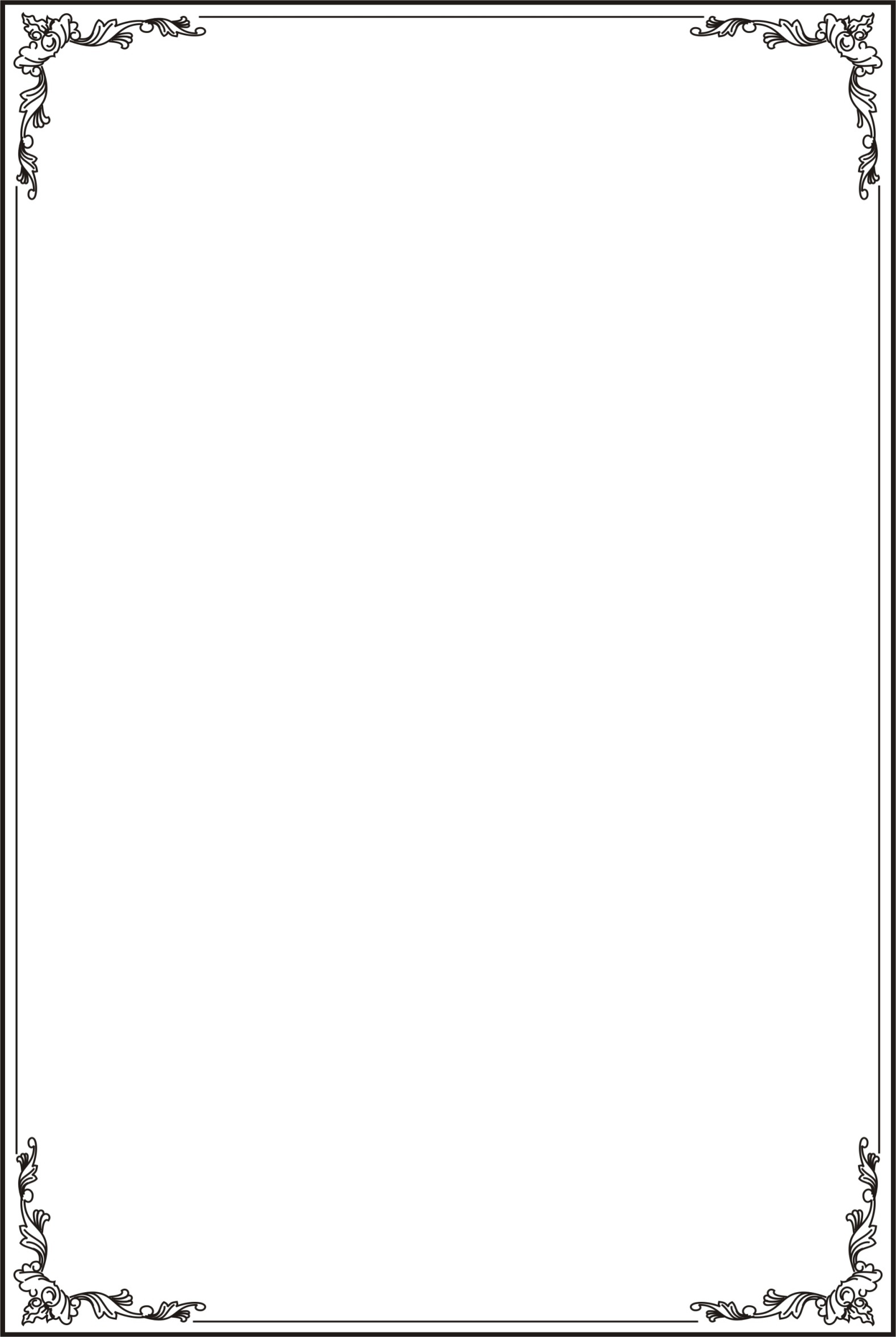
******TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

----🙞🙜🕮🙞🙜----

****

**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**Đề tài: Xây dựng Website bán đồ gỗ cho cửa hàng Điệp Hoa bằng PHP và MySQL**

|  |  |
| --- | --- |
| **Giáo viên hướng dẫn :** | **ThS. Phạm Thế Anh** |
|  |  |
| **Sinh viên thực hiện :** | **Nguyễn Mạnh Hiếu** |
| **Lớp :** | **ĐH - CNTT5 - K14** |
| **Mã sinh viên :** | **2019605147** |

**Hà Nội, 2023**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

----🙞🙜🕮🙞🙜----

****

**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**Đề tài: Xây dựng Website bán đồ gỗ cho cửa hàng Điệp Hoa bằng PHP và MySQL**

|  |  |
| --- | --- |
| **Giáo viên hướng dẫn :** | **ThS. Phạm Thế Anh** |
|  |  |
| **Sinh viên thực hiện :** | **Nguyễn Mạnh Hiếu** |
| **Lớp :** | **ĐH - CNTT5 - K14** |
| **Mã sinh viên :** | **2019605147** |

[Lời mở đầu 5](#_Toc130737499)

[1. Giới thiệu đề tài : 6](#_Toc130737500)

[2. Yêu cầu của đề tài : 6](#_Toc130737501)

[3. Khảo sát hiện trạng thực tế cửa hàng: 6](#_Toc130737502)

[4. Khảo sát sơ bộ các yêu cầu của khách hàng 7](#_Toc130737503)

[5. Công cụ phục vụ đề tài : 9](#_Toc130737504)

[6. Nội dung báo cáo 9](#_Toc130737505)

[PHẦN 2 :NỘI DUNG 10](#_Toc130737506)

[CHƯƠNG 1 : TỔNG QUAN VỀ CƠ SỞ LÝ THUYẾT 10](#_Toc130737507)

[1. Đôi nét về thương mại điện tử. 10](#_Toc130737508)

[**1.1.** **Thương mại điện tử là gì ?** 10](#_Toc130737509)

[**1.2.** **Lợi ích của thương mại điện tử.** 10](#_Toc130737510)

[2. Giới thiệu về PHP. 11](#_Toc130737511)

[**2.1.** **Lịch sử phát triển.** 11](#_Toc130737512)

[**2.2.** **Cấu trúc cơ bản.** 14](#_Toc130737513)

[**2.3.** **Xuất giá trị trên trình duyệt.** 14](#_Toc130737514)

[**2.4.** **Biến, hằng, chuỗi, và các kiểu dữ liệu.** 14](#_Toc130737515)

