nên sử dụng Angular 2:  
  Two-way data binding Đây là 1 trong những tính năng tuyệt với nhất trong angular 2. Dữ liệu được binding một cách tự động và nhanh chóng, những thay đổi trong view sẽ được tự động cập nhật vào trong các component class.
* Powerful routing support Angular 2 hỗ trợ mạnh mẽ các routing thông qua cách tải trang không đồng bộ trên cùng 1 trang cho phép chúng ta tạo ra 1 single page application.
* Expressive HTML Angular 2 cho phép chúng ta dùng các cấu trúc lập trình như câu lệnh if, vòng lặp for, .. để render và kiểm soát các trang HTML.
* Modular by design Angular 2 được thiết kế theo hướng modul hóa để tổ chức và quản lý code 1 cách tốt hơn.
* Built in back end support Angular 2 được xây dựng để hỗ trợ việc giao tiếp với back-end servers và thực thi bất kỳ business logic hoặc lấy dữ liệu.
* Active community Angular 2 được hỗ trợ bởi google và có 1 cộng đồng đông đảo sẵn sàng hỗ trợ và giải đáp bất cứ câu hỏi nào của bạn.

# TÓM TẮT CHƯƠNG 1

Trong chương 1, đồ án đã phác họa khái quát các khái niệm cơ bản về Website, quy trình cũng như các công cụ để xây dựng Website. Trên cơ sơ đó, đồ án sẽ đi vào xây dựng hệ thống cho Website bán sách trực tuyến thông qua các yếu tố khảo sát thực tế và nhu cầu của khách hàng.

# CHƯƠNG 2. KHẢO SÁT CÁC YÊU CẦU CỦA HỆ THỐNG

Trong chương này chúng ta sẽ phân tích các yêu cầu của đề tài như việc tóm tắt hoạt động của hệ thống mà dự án sẽ được ứng dụng, phạm vi ứng dụng của đề tài và đối tượng sử dụng, mục đích của dự án. Xác định yêu cầu của khách hàng: Xuất phát từ hệ thống hiện hành của cửa hàng cùng với yêu cầu của khách hàng để vạch ra được yêu cầu hệ thống cần xây dựng như việc thiết kế giao diện, yêu cầu về chức năng và yêu cầu về bảo mật Trong chương này chúng ta đi xây dựng một số biểu đồ Use Case, biểu đồ hoạt động của các chức năng trong hệ thống và thông tin cơ sở dữ liệu của chương trình.

## Phân tích yêu cầu đề tài

Tóm tắt hoạt động của hệ thống mà dự án sẽ được ứng dụng

Chức năng:

* Đăng ký, đăng nhập để trở thành thành viên
* Hiển thị và bán mọi sách, khuyến mãi, bán chạy….
* Hiển thị danh mục các loại sách
* Tìm tiếm theo từ khóa và lọc sách theo các tiêu chí
* Hiển thị nhiều ảnh mô tả và thông tin chi tiết của sách
* Hiển thị gợi ý các sách cùng loại, cùng tác giả
* Tính năng mua hàng gửi thông tin đơn hàng qua email
* Thêm sản phẩm, cập nhật, xóa giỏ hàng
* Đặt hàng không cần tài khoản / Đã có tài khoản
* Xem và theo dõi trạng thái đơn đặt hàng đã mua
* Hỗ trợ Lấy lại mật khẩu, thay đổi thông tin tài khoản
* Quản lý theo thông tin tài khoản, đơn hàng
* Theo dõi tình trạng đơn hàng, hủy đơn
* Quản lý danh mục loại đa cấp, sắp xếp linh hoạt, ẩn hiên danh mục
* Quản lý sản phẩm (Cập nhật Trạng thái, Thêm, Sửa, Xóa)
* Quản lý đơn hàng (Xem chi tiết đơn hàng, thay đổi trạng thái)
* Quản lý thông tin khách hàng (Xem)
* Lược đồ thống kê đơn hàng, doanh thu, khách hàng (Số sản phẩm, Số đơn hàng, Số doanh thu)

## Phạm vi dự án được ứng dụng

Dành cho mọi đối tượng có nhu cầu mua sắm trên mạng. Do nghiệp vụ của cửa hàng kết hợp với công nghệ mới và được xử lý trên hệ thống máy tính nên công việc liên lạc nơi khách hàng cũng như việc xử lý hoá đơn thực hiện một cách nhanh chóng và chính xác. Rút ngắn được thời gian làm việc, cũng như đưa thông tin về các sản phẩm mới nhanh chóng đến cho khách hàng.

## Đối tượng sử dụng

Có 2 đối tượng sử dụng cơ bản là người dùng và nhà quản trị:

* **Người dùng**: Qua website, khách hàng có thể xem thông tin, lựa chọn những sản phẩm ưa thích ở mọi nơi thậm chí ngay trong phòng làm việc của mình.
* **Nhà quản trị**: Nhà quản trị có toàn quyền sử dụng và cập nhật sản phẩm, hoá đơn, quản lý khách hàng, đảm bảo tính an toàn cho website.

## Mục đích của dự án

* Thực phẩm hữu cơ là 1 sản phẩm không thể thiếu trong cuộc sống của tất cả hộ gia đình hiện nay.
* Thúc đẩy phát triển buôn bán trực tuyến.
* Rút ngắn khoảng cách giữa người mua và người bán, tạo ra một Website trực tuyến có thể đưa nhanh thông tin cũng như việc trao đổi mua bán các loại thực phẩm sạch qua mạng.
* Việc quản lý hàng trở nên dễ dàng.
* Sản phẩm được sắp xếp có hệ thống nên người dùng dễ tìm kiếm sản phẩm.

## Xác định yêu cầu của khách hàng

### Hệ thống hiện hành của cửa hàng

Hiện tại cửa hàng đang sử dụng cơ chế quảng cáo, hợp đồng trưng bày, đặt hàng trực tiếp tại trung tâm do các nhân viên bán hàng trực tiếp đảm nhận (hoặc có thể gọi điện đến nhà phân phối để đặt hàng và cung cấp địa chỉ để nhân viên giao hàng tận nơi cho khách hàng). Trong cả hai trường hợp trên khách hàng đều nhận hàng và thanh toán trực tiếp bằng tiền mặt tại nơi giao hàng.Và trong tờ hoá đơn khách hàng phải điền đầy đủ thông tin cá nhân và số lượng mặt hàng cần mua. Kế toán phải chuẩn bị hoá đơn thanh toán, nhập một số dữ liệu liên quan như ngày, giờ, tên khách hàng, mã số mặt hàng, số lượng mua, các hợp đồng trưng bày sản phẩm, tổng số các hoá đơn vào trong sổ kinh doanh.

Dựa vào những ràng buộc cụ thể của hệ thống hiện hành chúng ta có thể đề nghị một hệ thống khác tiên tiến hơn, tiết kiệm được thời gian…

## Hệ thống đề nghị

Để có thể vừa bán hàng và giới thiệu sản phẩm của cửa hàng trên mạng thì website cần có các phần như:

### Về giao diện

* Giao diện người dùng:
* Phải có một giao diện thân thiện dễ sử dụng.
* Phải nêu bật được thế mạnh của website, cũng như tạo được niềm tin cho khách hàng ngay từ lần viếng thăm đầu tiên.
* Giới thiệu được sản phẩm của cửa hàng đang có.
* Có biểu mẫu đăng ký để trở thành thành viên.
* Có danh mục sản phẩm.
* Chức năng đặt hàng, mua hàng.
* Có biểu mẫu liên hệ.
* Chức năng tìm kiếm sản phẩm.
* Mỗi loại sản phẩm cần phải có trang xem chi tiết sản phẩm.
* Mục login của khách hàng khi đã trở thành thành viên gồm có:
* Tên đăng nhập (Username)
* Mật khẩu đăng nhập (Password).
* Giao diện người quản trị:
* Quản lý danh mục, sắp xếp linh hoạt, ẩn hiên danh mục
* Quản lý sản phẩm (Cập nhật Trạng thái, Thêm, Sửa, Xóa)
* Quản lý đơn hàng (Xem chi tiết đơn hàng, Trạng thái đơn hàng, Hủy đơn)
* Quản lý thông tin khách hàng (Xem)
* Lược đồ thống kê đơn hàng, doanh thu, khách hàng (Số sản phẩm, Số đơn hàng, Số doanh thu)

### Về nghiệp vụ

* Người xây dựng nên hệ thống phải am hiểu về thương mại điện tử, hiểu rõ cách thức mua bán hàng qua mạng.
* Người xây dựng nên hệ thống đặc biệt phải hiểu rõ các loại sách trên thị trường: Có thể đưa ra đựơc một cái nhìn tổng quát, sâu rộng về sản phẩm.

### Về bảo mật

* Người Admin có toàn quyền giữ bảo mật cho website bằng mật khẩu riêng.
* Quản lý User & Password của khách hàng an toàn: Thông tin của khách hàng được bảo mật mã hóa mật khẩu bằng phương thức MD5

### Về hệ thống

* Phần cứng: PC bộ vi xử lý Pentium III, Ram 4GB trở lên, ổ cứng 128GB trở lên.
* Hệ điều hành: Windown 7 trở lên
* Phần mềm hỗ trợ: Postman, mySQL.

## Lựa chọn giải pháp

* Chương trình sử dụng ngôn ngữ Java, Springboot Framework và cơ sở dữ liệu MySQL để thiết kế hệ thống backend
* Chương trình sử dụng ngôn ngữ TypeScript, HTML, CSS, Bootstrap Framework và Angular Framework để thiết kế giao diện frontend
* Các công cụ mà hệ thống sử dụng:
* Java 8 - Ngôn ngữ lập trình
* MySQL - Cơ sở dữ liệu
* Angular 12 – Thiết kế giao diện
* Postman – kiểm tra Api của hệ thống
* Heroku – nền tảng deploy backend
* Netlify – nền tảng deploy frontend
* Google FireBase – nền tảng Cloud lưu trữ hình ảnh

# TÓM TẮT CHƯƠNG 2

Trong chương này, đồ án đã giới thiệu khái quát về đề tài của tác giả thông qua việc phân tích đề tài cụ thể như tóm tắt hoạt động của hệ thống mà dự án sẽ được ứng dụng, phạm vi dự án ứng dụng, đối tượng sử dụng, mục đích của dự án và tiến hành phân tích kĩ lượng yêu cầu của khách hàng. Chương 2 là tiền đề để tác giả lập kế hoạch phân tích thiết kế hệ thống trang web phù hợp với nhu cầu của khách hàng.

# CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ HỆ THỐNG BACKEND

## Phân tích các chức năng của hệ thống

Các tác nhân của hệ thống gồm có:

**Administrator**: là thành viên quản trị của hệ thống, có các quyền và chức năng như: tạo các tài khoản, quản lý sản phẩm, quản trị người dùng, quản lý hoá đơn…

**Member**: là hệ thống thành viên có chức năng: Đăng kí, đăng nhập, tìm kiếm, xem, sửa thông tin cá nhân, xem giỏ hàng, đặt hàng, xem thông tin về các hóa đơn đã lập.