[**2.4.1.** **Biến** 14](#_Toc130737516)

[**2.4.2.** **Hằng** 15](#_Toc130737517)

[**2.4.3.** **Chuỗi** 15](#_Toc130737518)

[**2.4.4.** **Kiểu dữ liệu** 15](#_Toc130737519)

[**2.5.** **Các phương thức được sử dụng trong lập trình PHP.** 15](#_Toc130737520)

[**2.6.** **Cookie và Session trong PHP.** 16](#_Toc130737521)

[**2.7.** **Hàm** 17](#_Toc130737522)

[3. Giới thiệu về MySQL. 18](#_Toc130737523)

[**3.1.** **Tông quan về MySQL.** 18](#_Toc130737524)

[**3.2.** **Khởi động và sử dụng.** 18](#_Toc130737525)

[**3.3.** **Một số thuật ngữ.** 19](#_Toc130737526)

[**3.4.** **Những cú pháp cơ bản.** 19](#_Toc130737527)

[4. Giới thiệu về UML. 20](#_Toc130737528)

[**4.1.** **UML là gì ?** 20](#_Toc130737529)

[**4.2.** **Mục tiêu của UML.** 20](#_Toc130737530)

[**4.3.** **Các biểu đồ trong UML.** 21](#_Toc130737531)

[CHƯƠNG 2 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG 23](#_Toc130737532)

[Tài liệu tham khảo 23](#_Toc130737533)

# Lời mở đầu

Ngày nay, dưới sự ảnh hưởng của sự bùng lên phát triển mạnh mẽ của công nghệ thông tin và Internet tốc độ cao, để không bị lạc hậu về cách thức quảng cáo, hình thức kinh doanh và dần đi tới nguy cơ thua lỗ, không có sức cạnh tranh trên thị trường kinh tế càng ngày càng hiện đại thì các công ty, tổ chức kinh doanh buộc phải bắt kịp xu hướng kinh doanh hiện tại. Việc ứng dụng công nghệ thông tin và tin học hóa được coi là yếu tố bắt buộc quyết định để tạo ra những bước đột phá mạnh mẽ.

Với xu thế toàn cầu hóa nền kinh tế thế giới ngày càng mở hơn đồng nghĩa với đó là đời sống xã hội ngày càng được nâng cao hơn, đặc biệt là nhu cầu trao đổi hàng hóa của con người ngày càng tăng về cả số lượng lẫn chất lượng. Các cửa hàng, tổ chức bán buôn bán lẻ trên đang ra sức không ngừng đầu tư và cải thiện các phương thức kinh doanh cùng sản phẩm của họ nhằm tiến đến mục đích triển khai thương mại hóa trên Internet và đã đạt được nhiều lợi ích. Thông qua các sản phẩm và công nghệ này chúng ta dễ ràng nhận ra tầm quan trọng và tinh tất yếu của các thương mại điện tử.

Để tiếp cận, bắt kịp xu hướng và góp một phần vào đẩy mạnh sự phát triển thương mại điện tử trong thời kì hiện nay, em đã tìm hiểu, xây dựng và cài đặt website bán đồ gỗ cho cửa hàng “ Đồ gỗ mỹ nghệ Điệp Hoa” với các mặt hàng bày bán là các sản phẩm thủ công được chế tác tinh xảo.

Để hoàn thiện đề tài này, em xin được gửi lời cảm ơn chân thành đến giáo viên hướng dẫn của em là thầy giáo **Ths.** **Phạm Thế Anh,** giảng viên Khoa Công nghệ Thông tin đã tận tình giúp đỡ và hướng dẫn để em hoàn thiện đề tài này. Trong quá trình thực hiện đề tài em đã nỗ lực cố cắng để hoàn thiện đề tài nhưng vẫn rất mong có sự nhận xét đóng góp của thầy để để tài hoàn thiện hơn nữa.

Em xin chân thành cảm ơn !

**Sinh viên thực hiện**

Nguyễn Manh Hiếu

**PHẦN 1 : KHẢO SÁT ĐỀ TÀI**

1. **Giới thiệu đề tài**

Tên đề tài : ***Thiết kế website bán đồ gỗ cho cửa hàng Điệp Hoa bằng PHP và MySQL.***

Tính cấp thiết của đề tài : Trong công cuộc cách mạng số hóa 4.0 ngày càng đẩy mạnh trên mọi lĩnh vực, để bắt kịp với xu thế thương mại hóa toàn cầu, các công ty tổ chức kinh doanh trên thị trường buộc phải mở rộng phương thức kinh doanh theo kiểu truyền thống lẫn trực truyến mới đủ sức cạnh tranh với các thương hiệu khác, đặc biệt đối với dòng sản phẩm đồ gỗ đồ thủ công mỹ nghệ do các nghệ nhân làm ra tuy rất tinh sảo nhưng lại quen với hình thức buôn bán truyền thống nên rất ít được biết đến.Vậy nên, một trang web bày bán sản phẩm đồ gỗ thủ công mỹ nghệ là điều cần thiết và chắc chắn sẽ đem lại hiệu quả cao về mặt quảng cáo và doanh số bán hàng.

1. **Yêu cầu của đề tài**

Yêu cầu kiến thức : cần phải nắm rõ kiến thức về ngôn ngữ PHP, HTML-CSS và các cách thức sử dụng các công cụ để phục vụ đề tài.

Yêu cầu sản phẩm : sản phẩm đầu ra sau khi kết thúc quá trình thực hiện đề tài phải hoàn thiện đáp ứng đúng và đủ các yêu cầu về các trang giao diện và các chức năng mà khách hàng đã yêu cầu, đồng thời cần phải dễ sử dụng, bố cục trang và màu sắc cần phải hài hòa và hợp lý.

1. **Khảo sát hiện trạng thực tế cửa hàng:**

Đề tài được triển khai dựa triên yêu cầu đặt hàng bởi chủ cửa hàng “Đồ gỗ mỹ nghệ Điệp Hoa” là ông Nguyễn Văn Điệp.

Địa chỉ cửa hàng nằm tại xã Bắc Sơn, huyện Sóc Sơn, thành phố Hà Nội và liên hệ đường dây nóng là 0976097342.

Loại sản phẩm kinh doanh bao gồm các sản phẩm đồ gỗ trạm khắc, đồ mỹ nghệ để trang trí không gian nhà cửa, đồ mây tre đan.