**Customer**: Là khách vãng lai có chức năng: Đăng kí, tìm kiếm, xem thông tin sản phẩm, xem giỏ hàng, đặt hàng.

#### Danh mục use-case chung của hệ thống

***System***

Tìm kiếm thông tin



về sản phẩm

Xem thông tin về sản phẩm

Customer

Xem giỏ hàng

Thêm sản phẩm vào

giỏ hàng

Đăng kí thành viên

Đặt hàng



Member

Đăng nhập

Đăng xuất

Administrator

Đổi mật khẩu

X Sửa thông tina Member

Xem Lịch sử giao

dịch (HĐ đã lập)

Thêm hãng

Thêm sản

phẩm

Sửa thông tin sản phẩm

Xoá sản phẩm

Xoá Member

Liệt kê, cấp quyền, xoá User

Liệt kê, tìm kiếm

hóa đơn

Chuyển đơn hàng chưa

thanh toán->Đã thanh toán

**Hình 3.1: Biểu đồ Use Case**

**Bảng 1.1:** Use Case tổng quan về hệ thống

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Use case** | **Ý nghĩa/Ghi Chú** |
| **1** | Tìm kiếm thông tin sản phẩm | Use case này giúp người dùng tìm kiếm thông tin của sản phẩm |
| **2** | Xem thông tin sản phẩm | Use case này mô tả chức năng xem thông tin sản phẩm của người sử dụng như xem giá thành sản phẩm, tác giả, thể loại... |
| **3** | Xem giỏ hàng | Cho phép người dùng xem thông tin về giỏ hàng của mình, bao gồm các sản phẩm có trong giỏ hàng, số lượng sản phẩm, tổng giá |
| **4** | Thêm sản phẩm vào giỏ hàng | Cho phép người dùng thêm sản phẩm vào giỏ hàng |
| **5** | Đăng ký thành viên | Cho phép người dùng đăng ký thành viên |
| **6** | Đặt hàng | Use case này mô tả chức năng đặt hàng của khách hàng, khách hàng có điền thông tin giao hàng theo ý muốn. Có thể hủy bỏ việc đặt hàng nếu thay đổi ý định. |
| **7** | Đăng nhập | Cho phép người dùng đăng nhập vào hệ thống bằng tài khoản (username) và mật khẩu (password) |
| **8** | Đăng xuất | Cho phép người dùng đăng xuất ra khỏi hệ thống |
| **9** | Đổi mật khẩu | Cho phép người dùng đổi mật khẩu |
| **10** | Lấy lại mật khẩu | Cho phép người dùng lấy lại mật khẩu bằng cách điền chính xác thông tin về username và mail |
| **11** | Thay đổi thông tin tài khoản | Cho phép người dùng thay đổi thông tin tài khoản như: tên, số điện thoại, ảnh đại diện... |
| **12** | Xem lịch sử giao dịch | Cho phép người dùng đã có tài khoản được phép xem lịch sử giao dịch |
| **13** | Thêm tác giả | Cho phép Admin thêm tác giả |
| **14** | Thay đổi thông tin tác giả | Cho phép Admin thay đổi thông tin về tác giả như: tên, giới thiệu |
| **15** | Thêm sản phẩm | Cho phép Admin thêm sản phẩm |
| **16** | Thay đổi thông tin sản phẩm | Cho phép Admin thay đổi thông tin về sản phẩm như: giá, khuyến mãi, tên, thể loại... |
| **17** | Xoá sản phẩm | Cho phép Admin xoá sản phẩm |
| **18** | Liệt kê, xem thông tin Member | Cho phép Admin xem thông tin của các Member có trong hệ thống và các đơn hàng của những Member này |
| **19** | Liệt kê, xem thông tin đơn hàng của Member | Cho phép Admin xem các thông tin đơn hàng |
| **20** | Thay đổi trạng thái các đơn hàng của Member | Cho phép Admin thay đổi trạng thái các đơn hàng của Member như: huỷ, giao hàng |

#### Các chức năng của đối tượng Customer (Khách vãng lai)

Khi tham gia vào hệ thống thì họ có thể xem thông tin, tìm kiếm sản phẩm, đăng ký là thành viên của hệ thống, chọn sản phẩm vào giỏ hàng, xem giỏ hàng, đặt hàng.

#### Chức năng đăng ký thành viên

* Description: Dành cho khách vãng lai đăng ký làm thành viên để có thêm nhiều chức năng cho việc mua bán sản phẩm, cũng như tạo mối quan hệ lâu dài với công ty. Qua đó, thành viên sẽ được hưởng chế độ khuyến mại đặc biệt.
* Input: Khách vãng lai phải nhập đầy đủ các thông tin (\*: là thông tin bắt buộc)
* Tên tài khoản (\*)
* Email: (\*)
* Mật khẩu: (\*)
* Nhập lại mật khẩu (\*)
* Điện thoại: Nhập điện thoại. (\*)
* Process: Kiểm tra thông tin nhập. Nếu thông tin chính xác sẽ lưu thông tin vào CSDL và thêm thông tin của thành viên đó vào CSDL.
* Output: Đưa ra thông báo đăng ký thành công hoặc yêu cầu nhập lại nếu thông tin không hợp lệ hoặc tài khoản, email đã tồn tại.

#### Chức năng lọc theo giá tăng dần, giảm dần, theo A-Z…

* Input: Nhập vào hoặc lựa chọn thông tin tìm kiếm theo Giá, theo tên sản phẩm, theo chủng loại…
* Process: Lấy thông tin sản phẩm từ CSDL theo các trường tìm kiếm.

Output: Hiện chi tiết thông tin sản phẩm thỏa mãn yêu cầu tìm kiếm.

#### Chức năng xem thông tin sản phẩm

* Description: Cho phép xem chi tiết thông tin của sản phẩm.
* Input: Chọn sản phẩm cần xem.
* Process: Lấy thông tin sản phẩm từ CSDL thông qua ID.
* Output: Hiện chi tiết thông tin sản phẩm.

#### Chức năng giỏ hàng

* Description: Cho phép xem chi tiết giỏ hàng.
* Input: Click chọn vào giỏ hàng.
* Process: Lấy thông tin sản phẩm thông qua LocalStorage lưu các thông tin về mã sản phẩm và số lượng của từng sản phẩm mà khách hàng chọn vào giỏ hàng.
* Output: Hiển thị thông tin về tên sản phẩm, ảnh, đơn giá, số lượng, thành tiền, tổng tiền của các sản phẩm và giỏ hàng.

#### Chức năng đặt hàng

* Description: Cho phép tất cả các khách hàng có thể đặt hàng mà không cần phải là thành viên của hệ thống.
* Input: Sau khi xem giỏ hàng, khách hàng có nhu cầu đặt hàng qua mạng thì có thể click vào nút đặt hàng và điền một số thông tin cần thiết để người quản trị có thể xác nhận thông tin và giao hàng trực tiếp đến đúng địa chỉ một cách nhanh nhất có thể, nếu đang đăng nhập thì thông tin đặt hàng mặc định sẽ được lấy theo thông tin của tài khoản.
* Process: Lưu thông tin về khách hàng và thông tin hóa đơn đặt hàng vào các bảng trong cơ sở dữ liệu.
* Output: Đưa ra thông báo đơn đặt hàng đã được lập thành công hoặc không thành công.

### Các chức năng của đối tượng Member (thành viên)

Thành viên có tất cả các chức năng giống như khách vãng lai và còn có thêm một số chức năng khác như:

#### Chức năng đăng nhập

* Description: Cho Member login vào hệ thống.
* Input: Người dùng nhập vào các thông tin về username, password để login.
* Process: Kiểm tra username và password của người dùng nhập vào và so sánh với username và password trong CSDL.
* Output: Nếu đúng cho đăng nhập và hiển thị các chức năng của Member, ngược lại hiển thị thông báo yêu cầu nhập lại nếu thông tin không chính xác.

#### Chức năng đăng xuất

* Description: Cho phép người dùng đăng xuất khỏi hệ thống khi không còn nhu cầu sử dụng hệ thống.
* Input: Người dùng click vào nút thoát trên hệ thống.
* Process: Tiến hành xóa LocalStorage lưu thông tin đăng nhập để dừng phiên làm việc của tài khoản Member trong hệ thống.
* Output: Quay trở lại trang hiện hành. Ẩn hết các chức năng của Member.

#### Chức năng đổi mật khẩu

* Description: Cho phép thay đổi mật khẩu.
* Input: Người dùng điền thông tin mật khẩu cũ và mới.
* Process: Kiểm tra thông tin người dùng nhập vào. Nếu đúng thì cập nhật các thông tin mới, ngược lại thì không.
* Output: Hiển thị thông báo thành công nếu thông tin nhập vào chính xác hoặc hiển thị thông báo yêu cầu nhập lại nếu thông tin không chính xác.

#### Chức năng xem hóa đơn đã được lập

* Description: Cho phép Member xem các hóa đơn mua hàng của mình.
* Input: Click chọn lịch sử giao dịch trên menu của Member.
* Process: Gọi trang hiển thị thông tin hóa đơn đã được lập của Member đó
* Output: Hiển thị thông tin chi tiết các hóa đơn.

#### Chức năng lấy lại mật khẩu

* Description: Cho phép thay người dùng lấy lại mật khẩu.
* Input: Người dùng điền thông tin tên tài khoản và email.
* Process: Kiểm tra thông tin người dùng nhập vào. Nếu đúng thì hệ thống sẽ gửi mail báo cáo kèm theo mật khẩu mới, ngược lại thì không.
* Output: Hiển thị thông báo thành công nếu thông tin nhập vào chính xác hoặc hiển thị thông báo yêu cầu nhập lại nếu thông tin không chính xác.

Chức năng xem thông tin chi tiết các đơn đặt hàng (đang chờ được xử lý, đang giao, đã hoàn thành,hoặc hoá đơn bị huỷ bỏ):

* Description: Xem chi tiết đơn đặt hàng có trong CSDL
* Input: Chọn tên hoá đơn hoặc tên khách hàng của hoá đơn đó.
* Process: Lấy toàn bộ thông tin chi tiết của đơn đặt hàng có trong CSDL.
* Output: Hiển thị chi tiết thông tin trong đơn đặt hàng.

Chức năng chuyển các đơn đặt hàng:

* Description: Sau khi khách hàng xác nhận đặt hàng, hoá đơn sẽ được lưu vào trong CSDL. Khách hàng có thể tự huỷ bỏ trước khi Admin gọi điện cho khách hàng để xác nhận thông tin giao hàng. Sau khi đã nhận được đơn hàng thì người dùng có thể thay đổi đơn hành thành trạng thái đã hoàn thành hoặc Admin sẽ xử lý.
* Input: Người dùng chọn những đơn hàng cần xử lý
* Process: Xứ lý thay đổi của đơn hàng.
* Output: Load lại danh sách đơn hàng để xem lại sự thay đổi của hoá đơn.