Website được thiết kế và cài đặt sử dụng để bày bán và quảng cáo sản phẩm cho cửa hàng đồ gỗ có tên là “Đồ gỗ mĩ nghệ Điệp Hoa ” hiện vẫn còn đang kinh doanh theo kiểu truyền thống. Vốn nhờ có sản phẩm đồ gỗ chất lượng và đa dạng cũng như danh tiếng kinh doanh trong nghề tới 8 năm mà cửa hàng đã có được uy tín và lượng khách đông đảo gần xa. Để có thể tăng doanh số bán hàng cũng như phục vụ và tư vấn cho khách hàng tốt hơn, chính xác hơn, nhanh chóng hơn,và quảng bá hình ảnh “Cửa hàng đồ gỗ Điệp Hoa” rộng hơn nữa thì chủ cửa hàng muốn từng bước tin học hóa các khâu quản lý. Đặc biệt là trong công tác kế toán, quản lý nguồn hàng và chăm sóc khách hàng. Bởi các công tác thủ công hiện tại đang được nhân viên thực hiện bằng ghi chép sổ sách đang bộc lộ rất nhiều hạn chế và không hề tỏ ra hiệu quả như sau cần phải giải quyết :

* Thứ nhất, công tác tra cứu thông tin về hàng hóa có trong kho và trong cửa hàng , các bên guồn cung cũng như khách hàng mất rất nhiều thời gian và có những trường hợp thông tin không chính xác.
* Thứ hai, việc lưu trữ thông tin về hàng hóa nhập và và bán ra của cửa hàng , các khoản thu chi phí bảo dưỡng hàng hóa sản phẩm cần nhân viên ghi chép và nhiều loại giấy tờ cồng kềnh và không đạt hiệu quả như mong muốn của chủ cửa hàng.
* Thứ ba, cập nhật các thông tin hằng ngày tốn nhiều thời gian và khó khăn trong việc thực hiện báo cáo thống kê, nhất là khi có việc đột xuất.

Trước tình hình đó vấn đề sẽ càng trở nên khó khăn hơn và ảnh hưởng đến hiện trạng kinh doanh của cửa hàng nếu cứ kéo dài phương pháp theo kiểu truyền thống. Cách thức giải quyết vấn đề này là sử dụng một website bán hàng đáp án được các yêu cầu cơ bản sau :

* Giảm khối lượng ghi chép nhằm lưu trữ thông tin hiệu quả hơn nhờ sử dụng máy tính.
* Cập nhật dữ liệu nhanh chóng, chính xác và kịp thời.
* Thống kê được lượng sản phẩm hàng hóa nhập và bán ra, thu chi và tình hình doanh thu của cửa hàng.
* Có thể tra cứu hoặc in các hóa đơn nếu cần như: phiếu nhập hàng, hóa đơn mua, phiếu thu chi.
* Có khả năng lưu trữ thông tin lâu dài tiện lợi, đảm bảo truy vấn nhanh khi cần thiết.

1. **Khảo sát sơ bộ các yêu cầu của khách hàng**
2. Chức năng cơ bản :

Chủ cửa hàng yêu cầu một website bán và giới thiệu sản phẩm của cửa hàng đến người tiêu dùng với danh sách thông tin chi tiết về các mặt hàng được bày bán với giá cả chính xác. Nó có các chức năng cơ bản như sau :

* Cho phép người quản trị cập nhật các thông tin về cửa hàng và sản phẩm của cửa hàng lên website.
* Hiển thi danh sách các mặt hàng theo từng loại từng danh mục riêng biệt.
* Hiển thị thông tin chi tiết về một sản phẩm nào đó và thông tin về sản phẩm khách hàng đã chọn mua trong đơn hàng.
* Hiển thị thông tin khách hàng.
* Thêm mới cũng như hủy bán một mặt hàng, một loại hàng nào đó, tin khuyến mãi , hàng mới về , …
* Xử lý đơn hàng (chấp nhận hoặc hủy bỏ sau khi giao hàng thành công).

1. Các yêu cầu về mặt thiết kế :

Hệ thống website được chia thành hai phần như sau :

**Phần thứ nhất** : Phần giao diện và chức năng cho khách hàng. Khách hàng là những người có nhu cầu mua sắm hàng hóa, họ sẽ tìm kiếm các mặt hàng cần thiết từ hệ thống và đặt mua các mặt hàng này. Vì thế cần phải có các chức năng sau :

* Hiển thị danh sách các mặt hàng của cửa hàng để khách hàng dễ dàng xem, lựa chọn và mua hàng.
* Khách hàng xem các sản phẩm mới, sản phẩm khuyến mãi trên website nếu có.
* Sau khi khách hàng chọn và đặt hàng dựa theo thủ tục trực tuyến thì phải hiện lên đơn hàng để khách hàng có thể nhập thông tin mua hàng và xem hóa đơn mua hàng.

**Phần thứ hai** : Phần giao diện và chức năng cho admin- người quản trị. Người quản trị này có quyền kiểm soát toàn bộ hệ thống. Người này được cấp tên tài khoản(username) và mật khẩu để có thể đăng nhập hệ thông và thực hiện chức năng của mình như sau :

* Chức năng cập nhật , thêm mới, chỉnh sửa thông tin, xóa thông tin của các loại hàng, các mặt hàng, các danh mục thương hiêu đòi hỏi phải cực kì chính xác.
* Tiếp nhận kiểm tra đơn đặt hàng của khách hàng có thể chấp nhận đơn hàng và hủy đơn hàng đã được giao. Hiển thị đơn đặt hàng

1. **Công cụ phục vụ đề tài :**

Thiết bị và phần mềm cần có để sử dụng trong để tài bao gồm :

* Máy tính có thể thiết kế web
* Các công cụ hỗ trợ thiết kế web như các phần mềm Subline Text ,Visual Studio Code
* Hệ quản trị cơ sở dữ liệu Microsoft SQL Sever
* Cài đặt Xampp,MySQL, Apache.
* Phần mềm Adobe Photoshop

1. **Nội dung báo cáo**

Từ các khảo sát chi tiết đã được đề cập đến ở bên trên ta có được những tóm tắt phần nội dung báo cáo như sau. Đầu tiên cần đề cập đến các vấn đề lý thuyết về thương mại điện tử, lợi ích của nó đối với kinh doanh buôn bán cho các cửa hàng các tổ chức.Thứ hai, đó là các kiến thức về ngôn ngữ được sử dụng để triển khai đề tài là PHP, hệ quản trị và các kiến thức liên quan khác.Thứ ba, cần phải có một cái nhìn chi tiết về hệ thống website mà ta sắp thiết kế nhờ vào công đoạn phân tích - thiết kế hệ thống( lập ra các biểu đồ use case, biểu đồ lớp ….). Và cuối cùng là phần các cài đặt chương trình cho ta thấy được các trang giao diện của website kèm với đó là kiểm thử các chức năng của hệ thống xem có hoạt động mượt mà hay chưa.