### Chức năng của Administrator

#### **Các chức năng quản lý Member**

Chức năng xem thông tin Member:

* Description: Giúp Admin xem thông tin của member, danh sách các đơn hàng thuộc member đó
* Input: Chọn Member xem thông tin.
* Process: Lấy các thông tin của Member và danh sách các đơn hàng thuộc Member từ CSDL.
* Output:Hiển thị thông tin của Member và danh sách các đơn hàng thuộc Member .

#### Các chức năng quản lý Sản Phẩm

Chức năng chỉnh sửa thông tin sản phẩm:

* Description: Giúp Admin thay đổi thông tin của sản phẩm đã có trong CSDL.
* Input: Admin nhập thông tin mới của sản phẩm.
* Process: Cập nhật thông tin mới cho sản phẩm.
* Output: Hiển thị thông báo đã cập nhật sản phẩm.

Chức năng xoá sản phẩm:

* Description: Giúp Admin có thể xoá sản phẩm để sản phẩm không còn hiển thị trên trang Web nữa
* Input: Chọn sản phẩm cần xoá
* Process: Thay đổi trạng thái của sản phẩm trong CSDL
* Output: Load lại danh sách sản phẩm

Chức năng thêm sản phẩm:

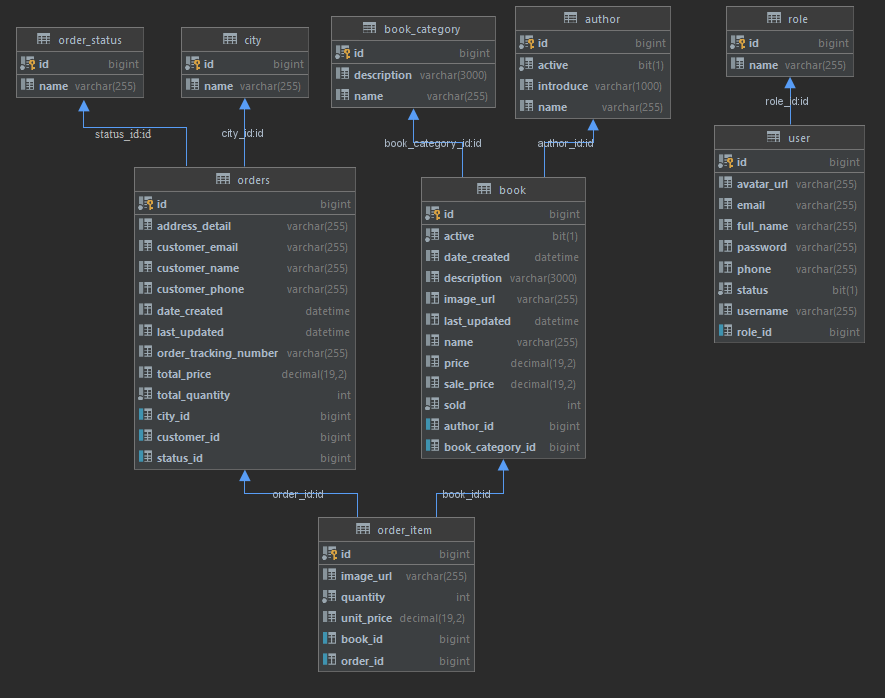
* Description: Giúp Admin có thể thêm sản phẩm mới.
* Input: Admin nhập vào những thông tin cần thiết của sản phẩm mới.
* Process: Kiểm tra xem những trường nào không được để trống. Nếu tất cả đều phù hợp thì thêm vào database. Ngược lại thì không thêm vào database.
* Output: Load lại danh sách sản phẩm để xem sản phẩm mới đã được thêm vào CSDL.

Chức năng xem thông tin chi tiết các đơn đặt hàng (đang chờ được xử lý, đang giao, đã hoàn thành,hoặc hoá đơn bị huỷ bỏ):

* Description: Xem chi tiết đơn đặt hàng có trong CSDL
* Input: Chọn tên hoá đơn hoặc tên khách hàng của hoá đơn đó.
* Process: Lấy toàn bộ thông tin chi tiết của đơn đặt hàng có trong CSDL.
* Output: Hiển thị chi tiết thông tin trong đơn đặt hàng.

Chức năng chuyển các đơn đặt hàng:

* Description: Sau khi khách hàng xác nhận đặt hàng, hoá đơn sẽ được lưu vào trong CSDL. Admin gọi điện cho khách hàng để xác nhận thông tin. Nếu đúng hoá đơn sẽ được chuyển sang trạng thái xác nhận đang giao còn không sẽ bị huỷ bỏ. Sau khi đã xác nhận đơn hàng được phía khách hàng nhận và thanh toán thì Admin sẽ thay đổi đơn hành thành trạng thái đã hoàn thành.
* Input: Admin chọn những đơn hàng cần xử lý
* Process: Xứ lý thay đổi của đơn hàng.
* Output: Load lại danh sách đơn hàng để xem lại sự thay đổi của hoá đơn.



## Biểu đồ phân tích thiết kế hệ thống

## Biểu đồ hoạt động

### Đăng nhập:

* Input: Username, password cùa người dùng.
* Output: Nếu đúng cho đăng nhập và hiển thị các chức năng của Member, ngược lại hiển thị thông báo yêu cầu nhập lại nếu thông tin không chính xác.

Mô tả thuật toán:

* Bước 1: Nhập username và password
* Bước 2: Kiểm tra thông tin đăng nhập với CSDL
* Bước 3: Nếu username và password không đúng, hiển thị thông báo yêu cầu nhập lại hoặc tài khoản chưa kích hoạt,Nếu username và password đúng, hiển thị thông báo đăng nhập thành công và lưu thông tin đăng nhập vào session

Begin

Nhập username, password

Hiển thị thông báo yêu cầu nhập lại thông tin hoặc tài khoản chưa kích hoạt

Thông tin đăng nhập phù hợp với CSDL ?

No

Yes

End

Hiển thị thông báo đăng nhập thành công và lưu thông tin đăng nhập vào session

**Hình 3.3: Biểu đồ hoạt động đăng nhập vào hệ thống**

### Đăng xuất:

* Input: Người dùng click vào nút thoát trên hệ thống.
* Output: Quay trở lại trang hiện hành. Ẩn hết các chức năng của Member.

Mô tả thuật toán:

* Bước 1: Click vào nút thoát hệ thống
* Bước 2: Xóa session lưu thông tin đăng nhập
* Bước 3: Hiển thị thông báo đăng xuất thành công

**Begin**

**Click vào link**

**logout**

**Thông báo logout**

**thành công**

**End**

**Xóa session lưu thông tin đăng nhập**

**Hình 3.4: Biểu đổ hoạt động thoát ra khỏi hệ thống**

### Đăng kí thành viên:

* Input: Các thông tin của User (họ tên, email, mật khẩu, địa chỉ,…).
* Output: Đưa ra thông báo đăng ký thành công hoặc yêu cầu nhập lại nếu thông tin không hợp lệ.

Mô tả thuật toán:

* Bước 1: User nhập thông tin cá nhân cần thiết Bước 2: Kiểm tra thông tin đã nhập với CSDL
* Bước 3: Nếu các thông tin đã nhập chưa tồn tại, cập nhật thông tin User vào CSDL,Nếu các thông tin đã nhập đã tồn tại, hiển thị thông báo yêu cầu nhập lại Username khác

Begin

User nhập thông tin Cá nhân cần thiết

Hiển thị thông báo yêu cầu nhập lại UserName khác

Kiểm tra trong CSDL Xem User đó đã tồn tại hay

chưa ?

Yes

No

**Hình 3.5: Biểu đồ hoạt động đăng kí thành viên**

Hiển thị thông báo thành công

End

Cập nhật thông tin về Người sử dụng vào CSDL

### Xem thông tin về sản phẩm:

* Input: Click chọn sản phẩm.
* Output: Hiển thị chi tiết thông tin sản phẩm đã chọn.

Mô tả thuật toán:

* Bước 1: Click chọn sản phẩm cần xem
* Bước 2: Lấy mã sản phẩm từ biến session
* Bước 3: Lấy thông tin của sản phẩm có mã đó trong CSDL
* Bước 4: Hiển thị thông tin chi tiết sản phẩm đã có trong CSDL

Begin

Click chọn vào sản phẩm cần xem

Lấy Mã SP từ biến session

Lấy thông tin của SP có mã đó trong CDSL

Hiển thị thông tin chi tiết SP đã có trong CSDL

End

**Hình 3.6: Biểu đồ hoạt động xem thông tin về sản phẩm**

### Thêm sản phẩm vào giỏ hàng:

* Input: Click chọn thêm sản phảm vào giỏ hàng.
* Output: Thêm sản phẩm vào giỏ hàng và hiển thị sản phẩm đã được thêm vào giỏ hàng.

Mô tả thuật toán:

* Bước 1: Chọn sản phẩm muốn thêm vào giỏ hàng
* Bước 2: Kiểm tra biến session với mã của sản phầm này
* Bước 3: Nếu chưa có lưu mã sản phẩm và số lượng vào biến session
* Bước 4: Quay lại trang sản phẩm, tiếp tục chọn sản phẩm muốn thêm vào giỏ hàng



Begin

Xem sản phẩm Click chọn sản phẩm

thêm vào giỏ hàng

Kiểm tra biến session đã có Mã SP này chưa ?

No

Lưu Mã SP và số lượng vào biến session

Tiếp tục thêm sản phẩm vào giỏ hàng

End

**Hình 3.7: Biểu đồ chức năng thêm SP vào giỏ hàng**

### Đặt hàng:

* Input: Click chọn nút đặt hàng và điền thông tin cần thiết để giao hàng.
* Output: Cập nhật thông tin sản phẩm được chọn vào CSDL, hiển thị thông báo đặt mua thành công.

Mô tả thuật toán:

* Bước 1: Chọn và nhập số lượng sản phẩm
* Bước 2: Cập nhật thông tin sản phẩm được chọn vào CSDL Bước 3: Hiển thị thông báo thành công

Begin

Member chọn và nhập số lượng sản phẩm

Hiển thị thông báo thành công

End

Cập nhật thông tin sản phẩm được chọn vào CSDL

**Hình 3.8: Biểu đồ chức năng đặt hàng**

### Đổi mật khẩu:

* Input: Điền thông tin mật khẩu cũ và mới.
* Output: Hiển thị thông báo thành công nếu thông tin nhập vào chính xác hoặc hiển thị thông báo yêu cầu nhập lại nếu thông tin không chính xác.

Mô tả thuật toán:

* Bước 1: Chọn chức năng đổi mật khẩu Bước 2: Nhập mật khẩu cũ và mới
* Bước 3: Kiểm tra mật khẩu cũ vừa nhập với CSDL
* Bước 4: Nếu đúng, lưu mật khẩu mới vào CSDL và hiển thị thông báo đổi mật khẩu thành công,nếu sai, hiển thị thông báo yêu cầu nhập lại

Begin

Member chọn chức năng đổi mật khẩu

Nhập mật khẩu cũ và mật khẩu mới

Hiển thị thông báo nhập lại thông tin thật chính xác

Kiểm tra mật khẩu cũ vừa nhập của Member đó

đã đúng chưa ?