**PHẦN 2 :NỘI DUNG**

**CHƯƠNG 1 : TỔNG QUAN VỀ CƠ SỞ LÝ THUYẾT**

1. **Đôi nét về thương mại điện tử.**
   1. **Thương mại điện tử là gì ?**

Thương mại điện tử là quá trình mua bán hàng hóa và dịch vụ thông qua các phương tiện điện tử và mạng viễn thông đặc biệt là qua máy tính và mạng Internet.

Ngày nay, người ta hiểu khái niệm thương mại điện tử thông thường là tất cả các phương pháp kinh doanh và các quy trình quản trị thông qua các kênh điện tử các sàn buôn bán điện tử mà trong đó Internet hay ít nhất là các kĩ thuật và giao thức được sử dụng trong Internet đóng vai trò cơ bản và công nghệ thông tin được coi là điều kiện tiên quyết. Một mặt quan trọng khác đó là không cần phải liên tục sử dụng phương tiện truyền thông quảng cáo, một đặc trưng bắt buộc phải làm khi kinh doanh truyền thống. Thêm vào đó là tác động của con người vào quy trình kinh doanh được giảm đến mức tối thiểu. Để làm được điều đó đòi hỏi cần có một hệ thống tích hợp rông lớn các tính năng phục vụ kinh doanh.

* 1. **Lợi ích của thương mại điện tử.**

Lợi ích lớn nhất mà thương mại điện tử đem lại đó là tiết kiệm được chi phí lớn tạo thuận lợi cho các bên giao dịch. Giao dịch bằng thương mại điện tử nhanh hơn là giao dịch theo các cách thức truyền thống. Các giao dịch qua Internet có chi phí rất rẻ thậm chí không mất phí giao dịch, một tổ chức hay doanh nghiệp có thể gửi các quảng cáo chào hàng đến hàng loạt các khách hàng một cách nhanh chóng mà chi phí chỉ bẳng gửi cho một khách hàng. Với thương mại điện tử các bên có thể tiến hành giao dịch khi ở cách xa nhau, loại bỏ được các khó khăn gặp phải về khoảng cách địa lý. Điều này càng giúp các doanh nghiệp tiết kiệm được các chi phín đi lại, thời gian gặp mặt trong khi mua bán. Với người tiêu dùng thì họ có thể chỉ cần ngồi ở nhà với chiếc điện thoại hay máy tính là có thể thoải mái chọn lựa và đặt mua nhiều loại hàng hóa , dịch vụ thật nhanh chóng.

Những lợi ích như trên chỉ có được với những tổ chức hay cá nhân là chủ cửa hàng thực sự nhận thức được tầm quan trọng và giá trị của thương mại điện tử. Vì vậy thương mại điện tử góp phần thúc đẩy sự cạnh tranh giữa các doanh nghiệp, các cửa hàng nhưng cũng đem lại đó là cơ hội phát triển và lợi ích kinh tế vô cùng to lớn. Và càng đặc biệt hơn trong bối cảnh nước ta đang hội nhập kinh tế quốc tế ngày càng sâu và rộng hơn đem lại cơ hội cho các doanh nghiệp trong nước được vươn tầm thế giới cạnh tranh bình đẳng với các doanh nghiệp nước ngoài.

1. **Giới thiệu về PHP.**
   1. **Lịch sử phát triển.**

**PHP/FI**

PHP được phát triển từ một sản phẩm có tên là PHP/FI.PHP/FI được Rasmus Lefdorf tạo ra năm 1995, ban đầu được xem như là một tập con đơn giản của các mã kịch bản Perl để theo dõi tình hình truy cập đến bản sơ yếu lý lịch của ông trên mạng. Ông đã đặt tên cho bộ mã kịch bản này là “Personal Home Page Tools “.

**PHP 3**

PHP 3 là phiên bản đầu tiên cho chúng ta thấy hình ảnh gần gũi với các phiên bản PHP mà chúng ta biết ngày nay. Nó đã được Andi GutMans và Zeev Surasky tạo ra năm 1997 sau khi viết lại toàn bộ mã nguồn trước đó. Lí do chính họ tạo ra phiên bản này là do họ nhận thấy PHP/FI hết sức yếu kém trong việc phát triển các ứng dụng thương mại điện tử mà họ đang xúc tiến trong một dự án của trường đại học. Trong nỗ lực hợp tác và bắt đầu xây dựng dựa trên cơ sở người dùng đã có của PHP/FI, Andi, Rasmus và Zeev đã quyết định hợp tác và công bố PHP 3.0 như là phiên bản thế hệ kế tiếp của PHP/FI 2.0 và chấm dứt phát triển PHP/FI 2.0.

Ngôn ngữ hoàn toàn mới được công bố dưới cái tên mới, xóa bỏ mối liên hệ với việc sử dụng vào mục đích cá nhân hạn hẹp mà cái tên PHP/FI gợi nhắc. Nó được đặt tên ngắn gọn là “PHP”, một kiểu viết tắt hồi quy của “PHP : Hypertext Preprocessor “. PHP 3.0 được chính thức công bố vào tháng 6 năm 1988 sau 9 tháng được cộng đồng kiểm nghiệm.

**PHP 4**

Vào mùa đông năm 1988, ngay sau khi PHP 3.0 được công bố, Andi và Zeep đã bắt đầu bắt tay vào viết lại phần lõi của PHP. Một engine mới, có tên “Zend Engine” (ghép các chữ đầu trong tên của Zeep và Andi ) đã đáp ứng được các nhu cầu thiết kế này một cách thành công, và lần đầu tiên được giới thiệu vào giữa năm 1999. PHP 4.0 dựa trên engine này và đi kèm hàng loạt tính năng mới bổ sung, đã chính thức được công bố vào tháng 5 năm 2000, gần 2 năm sau khi bản PHP 3.0 ra đời. Ngoài tốc độ xử lý được cải thiện rất nhiều, PHP 4.0 đem đến các tính năng chủ yếu khác gồm có sự hỗ trợ nhiều máy chủ Web hơn, hỗ trợ phiên làm việc HTTP, tạo bộ đệm thông tin đầu ra, nhiều cách xử lý thông tin người sử dụng nhập vào bảo mật hơn và cung cấp một vài cấu trúc ngôn ngữ mới.