No

Yes

Lưu mật khẩu mới của Member đó vào CSDL

End

**Hình 3.9: Biểu đồ chức năng đổi mật khẩu**

### Xoá Member:

* Input: Click chọn Member muốn xóa.
* Output: Hiển thị thông tin Member đã chọn, cập nhật lại danh sách Member để biết được đã xoá thành công Member ra khỏi CSDL.

Mô tả thuật toán:

* Bước 1: Chọn Member muốn xóa và click chọn biểu tượng xóa Bước 2: Xác nhận lại chắc chắn xóa
* Bước 3: Nếu đồng ý xóa, xóa Member đó ra khỏi CSDL Nếu không đồng ý xóa, quay trở lại bước 1.

Begin

Tìm Member muốn xoá và chọn biểu tượng xoá

No

Xác nhận lại thông tin xoá?

Yes

End

Xoá Member đó ra khỏi CSDL

**Hình 3.10: Biểu đồ chức năng xoá Member**

### Thêm sản phẩm:

* Input: Nhập thông tin về sản phẩm muốn thêm.
* Output: Cập nhật thông tin của sản phầm vào CSDL, hiển thị thêm sản phẩm thành công.

Mô tả thuật toán:

* Bước 1: Nhập thông tin về sản phẩm muốn thêm
* Bước 2: Kiểm tra thông tin vừa nhập với CSDL
* Bước 3: Nếu sản phẩm đã tồn tại, hiển thị thông báo yêu cầu nhập lại Nếu sản phẩm chưa tồn tại, cập nhật thông tin của sản phẩm vào CSDL

Begin

Nhập thông tin về sản phẩm cần thêm

Hiển thị thông báo yêu cầu nhập lại sản phẩm khác

Kiểm tra trong CSDL Xem Sản phẩm đó đã tồn tại

hay chưa ?

Yes

No

**Hình 3.11: Biểu đồ chức năng thêm sản phẩm**

End

Cập nhật thông tin về Sản phẩp đó vào CSDL

Lưu thông tin sản phẩm vừa sửa vào CSDL

### Sửa thông tin sản phẩm:

* Input: Nhập thông tin chỉnh sửa của sản phẩm được chọn
* Output: Hiển thị thông báo đã cập nhật sản phẩm.

Mô tả thuật toán:

Bước 1: Chọn sản phẩm muốn thay đổi và nhập thông tin

Bước 2: Click chọn lưu để lưu thông tin sản phẩm vừa sửa vào CSDL

Begin

Chọn sản phẩm và thay đổi thông tin về sản phảm cần sửa

End

**Hình 3.12: Biểu đồ chức năng sửa thông tin sản phẩm**

### Xóa sản phẩm:

* Input: Chọn sản phẩm muốn xóa.
* Output: Xóa thông tin sản phẩm trong CSDL, cập nhật lại danh sách sản phẩm.

Mô tả thuật toán:

* Bước 1: Chọn sản phẩm muốn xóa
* Bước 2: Hiển thị thông báo xác nhận thông tin xóa
* Bước 3: Nếu xác nhận xóa, xóa thông tin sản phẩm đó khỏi CSDL Nếu xác nhận không xóa, quay lại bước 1
* Bước 4: Cập nhật lại danh sách sản phẩm

Begin

Tìm sản phẩm muốn xoá và chọn biểu tượng xoá

No

Xác nhận lại thông tin xoá?

Yes

End

Xoá sản phẩm đó ra khỏi CSDL

**Hình 3.13: Biểu đồ chức năng xoá sản phẩm**

### Use-case quản lý sản phẩm



|  |
| --- |
| **Interface** |
| + Danh sách sản phẩm |
| + Thêm() : int  +Sửa () : int  + Xoá() : int |

+ Chi tiết SP () : void

+ Thông tin

**Control**

Admin



**Hình 3.15: Use-case quản lý sản phẩm**

|  |
| --- |
| **login** |
| * username * pass |
| + login () |

|  |
| --- |
| **Database** |
|  |
|  |

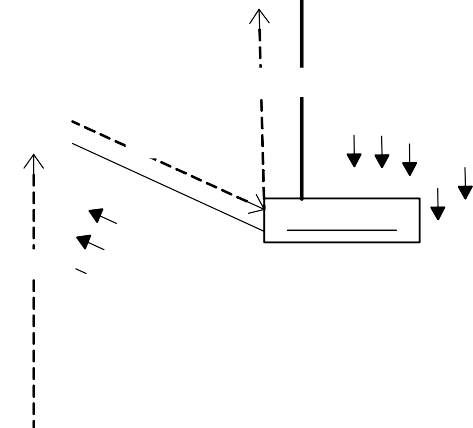
|  |  |
| --- | --- |
|  | **Process** |
|  | - ID sản phẩm : int |
| + Thêm() : int  +Sửa () : int  + Xoá() : int |

Lớp biên: Login, Interface

Lớp điều khiển: Control

Lớp thực thể: Database

/ Kiểm tra ( user / pass )



Ðăng nhập(user/pass)

Kết quả

Admin

Kết quả

C\_ontrol

Kết quả

Database

P\_rocess

\_Login

PageLoad() Danh sách TL() Thêm()

Sửa()

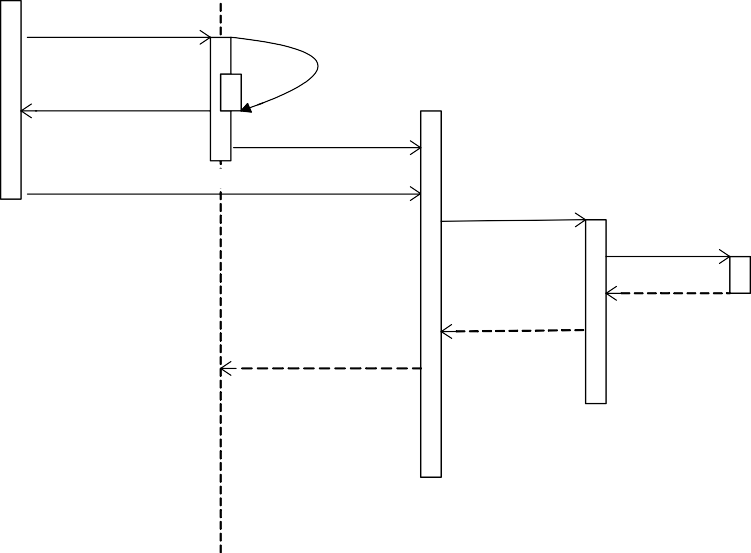
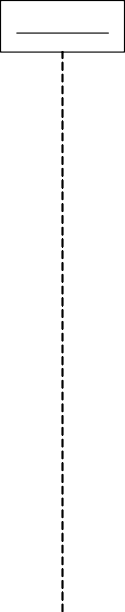
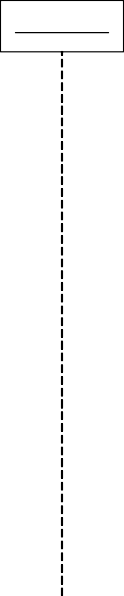
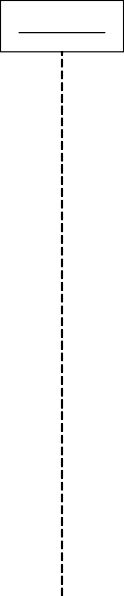
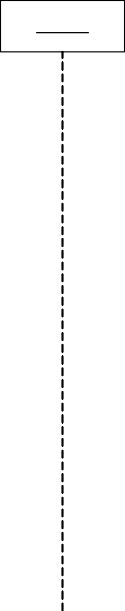
Xoá()

Thêm();sửa();Xoá()



**Hình 3.16: Biểu đồ cộng tác chức năng cập nhật**

### Chức năng thêm mới sản phẩm



Admin

Control

P\_rocess

Dat\_abase

Đăng nhập( user / pass)

Kiểm tra (user / pass)

Yêu cầu chọn thao tác

Hiển thị danh sách()

Thêm mới ()

Thêm mới ()

Thêm mới ()

Kết quả

Kết quả

Kết quả

l\_ogin

**Hình 3.17: Biểu đồ tuần tự chức năng thêm sản phẩm mới**

### Chức năng sửa thông tin sản phẩm

Admin

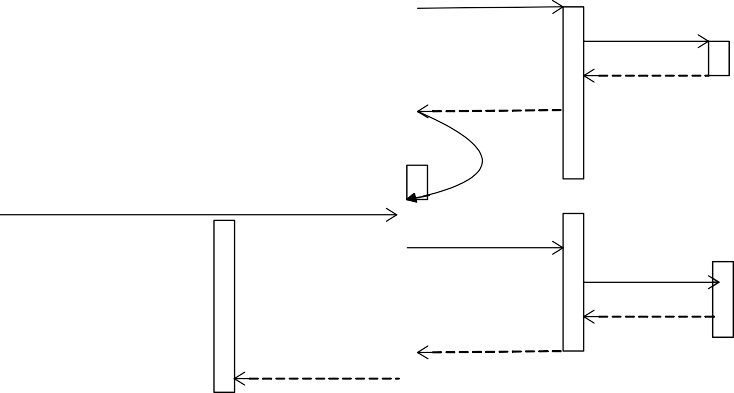
login

Control

P\_rocess

Database

**Hình 3.18: Biểu đồ tuần tự chức năng sửa thông tin sản phẩm**



Đăng nhập( user / pass)

Kiểm tra( user/ pass)

Chọn chỉnh sửa SP

Danh sách SP()

Chọn sản phẩm ()

Lấy SP (id)

Lấy SP (id)

Kết quả

Kết quả

Hiển thị()

Chỉnh sửa ()

Chỉnh sửa ()

Sửa()

Kết quả

Kết quả

Kết quả

Yêu cầu thao tác

### Chức năng xóa sản phẩm

Admin

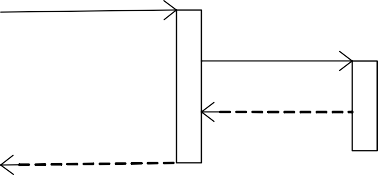
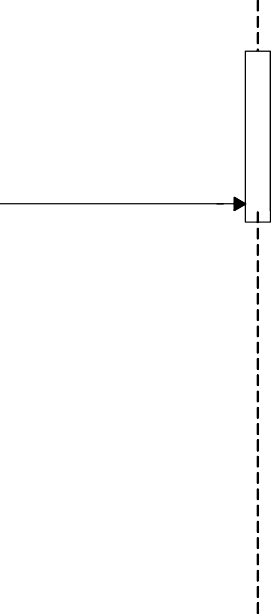
l\_ogin

Control

P\_rocess

Dat\_abase

**Hình 3.19: Biểu đồ tuần tự chức năng xóa sản phẩm**



Đăng nhập( user / pass)

Kiểm tra (user / pass)

Yêu cầu chọn thao tác

Chọn xoá SP Danh sách tài liệu( Chọn sản phẩm()

Xoá(id)

Xoá(id)

Kết quả

Kết quả

Kết quả

Thao tác Xoá()

)

Tương tự đối với Use-case quản lý người dùng hệ thống và quản lý khách hàng (với các chức năng thêm, sửa, xóa).

### Chức năng đăng nhập

Proccess

1:[Chọn chức năng Đăng nhập] Gửi yêu cầu đăng nhập

2:Nhập UserName và Password

3: Kiểm tra User

4: Kiểm tra

5: Xác thực thành công

6: Xác thực thành công

7: Thông báo đăng nhập thành công

8:[Chọn chức năng đổi mật khẩu] Gửi yêu cầu đổi mật khẩu

9: Hiện chức năng đổi mật khẩu

10: Nhập và xác nhận Password mới

11: Thay đổi Password

12: Thay đổi Password

15: Thông báo đổi Password thành công

13: Xác nhận thay đổi thành công

14: Xác nhận thay đổi thành công

Database

Control

Interface

Member

**Hình 3.20: Biểu đồ tuần tự chức năng đăng nhập**

### C***hức năng đăng ký thành viên***

Proccess

Custome

1:[Chọn chức năng đăng kí thành viên] Gửi yêu cầu

3: Nhập các thông tin đăng ký

4: Kiểm tra UserName này đã có trong CSDL chưa

5: Kiểm tra

7: Xác nhận đã tồn tại

Gọi lại form Đăng ký để nhập lại

{OR}

6: Xác nhận đăng ký thành công

7: Xác nhận thành công

8: Thông báo đăng ký thành viên thành công

6:(Nếu có) Xác nhận đã tồn tại

2: Hiển thị form đăng ký

Database

Control

Interface

Customer

**Hình 3.21: Biểu đồ tuần tự chức năng đăng ký thành viên**

## Cách tổ chức dữ liệu và thiết kế chương trình cho trang Web

Việc tổ chức dữ liệu phải giải quyết được các yêu cầu đã phân tích để lúc hiển thị lên trang Web có giao diện dễ nhìn, dễ dàng, hấp dẫn khách hàng... Một phương pháp sử dụng khá rộn rãi từ trước đến nay là quản lí theo từng nhóm sản phẩm. Mỗi nhóm sẽ có một mã nhóm và tên nhóm để nhận biết. Đối với Website này, với các sản phẩm, các sản phẩm sẽ tương ứng với mã hãng và mã loại sản phẩm của nó và mã đó sẽ làm khóa chính trong bảng đó. Sau đây là một số bảng cơ sở dữ liệu chính trong toàn bộ cơ sở dữ liệu:

### Bảng 1: Bảng book\_category(thể loại sách)

*Dùng để quản lí danh mục loại sản phẩm*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **id** | bigint | Null | ID của loại sách |
| name | varchar(255) | Null | Tên loại sách |
| description | varchar(1000) | Null | Chú thích về loại sách |

### Bảng 2: Bảng author(tác giả)

*Bảng cấu hình mail\_smtp*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **id** | bigint | Null | ID của tác giả |
| name | varchar(255) | Null | Tên của tác giả |
| introduce | varchar(1000) | Null | Giới thiệu về tác giả |
| active | bit(1) | 1 | Trạng thái hoạt động |

### Bảng 3: Bảng book(sách)

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | bigint | Null | ID của sách |
| active | bit(1) | 1 | Trạng thái hoạt động |
| date\_created | datetime | Null | Thời gian khởi tạo |
| description | varchar(3000) | Null | Giới thiệu về sách |
| image\_url | varchar(255) | Null | Đường dẫn hình ảnh |
| last\_update | datetime | Null | Thời gian chỉnh sửa cuối cùng |
| name | varchar(255) | Null | Tên sách |
| price | decimal(19,2) | Null | Giá của sách |
| sale\_price | decimal(19,2) | 0 | Giá khuyến mãi |
| sold | int | 0 | Số lượng đã bán |
| author\_id | bigint | Null | Thuộc về tác giả |
| book\_category\_id | bigint | Null | Thuộc về thể loại |

### Bảng 4: city

*Bảng nội dung (phần tin tức)*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | Int(20) | Null | Mã |
| title | varchar(255) | Null | Tiêu đề |
| alias | varchar(255) | Null | Link bài viết |
| introtext | mediumtext | Null | Mô tả ngắn |
| fulltext | mediumtext | Null | Bài viết đầy đủ |
| img | varchar(255) | Null | Hình ảnh |
| created | datetime | Null | Được tạo |
| created\_by | varchar(50) | Null | Được tạo bởi |
| modified | datetime | Null | Sửa đổi |
| modified\_by | varchar(50) | Null | Sửa đổi bởi |
| trash | int(1) | Null | Thùng rác |
| status | int(1) | Null | Trạng thái |

### Bảng 5: db\_customer

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | Int(20) | Null | Mã ID |
| fullname | varchar(100) | Null | Tên đầy đủ |
| username | varchar(100) | Null | Tên đăng nhập |
| password | varchar(32) | Null | Mật khẩu |
| address | varchar(100) | Null | Địa chỉ |
| phone | varchar(13) | Null | Số điện thoại |
| email | varchar(50) | Null | Email |
| created | datetime | Null | Được tạo bởi |
| trash | int(1) | Null | Thùng rác |
| status | int(1) | Null | Trạng thái |

### Bảng 6: db\_ discount

*Bảng mã giảm giá*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | Int(20) | Null | Mã ID |
| code | varchar(100) | Null | Mã giảm giá |
| discount | int(11) | Null | Số tiền |
| limit\_number | int(11) | Null | Giới hạn mua |
| number\_used | int(11) | Null | Số lượng đã nhập |
| expiration\_date | date | Null | Ngày hết hạn |
| payment\_limit | int(11) | Null | Giới hạn đơn hàng |
| description | varchar(255) | Null | Mô tả |
| created | date | Null | Được tạo bởi |
| orders | int(11) | Null | Đơn đặt hàng |
| trash | int(1) | Null | Thùng rác |
| status | int(1) | Null | Trạng thái |

### Bảng 7: db\_district

*Bảng địa chỉ*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | Int(5) | Null | Mã ID |
| name | varchar(100) | Null | Tên |
| type | varchar(30) | Null | Kiểu |
| provinceid | int(5) | Null | Tỉnh |

### Bảng 8: db\_ order

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | Int(20) | Null | Mã ID |
| orderCode | varchar(8) | Null | Mã đặt hàng |
| customerid | int(11) | Null | ID khách hàng |
| orderdate | datetime | Null | Ngày đặt hàng |
| fullname | varchar(100) | Null | Tên đầy đủ |
| phone | varchar(100) | Null | Số điện thoại |
| money | Int(12) | Null | Số tiền |
| price\_ship | int(11) | Null | Phí vận chuyển |
| coupon | Int(11) | Null | Mã giảm giá |
| province | Int(5) | Null | Tỉnh |
| district | int(5) | Null | Huyện |
| address | varchar(255) | Null | Địa chỉ |
| trash | int(1) | Null | Thùng rác |
| status | int(1) | Null | Trạng thái |

### Bảng 9: db\_ orderdetail

*Bảng chi tiết đặt hàng*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | int(11) | Null | Mã ID |
| orderid | int(11) | Null | Orderid |
| productid | int(11) | Null | ID sản phẩm |
| count | int(11) | Null | count |
| price | int(11) | Null | Giá bán |
| trash | Tinyint(1) | Null | Thùng rác |
| status | Tinyint(1) | Null | Trạng thái |

### Bảng 10: db\_producer

*Bảng nhà phân phối*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | int(11) | Null | Mã ID |
| name | varchar(255) | Null | Tên nhà phân phối |
| code | varchar(100) | Null | ID |
| keyword | varchar(255) | Null | Từ khóa |
| created\_at | datetime | Null | Được tạo lại |
| created\_by | int(11) | Null | Được tạo bởi |
| modified | datetime | Null | Sửa đổi |
| modified\_by | int(11) | Null | Sửa đổi bởi |
| trash | int(11) | Null | Thùng rác |
| status | int(11) | Null | Trạng thái |

### Bảng 11: db\_product

*Bảng đặt hàng*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Kiểu giá trị trong CSDL | Giá trị mặc định | Chú thích |
| **Id** | Int(20) | Null | Mã ID |
| catid | int(11) | Null | catid |
| name | varchar(255) | Null | Tên |
| alias | varchar(255) | Null | Bí danh |
| avatar | varchar(255) | Null | Ảnh đại diện |
| img | varchar(255) | Null | Hình ảnh |
| sortDesc | text | Null | sortDesc |
| detail | text | Null | Chi tiết |
| producer | int(11) | Null | Nhà phân phối |
| number | int(11) | Null | Số |

|  |  |  |  |
| --- | --- | --- | --- |
| number\_buy | int(11) | Null | Số mua |
| sale | int(3) | Null | Khuyến mãi |
| price | int(11) | Null | Giá bán |
| price\_sale | int(11) | Null | Giá khuyến mãi |
| created | datetime | Null | Tạo |
| created\_by | datetime | Null | Tạo bởi |
| modified | datatime | Null | Sửa đổi |
| modified\_by | varchar(100) | Null | Sửa đổi bởi |
| trash | int(1) | Null | Thùng rác |
| status | int(1) | Null | Trạng thái |

### Bảng 12: db\_province

*Bảng tỉnh*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id** | Int(5) | Null | Mã ID |
| name | varchar(100) | Null | Tên |
| type | varchar(30) | Null | Kiểu |

### Bảng 13: db\_slider

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id** | Int(5) | Null | Mã ID |
| name | varchar(255) | Null | Tên |
| link | varchar(255) | Null | Kiểu |
| img | varchar(100) | Null | Tên |
| created | datetime | Null | Được tạo lại |
| created\_by | varchar(50) | Null | Được tạo bởi |
| modified | datetime | Null | Sửa đổi |

|  |  |  |  |
| --- | --- | --- | --- |
| modified\_by | varchar(50) | Null | Sửa đổi bởi |
| trash | int(1) | Null | Thùng rác |
| status | int(1) | Null | Trạng thái |

### Bảng 14: db\_user

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id** | Int(5) | Null | Mã ID |
| fullname | varchar(255) | Null | Tên |
| username | varchar(255) | Null | Kiểu |
| password | varchar(64) | Null | Tên |
| role | int(11) | Null | Được tạo lại |
| email | varchar(255) | Null | Được tạo bởi |
| gender | int(1) | Null | Sửa đổi |
| phone | varchar(15) | Null | Sửa đổi bởi |
| address | varchar(255) | Null | Sửa đổi bởi |
| img | varchar(255) | Null | Sửa đổi bởi |
| created | datetime | Null | Sửa đổi bởi |
| trash | int(1) | Null | Thùng rác |
| status | int(1) | Null | Trạng thái |

### Bảng 15: db\_usergroup

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id** | Int(5) | Null | Mã ID |
| name | varchar(255) | Null | Tên |
| created | datetime | Null | Kiểu |
| created\_by | Int(11) | Null | Tên |
| modified | datetime | Null | Được tạo lại |

|  |  |  |  |
| --- | --- | --- | --- |
| modified\_by | Int(11) | Null | Được tạo bởi |
| trash | tinyint(1) | Null | Thùng rác |
| access | tinyint(1) | Null | Sửa đổi bởi |
| status | tinyint(1) | Null | Trạng thái |

## Yêu cầu về bảo mật

Có tính bảo mật cao, đảm bảo nếu không có được username, password và được cho phép hoạt động của Admin thì không có một User nào có thể thay đổi thông tin về sản phẩm cũng như toàn bộ Website ngoài việc thay đổi thông tin cá nhân của User vì vậy việc quản lý dữ liệu bằng mật khẩu đăng nhập:

Tên đăng nhập

Mật khẩu đăng nhập

Các thông tin của khách hàng được bảo mật.