Với PHP 4.0, số nhà phát triển đã lên đến hàng trăm ngìn và hàng triệu Site đã công bố cài đặt PHP, chiếm khoảng 20% số tên miền trên mạng internet.

**PHP 5**

Sự thành công của PHP 4 đã không làm cho nhóm phát triển tự mãn. Cộng đồng php đã giúp cho họ nhận ra những yếu kém của PHP 4.0, đặt biệt với khả năng hỗ trợ lập trình hướng đối tượng (OPP), xử lý XML, không hỗ trợ giao thức máy khách mới của MySQL 4.1 và 5.0, hỗ trợ dịch vụ web yếu. Những điểm này chính là mục đích để Andi và Zeev viết Zend Engine 2.0-lõi của PHP 5.0. Ngày 29 tháng 6 năm 2003, PHP 5 beta 1 đã chính thức công bố để cộng động kiểm nghiệm. Đó cũng là phiên bản đầu tiên của Zend Engine 2.0. Phiên bản beta 2 sau đó đã ra mắt vào tháng 10 năm 2003 với sự xuất hiện của 2 tính năng rất được chờ đợi : Iterators, Reflection nhưng NameSpace – một tính năng gây tranh cãi đã bị loại khỏi mã nguồn. Ngày 21 tháng 12 năm 2003, PHP 5 beta 3 đã được công bố để kiểm tra với việc phân phối kèm với tidy, bỏ hỗ trợ Windows 95, khả năng gọi các hàm PHP bên trong XSLT, sửa chữa nhiều lỗi vào thêm khá nhiều hàm mới. PHP 5 bản chính thức đã ra mắt ngày 13 tháng 7 năm 2004 sau một chuỗi khá dài các bản beta. Mặc dù coi đây là phiên bản chính thức đầu tiên nhưng PHP 5 vẫn còn một số lỗi trong đó đáng kể là lỗi xác thực HTTP.

Ngày 14 tháng 7 năm 2005, PHP 5.1 beta 3 được PHP team công bố đánh dấu sự chín muồi mới của PHP với sự có mặt của PDO, một nỗ lực trong việc tạo ra một hệ thống API nhất quán trong việc truy cập cơ sở dữ liệu và thực hiện các câu truy vấn. Ngoài ra, trong PHP 5.1 các nhà phát triển tiếp tục có những cải tiến trong nhân Zend Engine 2, nâng cấp module PCRE lên bản PCRE 5.0 cùng với những tính năng và cải tiến mới trong SOAP, Streams và SP.

**PHP 6**

Hiện nay, phiên bản tiếp theo đang được phát triển. Phiên bản PHP 6 được kì vọng sẽ lấp đầy những khiếm khuyết của PHP ở bản hiện tại.

Rất nhiều nhà phát triển ứng dụng và quản lý dự án có quan điểm rằng PHP vẫn chưa sẵn sàng cho cấp doanh nghiệp (enterprise), và trên thực tế, PHP vẫn chưa thâm nhập sâu được vào thị trường này. Chính vì thế, Zend đã tiến hành chuẩn hóa PHP, tạo được sự tin cậy hơn cho giới người dùng cao cấp.

Zend Platform là một bộ sản phẩm giúp quản lý ứng dụng PHP, nâng cao hiệu suất, tăng tốc độ của ứng dụng PHP.

Zend Framework là tập hợp các lớp, các thư viện lập trình viết bằng PHP (PHP 5) nhằm cung cấp một giao diện lập trình chuẩn cho các nhà phát triển ứng dụng.

Ngoài ra, một số Framework khác cũng được phát triển nhằm hỗ trợ lập trình PHP ở cấp doanh nghiệp, trong đó đáng chú ý có thể kể đến CodeIgniter, CakePHP, Symfony, Seagull…

* 1. **Cấu trúc cơ bản.**

PHP cũng có thẻ bắt đầu và thẻ kết thúc giống như ngôn ngữ HTML.Chỉ khác đối với PHP chúng ta có nhiều cách thể hiện nó.

Cách 1 : Cú pháp chính thường dùng.

<? php *Mã lệnh PHP* ?>

Cách 2 : Cú pháp rút gọn.

<? *Mã lệnh PHP* ?>

Cách 3 : Cú pháp giống với ASP.

<% *Mã lệnh PHP* %>

Cách 4 : Cú pháp bắt đầu bằng script.

<script language=php>…….</script>

Mặc dù có tới 4 cách thể hiện nhưng đối với lập trình viên có kinh nghiệm thì viêc sử dụng cách một vẫn là lựa chọn tối ưu và thương xuyên hơn.

Trong PHP để kết thúc một dòng lệnh ta sử dụng dấu “ ; ”.

Để chú thích một đoạn dữ liệu nào đó ta sử dung cặp dấu “//” cho từng dòng hoặc cặp dấu “/\* ………\*/” cho từng đoạn mã lệnh.

* 1. **Xuất giá trị trên trình duyệt.**

Để xuất dữ liệu ra trình duyệt ta có những dòng cú pháp như sau :

* echo “ ……Thông tin cần đưa ra….”;
* printf “……Thông tin cần đưa ra…”;

Thông tin bao gồm biến, chuỗi hoặc lệnh HTML…..

Nếu giữa hai chuỗi ta muôn liên kết với nhau ta sửu dụng dấu “ . ”.

* 1. **Biến, hằng, chuỗi, và các kiểu dữ liệu.**
     1. **Biến**

Biến được xem là một vùng nhớ dữ liệu tạm thời. Và giá trị có thể thay đổi được bắt đầu bằng kì hiệu “ $ ” và theo sau là một từ hoặc một cụm từ (phải viết liền hoặc nối nhau bằng dấu gạch dưới – lưu ý biến có phân biệt chữ hoa chữ thường và không cho phép trùng với từ khóa có sẵn của PHP).

Trong PHP để sử dụng được 1 biến ta cần phải khai báo nó trước, tuy nhiên đối với đa phần khi sử dụng biến ta có thể vừa khai báo vừa gán giá trị cho biến để tiết kiệm các dòng lệnh.

Bản thân biến cũng có thể gán cho các kiểu dữ liệu khác và tùy theo ý định của người lập trình mong muốn.