Hệ thống quản trị tuyệt đối an toàn, không thể bị truy cập do lỗi của hệ thống.

Như vậy, từ các biểu đồ Use Case và biểu đồ hoạt động của hệ thống đã thể hiện được các chức năng của các tác nhân và hệ thống làm việc như thế nào, xuất phát từ hệ thống hiện tại và các yêu cầu về hệ thống, yêu cầu khách hàng, chúng ta nên xây dựng cho trang Website có đầy đủ chức năng của người quản trị và người dùng và phải đảm bảo có được tính bảo mật cao.

# TÓM TẮT CHƯƠNG 3

Sau khi thực hiện bước khảo sát ở chương 2, chương 3 đã kết hợp với cơ sở lý thuyết để đi vào thiết kế trang web bán hàng cụ thể là lập ra các thuật toán cho giao diện và các chức năng hệ thống của người dùng, người quản trị. Ngoài ra đưa đến cái nhìn rõ hơn về các biểu đồ tác giả sử dụng để phân tích thiết kế hệ thống, cách tổ chức dữ liệu và thiết kế chương trình, cuối cùng là thiết kế về bảo mật trang web.

# CHƯƠNG 4. PHÂN TÍCH THIẾT KẾ GIAO DIỆN FRONTEND

## Phân tích yêu cầu về mặt giao diện

### Giao diện người dùng

Phải có một giao diện thân thiện dễ sử dụng:

* Màu sắc hài hoà làm nổi bật hình ảnh của sản phẩm, font chữ thống nhất, tiện lợi khi sử dụng.
* Giới thiệu những thông tin về cửa hàng: Địa chỉ liên hệ của cửa hàng hoặc các đại lý trực thuộc, các số điện thoại liên hệ, email liên hệ.
* Sản phẩm của cửa hàng: sản phẩm đang có, sản phẩm bán chạy, sản phẩm mới nhất, sản phẩm sắp ra mắt…

### Giao diện người quản trị

* Giao diện đơn giản, dễ quản lý dữ liệu.
* Phải được bảo vệ bằng **User & Password** riêng của **Admin**.

## Thiết kế

### Đăng ký tài khoản

### Đăng nhập

### Header

### SiderBar

## Giao diện

### Giao diện trang chủ

### Giao diện form đăng ký

### Giao diện form đăng nhập

### Giao diện trang giỏ hang

### Giao diện trang chi tiết sản phẩm

### Giao diện trang đặt hàng

### Giao diện trang quản lý đơn hàng

### Giao diện trang quản lý của Admin



**Hình 4.1: Giao diện trang chủ**

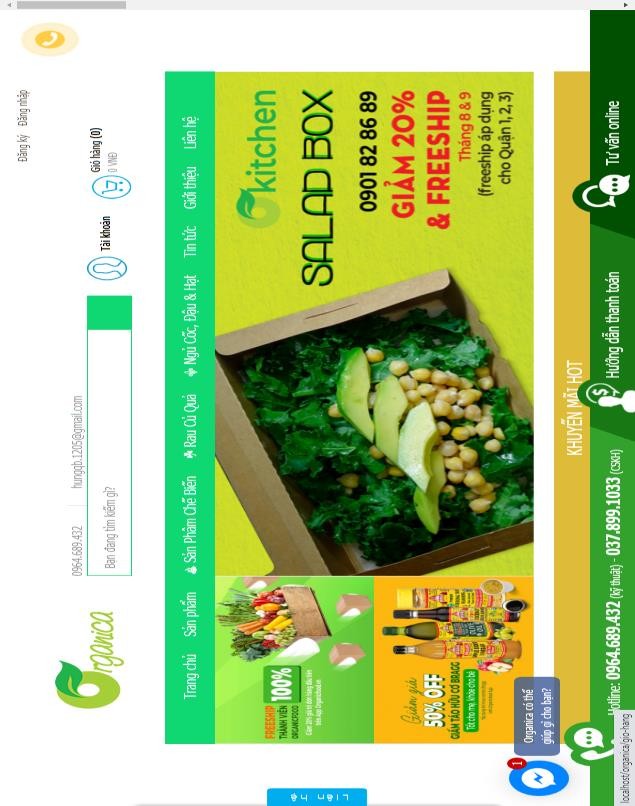
Trang chủ *“Index.php”*: Là trang chính, chứa tất cả các mục để lựa chọn.

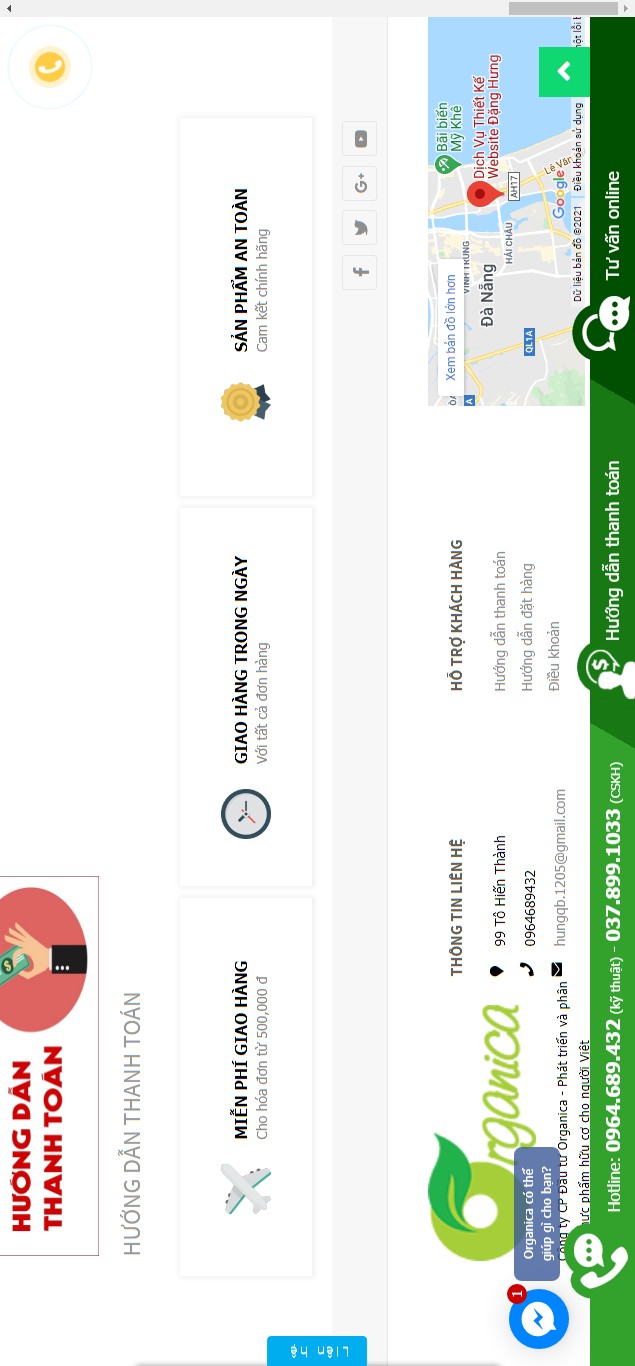
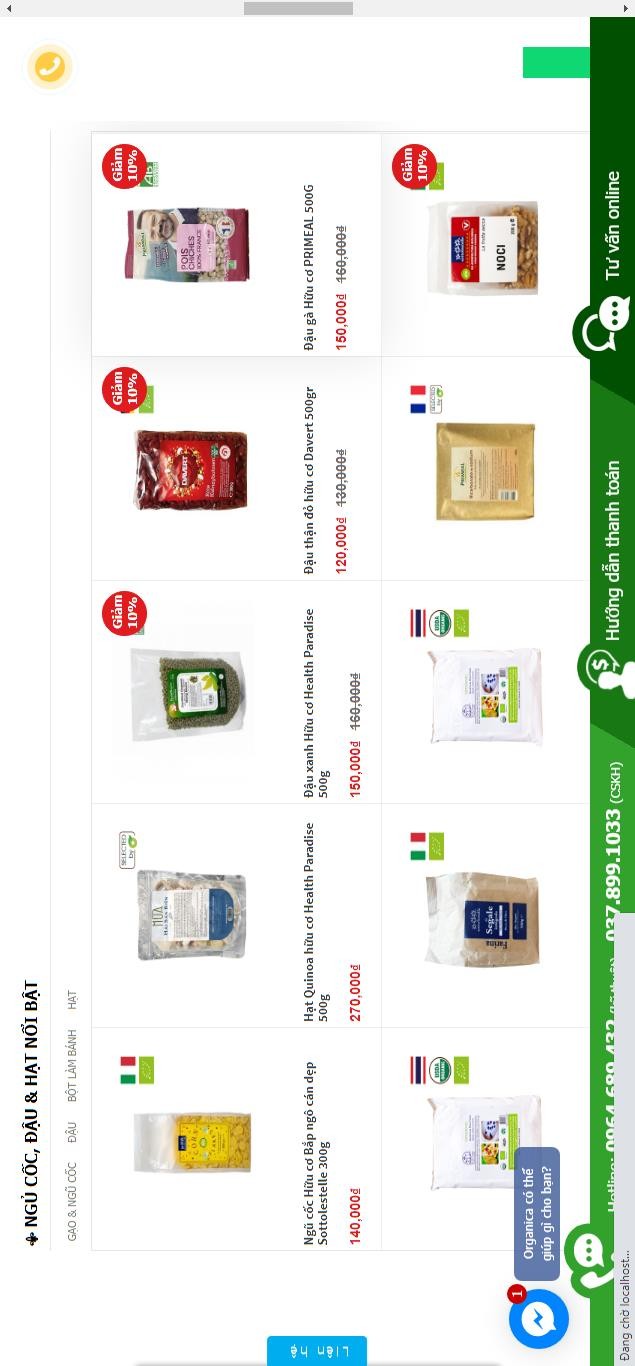
Từ trang chủ khách hàng có thể tìm thấy mọi thông tin cần thiết:

Thông tin về các loại sản phẩm với giao diện gần gũi, thân thiện và dễ sử dụng... từ đó khách hàng có thể đặt mua cho mình những sản phẩm yêu thích, phù hợp với túi tiền...

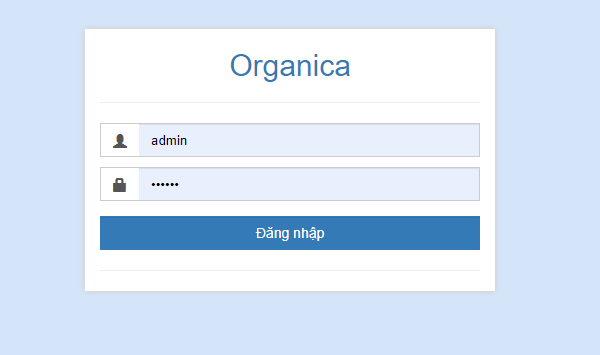
Các bài viết hay về thực phẩm hữu cơ được quản trị hệ thống chọn lọc...

Bạn có thắc mắc, góp ý với công ty…



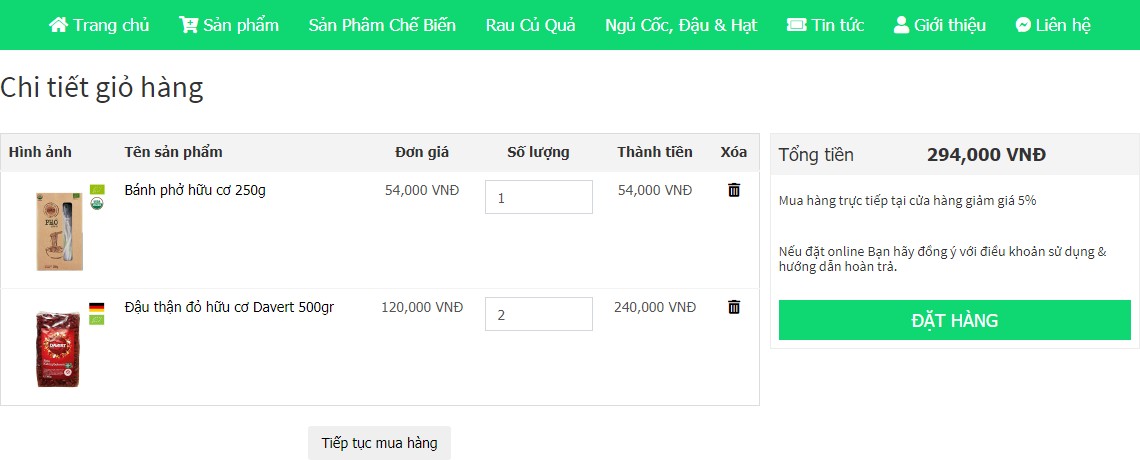
### Giao diện Form đăng nhập



**Hình 4.2: Giao diện form đăng nhập**

Trang “login.php” là trang mà khách hàng sẽ nhập User và Pass của mình để đăng nhập vào website, nếu chưa có tài khoản khách hàng nhấn vào nút Đăng ký để tạo cho mình một tài khoản mới.

### Giao diện trang giỏ hàng



**Hình 4.3: Giao diện giỏ hàng**

Trang *“gio-hang”* là trang mà khách hàng có thể xem được những sản phẩm của mình đã chọn. Tại đây khách hàng có thể biết được số lượng mặt hàng mình đã chọn cũng như tổng số tiền phải thanh toán. Khách hàng có thể thêm, cập nhập cũng như xóa sản phẩm ngay tại giỏ hàng.

### Giao diện Form đăng ký thành viên



**Hình 4.4: Giao diện form đăng ký thành viên**

Trang “*dang-ky.php*” Tại đây khách hàng sẽ điền đầy đủ thông tin cá nhân và đăng ký cho mình một tài khoản, để dùng cho việc mua hàng tại website.

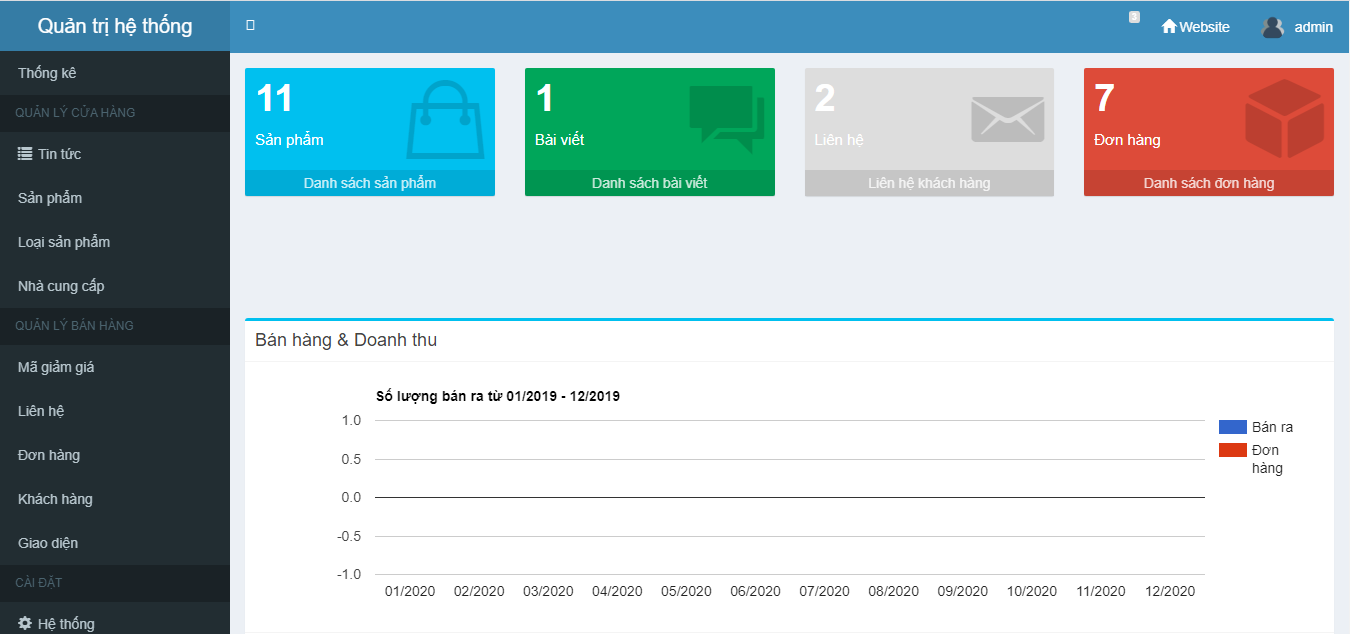
* 1. **Giao diện trang chi tiết sản phẩm**



**Hình 4.5: Giao diện trang chi tiết sản phẩm**

Trang “*chitiet.php*” là trang hiển thị thông tin chi tiết về sản phẩm: giá, bảo hành, thông số kỹ thuật…giúp cho khách hàng có cái nhìn tổng quan về sản phẩm. Là cơ sở để khách hàng có quyết định mua sản phẩm hay không.

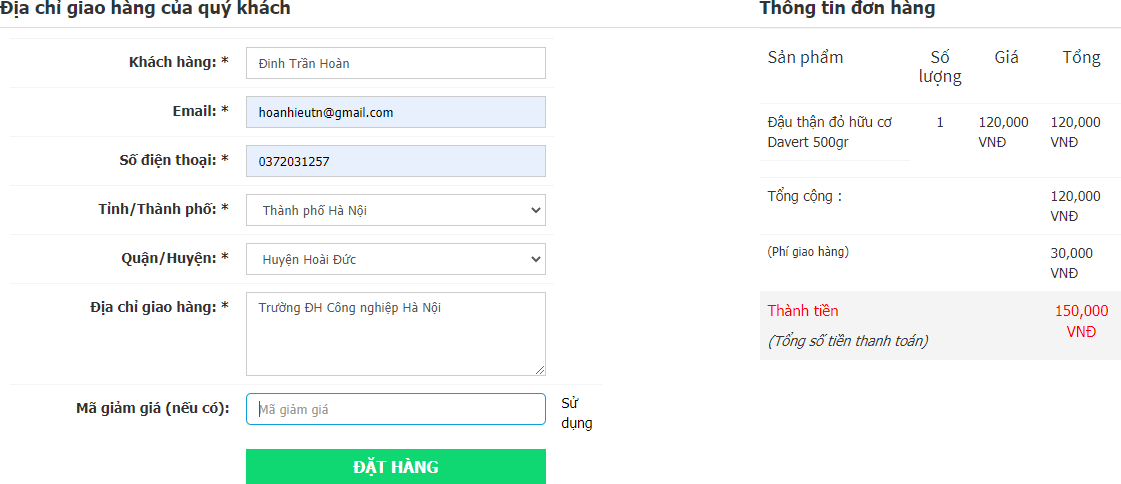
* 1. **Giao diện trang quản trị Admin**



**Hình 4.6: Giao diện trang quản trị Admin**

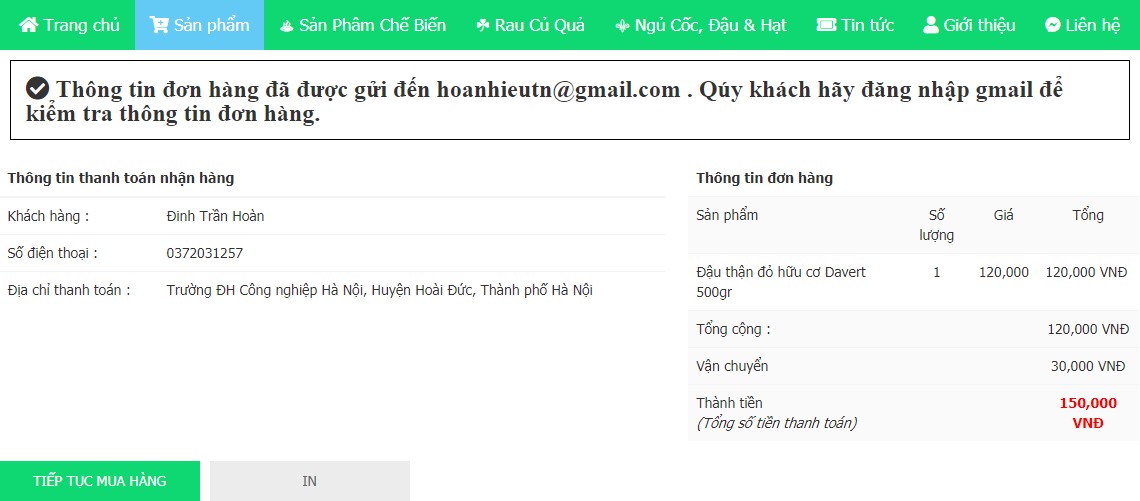
Đây là trang quản trị dành cho người Admin của cửa hàng, cung cấp đầy đủ tính năng cho người quản trị như: quản lý sản phẩm, quản lý thành viên, quản lý các đơn đặt hàng…