* + 1. **Hằng**

Nếu biến là cái có thể thay đổi được thì hằng lại ngược lại là không thay đổi được. Hằng trong PHP được định nghĩa bời hàm define theo cú pháp : define (string tên\_hằng, giá\_trị\_hằng). Hằng được xem là hợp lệ khi thỏa mãn các yêu cầu như : hằng không có dấu “ $ ” như biến, chỉ được gán giá trị duy nhất một lần và dùng được ở bất cứ đoạn mã nào, thường được viết hoa để dễ phân biệt với biến.

* + 1. **Chuỗi**

Chuỗi là một nhóm các kí tự, số, khoảng trắng, dấu ngắt, được đặt trong các cặp dấu nháy kép. Để tạo ra một biến chuỗi ta cần phải gán giá trị cho chuỗi vào một biến hợp lệ. Để liên kết một chuỗi và một biến ta sử dụng dấu “ . ”.

* + 1. **Kiểu dữ liệu**

Các kiểu dữ liệu khác nhau chiếm các lượng bộ nhớ khác nhau và có thể được xử lý theo các cách khác nhau khi chúng được thao tác trong 1 script. PHP hỗ trợ các kiểu dữ liệu sau : String, Integer, Float, Boolean, Array ,Object , Null, Resource.

Chúng ta có thể sử dụng hàm có sẵn GETtype() để kiểm tra kiểu dữ liệu của bất kì biến nào trong đoạn mã.

* 1. **Các phương thức được sử dụng trong lập trình PHP.**

Có hai phương thức được sử dung trong lâp trình PHP đó là GET và POST.

Phương thức GET được dùng để lấy dữ liệu từ form nhập liệu và nhiệm vụ chính của nó là lấy nội dung trang dữ liệu từ webserver.

Phương thức POST cũng được dùng để lấy dữ liệu từ form nhập liệu, tuy nhiên , nhiệm vụ của nó lại là lấy dữ liệu đó chuyển chúng lên trang chủ webserver.

* 1. **Cookie và Session trong PHP.**

Cookie và Session trong PHP là hai phương pháp sử dụng để quản lý các phiên làm việc giữa người sử dụng và hệ thống.

**Cookie** là một đoạn dữ liệu được ghi vào đĩa cứng hoặc bộ nhớ của máy tính. Nó được trình duyệt gửi ngược lên lại server mỗi khi browser tải một trang web từ server.Những thông tin được lưu trữ trong Cookie hoàn thoàn phụ thuộc vào website trên server. Mỗi websitecó thể lưu trữ một thong tin khác nhau trong Cookie. Ví dụ thời điểm lần cuối truy cập website, đánh dấu đã login hay chưa…Cookie được tạo bởi website và gửi tới browser, do vậy hai website khác nhau sẽ có hai Cookie khác nhau gửi tới browser. Ngoài ra mỗi browser quản lý và lưu trữ Cookie theo cách riêng của mình, cho nên hai browser cùng truy cập vào một website sẽ nhân được hai Cookie khác nhau.

* Để thiết lập được Cookie ta sử dụng cú pháp :

SetCookie(“tên\_Cookie”, “giá trị”, thời gian sống)

Tên\_Cookie là tên mà chúng ta đặt cho phiên làm việc.

Gía trị là thông số của tên\_Cookie.

Ví dụ : SetCookie(“username”, “admin”, time()+3000);

* Để sử dụng lại Cookie vừa thiết lập ta sử dụng cú pháp :

Cú pháp : $\_Cookie(“ tên\_Cookie\_ đã\_thiết\_lập”)

* Để hủy một Cookie đã được tạo ta dùng 1 trong hai cách sau :

Cách 1 cú pháp : setCookie(“ tên\_Cookie ”)

🡪Gọi hàm setCookie với duy nhất tên Cookie

Cách 2 dùng thời gian hết hạn Cookie là thời điểm trong quá khứ .

Ví dụ : SetCookie(“username”, “admin”, time() - 3000);

**Section** được hiểu là khoảng thời gian sử dụng giao tiếp với một ứng dụng. Một Section được sử dụng khi người dùng truy cập vào ứng dung lần đầu tiên và kết thúc khi người sử dụng thoát khỏi ứng dụng. Mỗi Section sẽ có một mã định danh khác nhau.

Để thiết lập một Section ta sử dụng cú pháp : Section\_start()

Đoạn code section này phải đặt trên các kịch bản HTML. Hoặc những lệnh echo , printf. Để thiết lập 1 giá trị Section, ngoài việc cho phép bắt đầu thực thi Section chúng ta còn phải đăng kí 1 giá trị Section để tiện cho việc gán giá trị cho Section đó.

Ta có cú pháp sau :Section\_register(“name”)

* Để sử dụng giá trị của Section ta sử dụng mã lệnh sau :

$\_ Section[“name”]

Với name là tên mà chúng ta sử dụng Section\_register(“name”) để khai báo.

* Để hủy bỏ giá trị của Section ta có những cách sau :

Section\_destroy() – Cho phép hủy bỏ toàn bộ giá trị cuả Section

Section\_unset() – Cho phép hủy bỏ Section

* 1. **Hàm**

Để giảm thời gian lặp lại 1 thao tác code nhiều lần. PHP hỗ trợ người lập trình việc tự định nghĩa cho mình những hàm có khả năng lặp lại nhiều lần trong website. Việc này cũng giúp cho người lập trình kiểm soát mã nguồn một cách mạch lạc. Đồng thời có thể tỳ biến ở mọi trang mà không cần phải khởi tạo hay viết lại mã lệnh như HTML thuần.

* Hàm tự định nghĩa

Cú pháp :

function functon\_name(){

// Các dòng mã thực thi

}

Tên hàm có thể là một tổ hợp bất kì những chữ cái, con số và dấu gạch dưới nhưng phải bắt đầu bằng chữ cái và dầu gạch dưới.

* Hàm tự định nghĩa với các tham số

Cú pháp :

function functon\_name($gt1, $gt2){

// Các dòng mã thực thi

}

* Hàm tự định nghĩa với giá trị trả về

function functon\_name(có hoặc không có đối số){

// Các dòng mã thực thi return giatri;

}

* Gọi lại hàm

PHP cung cấp nhiều hàm cho phép triệu gọi lại file. Như là hàm Include(“ URL đến file ”) hoặc Require(“ URL đến file ”).

Ngoài hai cú pháp trên còn có include\_once() và require\_once(). Hai hàm này cũng có trách nhiệm gọi lại hàm nhưng chúng sẽ chỉ gọi lại duy nhất 1 lần mà thôi.

1. **Giới thiệu về MySQL.** 
   1. **Tông quan về MySQL.**

MySQL là hệ quản trị cơ sở dữ liệu miễn phí, được tích hợp sử dụng chung với Apache, PHP. Chính yếu tố phát triển trong cộng đồng mã nguồn mở nên MySQL đã qua rất nhiều sự hỗ trợ của những lập trình viên yêu thích mã nguồn mở. MySQL cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL. Nhưng MySQL không bao quát toàn bộ những câu truy vấn cao cấp như SQL. Về bản chất thì MySQL chỉ đáp ứng việc truy xuất đơn giản trong qua trình vận hành của website nhưng hầu hết có thể giải quyết các bài toán trong PHP.

* 1. **Khởi động và sử dụng.**

Chúng ta sử dụng command như sau : MySQL-hname-uuser-ppass để truy cập vào cơ sở dữ liệu hoặc sử dụng bộ appserv để vào nhanh hơn theo đường dẫn sau Start/Appserv/MySQL command Line client. Sau đó nhập password mà chúng ta đã đặt vào.

Tuy nhiên nếu đã cài đặt ứng dung Xampp hay Wampp thì trong đó đã có tích hợp sẵn cho chúng ta chỉ cần bật chúng lên và khởi đông MySQL một cách vô cùng tiện lợi.

* 1. **Một số thuật ngữ.**

NULL : giá trị cho phép để rỗng, AUTO\_INCREMENT : cho phép giá trị tăng dần một cách tự động, UNSIGNED : phải là giá trị sô nguyên dương, PRIMARY KEY : cho phép thuộc tính là khóa chính trong bảng.

* 1. **Những cú pháp cơ bản.**
* Tạo một cơ sở dữ liệu:

CREATE DATABASE tên\_cơ\_sở\_dữ\_liệu;

Cú pháp sử dụng cơ sở dữ liệu : Use tên\_database;

Cú pháp thoát khỏi cơ sở dữ liệu :Exit;

* Cú pháp tạo một bảng trong cơ sở dữ liệu:

CREATE TABLE user(<tên cột 1><mô tả 1>,..,<tên cột n><mô tả n>)

Hiển thị có bao nhiều bảng: SHOW <tables>;

Hiển thị có bao nhiêu cột trong bảng: SHOW<columns>FROM <table>

Thêm một cột vào bảng có sẵn :

ALTER TABLE tên\_bảng ADD<tên\_cột><thuộc\_tính>AFTER

<tên\_cột>

* Thêm giá trị vào bảng

INSERT INTO tên\_bảng(tên\_cột) VALUE (Giá tri tương ứng)

* Truy vấn dữ liệu

SELECT tên\_cột FROM tên\_bảng

* Truy vấn dữ liệu có điều kiện

SELECT tên\_cột FROM tên\_bảng WHERE điều kiện

* Truy vấn dữ liệu và sắp xếp theo trinh tự :

SELECT tên\_cột FROM tên\_bảng WHERE điều kiện

ORDER BY kiểu sắp xếp( ASC hoặc DESC)

* Cập nhật dữ liệu trong bảng:

UPDATE tên\_bảng SET tên\_cột=giá\_trị\_mới WHERE điều\_kiện

* Xóa dữ liệu trong bảng

DELETE FROM tên\_bảng WHERE điều\_kiện

1. **Giới thiệu về UML.** 
   1. **UML là gì ?**

UML (Unified Modeling Language) là ngôn ngữ dành cho việc đặc tả, hình dung, xây dựng và làm tài liệu của các hệ thống phần mềm. UML tạo cơ hội để viết thiết kế hệ thống, bao gồm những khái niệm như tiến trình nghiệp vụ và các chức năng của hệ thống. Cụ thể, nó hữu dụng cho những ngôn ngữ khai báo, giản đồ cơ sở dữ liệu, thành phần phần mềm có khả năng tái sử dụng. UML được phát triển bởi Rational Rose và một số nhóm cộng tác, nó nhanh chóng trở thành một trong những ngôn ngữ chuẩn để xây dựng hệ thống phần mềm hướng đối tượng (Object-Oriented). Đây là ngôn ngữ kế vị xứng đáng cho những ngôn ngữ mô hình hoá như Booch, OOSE/Jacobson, OMT và một số các phương thức khác.

* 1. **Mục tiêu của UML.**

UML cung cấp cho người dùng một ngôn ngữ mô hình hoá trực quan sẵn sàng để dùng và có ý nghĩa:

* Cung cấp khả năng mở rộng và chuyên môn hoá để mở rộng những khái niệm cốt lõi.
* Độc lập với ngôn ngữ lập trình chuyên biệt và các tiến trình phát triển.
* Cung cấp nền tảng về sự hiểu biết ngôn ngữ mô hình hoá.
* Khuyến khích và hỗ trợ sự phát triển của các công cụ hướng đối tượng.
* Hỗ trợ những khái niệm phát triển cấp độ cao như collaboration, framework, pattern and component.
* Tích hợp một cách tốt nhất với thực tiễn.
  1. **Các biểu đồ trong UML.**
* Biểu Đồ Lớp (Class Diagram)

Class diagram là xương sống của hầu như tất cả các phương pháp hướng đối tượng, bao gồm cả UML. Chúng mô tả các cấu trúc tĩnh của hệ thống.

* Biểu Đồ Gói (Package Diagram)

Package Diagram là tập hợp các class diagram. Các package diagram thiết lập mối quan hệ giữa các pakage, trong đó pakage là những nhóm phần tử của hệ thống có mối quan hệ liên quan đến nhau.

* Biểu Đồ Chức Năng( Uses Case Diagram)

Uses case diagram đưa ra cách nhìn bao quát (từ trên xuống) cách sử dụng của hệ thống cũng như cách nhìn hệ thống từ bên ngoài. Biểu đồ này hiển thị những chức năng của hệ thống hoặc các lớp và tương tác của hệ thống với thế giới bên ngoài như thế nào. Uses case diagram được dùng trong quá trình phân tích hệ thống để nắm bắt được yêu cầu của hệ thống và hiểu được sự hoạt động của hệ thống.