* 1. **Giao diện trang đặt hàng**



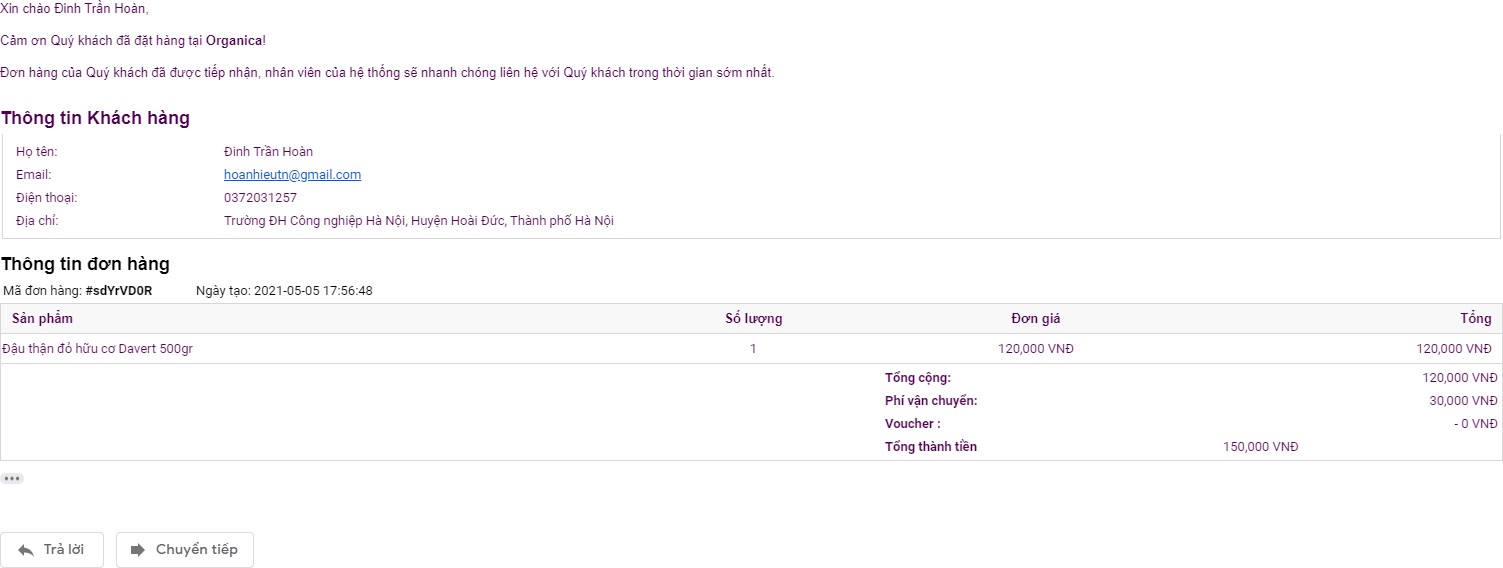
**Hình 4.7: Giao diện trang đặt hàng**

* 1. **Giao diện trang đặt hàng thành công**



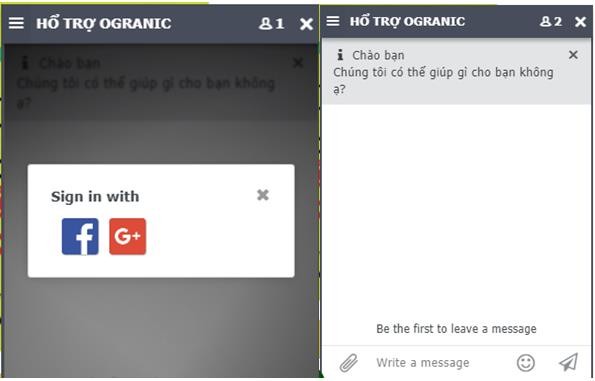
**Hình 4.8: Giao diện trang đặt hàng thành công**

* 1. **Hệ thống sẽ gửi mail thông báo đơn hàng về cho khách hàng**



**Hình 4.9: Kết quả trả về**

* 1. **Hỗ trợ online trực tuyến trên Website**



**Hình 4.10: Hỗ trợ online trực tuyến**

# TÓM TẮT CHƯƠNG 4

Chương 4 đã trình bày kết quả sau nghiên cứu và thực nghiệm chương trình của trang web cụ thể với các giao diện trang chủ, giao diện form đăng nhập, giao diện trang giỏ hàng, giao diện form đăng kí thành viên, giao diện trang chi tiết sản phẩm, giao diện trang quản trị Admin, giao diện trang đặt hàng và hệ thống gửi mail thông báo khi khách hàng đặt hàng thành công, cuối cùng là hỗ trợ online trực tuyến trên chính Website.

# KẾT LUẬN

Với mức độ phức tạp và quy mô ứng dụng, cộng thêm vấn đề thời gian cho nên đề tài: “**Lập trình backend trong thiết kế website bán hàng online cho siêu thị**” chỉ dừng ở mức tìm hiểu ngôn ngữ lập trình Web PHP & MYSQL, và áp dụng xây dựng ứng dụng thực nghiệm trang Web bán hàng trên mạng.

Website với mục đích cung cấp cho khách hàng các thông tin chính xác về các loại **NGŨ CỐC, ĐẬU HẠT, THUỶ HẢI SẢN** đang có trên thị trường của các hãng có uy tín và cách thức đặt mua hàng qua mạng. Các thông tin được cập nhật thường xuyên và nhanh chóng. Vì vậy, rút ngắn được khoảng cách giữa người mua và người bán, đưa thông tin về các sản phẩm mới nhanh chóng đến cho khách hàng.

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

Để hoàn thành đề tài này, một lần nữa em xin chân thành cảm ơn cô **Phan Thị Thu Hằng** là người đã quan tâm, giúp đỡ em trong suốt thời gian qua.

### Em xin chân thành cảm ơn!

# TÀI LIỆU THAM KHẢO

1. W3school.com – *Website cổng thông tin công nghệ web.*
2. Nguyễn Hồng Phương (chủ biên) và các tác giả, *Cơ sở dữ liệu, Đại học Bách khoa Hà Nội, 2016.*
3. Vũ Đức Vượng (chủ biên) và các tác giả, *Kỹ thuật lập trình, Đại học Bách khoa Hà Nội, 2019.*
4. Nguyễn Khanh Văn (chủ biên) và các tác giả, *Lập trình hướng đối tượng, Đại học Bách khoa Hà Nội, 2019.*
5. Lương Ánh Hoàng (chủ biên) và các tác giả, *Lập trình mạng, Đại học Bách khoa Hà Nội, 2019.*
6. Công ty TNHH phần mềm Sắc màu Hà Nội, *Dự án website hangnhatso1, Tài liệu lưu hành nội bộ.*

<div class="col-lg-3 col-xs-6">

<!-- small box -->

<div class="small-box bg-green">

<div class="inner">

<h3><?php echo $total2; ?></h3>

<p>Bài viết</p>

</div>

<div class="icon">

<i class="ion ion-android-chat "></i>

</div>

<a href="<?php echo base\_url() ?>admin/content" class="small-box- footer">Danh sách bài viết</a>

</div>

</div>

<!-- ./col -->

<div class="col-lg-3 col-xs-6">

<!-- small box -->

<div class="small-box bg-yellow">

<div class="inner">

<h3><?php echo $total3; ?></h3>

<p>Liên hệ</p>

</div>

<div class="icon">

<i class="ion ion-email"></i>

</div>

<a href="<?php echo base\_url() ?>admin/customer" class="small-box- footer">Liên hệ khách hàng</a>

</div>

</div>

<!-- ./col -->

<div class="col-lg-3 col-xs-6">

<!-- small box -->

<div class="small-box bg-red">

<div class="inner">

<h3><?php echo $total4; ?></h3>

<p>Đơn hàng</p>

</div>

<div class="icon">

<i class="ion ion-cube"></i>

</div>

<a href="<?php echo base\_url() ?>admin/orders" class="small-box- footer">Danh sách đơn hàng</a>

</div>

</div>

<!-- ./col -->

</div>

<!-- /.row -->

</section>

<section class="content">

<div class="row">

<!-- /.col (LEFT) -->

<div class="col-md-12">

<!-- LINE CHART -->

<div class="box box-info">

<div class="box-header with-border">

<h3 class="box-title">Bán hàng & Doanh thu</h3>

<div class="box-tools pull-right">

<button type="button" class="btn btn-box-tool" data- widget="collapse"><i class="fa fa-minus"></i>

</button>

<button type="button" class="btn btn-box-tool" data- widget="remove"><i class="fa fa-times"></i></button>

</div>

</div>

<div class="box-body">

<div class="chart">

<div id="chart\_div" style="width: 100%; height: 250px;"></div>

</div>

</div>

<div class="box-footer">

<div class="row">

<div class="col-sm-4 col-xs-6">

<div class="description-block border-right">

<h5 class="description-

header" style="color: #e90000;"><?php echo number\_format($total);?> VNĐ</h5>

<span class="description-text">Tổng doanh thu</span>

</div>

<!-- /.description-block -->

</div>

<!-- /.col -->

</div>

<?php

$d=getdate();

$year=$d['year'];

for ($i=1; $i <= 12 ; $i++)

{

$list\_orrders = $this->Morders->order\_follow\_month($year, $i);

$total\_month = 0;

foreach ($list\_orrders as $row\_orrder)

{

$total\_month += $row\_orrder['money'];

}

echo '<div class="col-sm-4 col-xs-6">

<div class="description-block border- right" style="display: inline-flex;">

<span class="description- text">Doanh thu tháng '.$i.' : </span>

<h5 class="description- header" style="color: #e90000;padding-

left: 10px;">'.number\_format($total\_month).' VNĐ</h5>

</div>

<!-- /.description-block -->

</div>';

}

?>

<!-- /.row -->

</div>

<!-- /.box-body -->

</div>

</div>

</section>

<!-- /.content -->

</div>

<!-- /.content-wrapper -->

<script>

google.charts.load('current', {'packages':['corechart']}); google.charts.setOnLoadCallback(drawVisualization);

function drawVisualization() {

var data = google.visualization.arrayToDataTable([ ['Month', 'Bán ra', 'Đơn hàng'],

<?php

$d=getdate();

$year=$d['year'];

for ($i=1; $i <= 12 ; $i++)

{

$list\_orrders = $this->Morders->order\_follow\_month($year, $i);

$sum = 0;

foreach ($list\_orrders as $row\_orrder)

{

$order\_detail = $this->Morderdetail-

>orderdetail\_orderid($row\_orrder['id']); foreach ($order\_detail as $value) {

$sum += $value['count'];

}

}

if($i >= 1 && $i <=9)

{

echo "['0".$i.'/'.$year."',".$sum.",".count($list\_orrders)."],";

}

else

{

echo "['".$i.'/'.$year."',".$sum.",".count($list\_orrders)."],";

}

}

?>

]);

var options = {

title: 'Số lượng bán ra từ 01/2019 - 12/2019', seriesType: 'bars'

};

var chart = new google.visualization.ComboChart(document.getElementById( 'chart\_div'));

chart.draw(data, options);

}

</script>