* Biểu Đồ Trình Tự ( Sequence Diagram)

Sequence diagram mô tả sự tương tác của các lớp trong trình tự về thời gian. Những mô hình này được liên kết với phương pháp case (tình huống). Sequence diagram hiển thị cho bạn từng bước những sự kiện xảy ra trong phương pháp case. Biểu đồ này là cách tốt nhất để phân tích và thiết kế hệ thống bởi vì nó khá đơn giản và dễ lĩnh hội. Đây là một ví dụ nhỏ mô tả một quá trình phục vụ việc bấm nút đi thang máy.

* Biểu Đồ Hoạt Động (Activity Diagram)

Activity Diagram mô tả tiến trình xử lý và trình tự những hành động trong tiến trình xử lý, Trông nó giống như biểu đồ tiến trình (flowchart) bởi vì nó mô tả dòng làm việc từ hoạt động sang hoạt động và từ hoạt động sang trạng thái. Khi xây dựng activity diagram nó giúp bạn có thể hiểu được toàn bộ tiến trình hoạt động. Nó rất hữu dụng khi bạn mô tả những tiến trình song song hoặc mô tả một vài tương tác trong use case.

* Biểu Đồ Trạng Thái (Statechart Diagram)

Bạn sử dụng Statechart Diagram mô tả những hành động của các lớp và đối tượng riêng lẻ, mô tả trình tự những trạng thái mà các đối tượng sẽ đi qua.

* Biểu Đồ Tương Tác ( Collaboration Diagram)

Collaboration Diagram cung cấp về cách nhìn sự tương tác hoặc mối quan hệ có cấu trúc giữa các đối tượng trong mô hình hiện thời. Collaboration Diagram bao gồm các đối tượng, liên kết và thông báo. Sử dụng mô hình như là một phương tiện chính để mô tả những tương tác và cách giải quyết của các hành vi trong hệ thống.

* Biểu Đồ Thành Phần (Component Diagram)

Component Diagram cho chúng ta cách nhìn vật lý của mô hình thực tế. Nó thể hiện rõ cho chúng ta thấy sự cấu tạo và sự phụ thuộc giữa các thành phần của phần mềm bao gồm mã nguồn, mã nhị phân (binary code) và những thành phần có khả năng thực thi.

* Biểu Đồ Triển Khai (Deployment Diagram)

Deployment Diagrams mô tả các tài nguyên vật lý trong hệ thống, bao gồm các nút (node), thành phần và kết nối. Mỗi mô hình chỉ bao gồm một deployment diagram hiển thị ánh xạ giữa những tiến trình xử lý tới thiết bị phần cứng.

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

1. **Tổ chức hệ thống website**
   1. **Yêu cầu hệ thống**
      1. **Các chức vụ cần có trong hệ thống**

* Phía người quản trị

Admin: có quyền sử dụng toàn bộ chức năng của hệ thống.

* Phía khách hàng

Visiter: Chỉ có thể xem mặt hàng và các thông tin liên quan

Customer: Yêu cầu đăng nhập, cho phép đặt hàng và thanh toán cùng với các chức năng liên quan: xem giỏ hàng, mua hàng, thông tin khách hàng,…

* + 1. **Chức năng chính**
* Phía người quản trị
* Quản lý sản phẩm

+ Mục đích: quản lý sản phẩm cho cửa hàng

+ Quy trình hoạt động: chức năng giành cho người dùng có quyền Admin, có thể thêm/sửa/xoá sản phẩm.

* Quản lý danh mục sản phẩm

+ Mục đích: quản lý các danh mục sản phẩm

+ Quy trình hoạt động: chức năng giành cho người dùng có quyền Admin, có thể thêm/sửa/xoá các danh mục

* Quản lý xưởng sản xuất

+ Mục đích: quản lý xưởng sản xuất, làng nghề cung ứng sản phẩm.

+ Quy trình hoạt động: chức năng giành cho người dùng có quyền Admin, có thể thêm/sửa/xoá các xưởng sản xuất.

* Quản lý slider

+ Mục đích: quản lý các slider có trên giao diện của website.

+ Quy trình hoạt động: chức năng giành cho người dùng có quyền Admin, có thể thêm/sửa/xoá các slider.

* Quản lý đơn hàng

+ Mục đích: Quản lý các đơn hàng đã đặt mua

+ Quy trình hoạt động: chức năng giành cho người dùng có quyền Admin.

* Phía khách hàng
* Xem sản phẩm

+ Mục đích: xem các thông tin về sản phẩm

+ Quy trình hoạt động: tiến hành chọn một sản phẩm bất kỳ để xem thông tin

* Thêm sản phẩm vào giỏ

+ Mục đích: chuẩn bị đặt mua sản phẩm.

+ Quy trình thực hiện: phải đăng nhập để tiến hàng đặt hàng, người dùng chọn một sản phẩm rồi đặt mua bằng các thêm vào giỏ hàng

* Mua hàng

+ Mục đích: mua các sản phẩm có trong giỏ hàng.

+ Quy trình thực hiện: đặt mua các sản phẩm trong giỏ hàng, khách hàng cần đăng nhập và tiến hành xác nhận thông tin chọn mua hàng

* 1. **Yêu cầu chức năng**
* Đăng nhập quản trị: cho phép người quản trị đăng nhập vào trang quản trị.
* Thay đổi mật khẩu tài khoản quản trị : cho phép người dùng thay đổi mật khẩu tài khoản quản trị.
* Quản lý danh mục: thêm/sửa/xoá, xuất file với các danh mục. Chức năng giành cho người dùng có quyền admin.
* Quản lý xưởng sản xuất : thêm/sửa/xoá với các nhà xuất bản. Chức năng giành cho người dùng có quyền admin.
* Quản lý sản phẩm: thêm/sửa/xoá với các mặt hàng sách. Chức năng giành cho người dùng có quyền admin.
* Quản lý slider: thêm/sửa/xoá các slider trên trang chủ. Chức năng giành cho người dùng có quyền admin.
* Quản lý khách hàng: sửa/xoá các thông tin khách hàng. Chức năng giành cho người dùng có quyền admin.
* Quản lý đơn hàng của admin: Sửa/xoá/cập nhật trạng thái các đơn hàng đặt.
* Cài đặt chung cho trang web: hiển thị nội dung (đơn vị tiền tệ, định dạng tiền,…)
* Xem thông tin sản phẩm: khách hàng xem danh sách sản phẩm theo danh mục, xem thông tin chi tiết từng sản phẩm, sản phẩm gợi ý
* Tìm kiếm: chức năng tìm kiếm hỗ trợ ở hầu hết các chức năng: tìm kiếm người dùng, sản phẩm, khách hàng, đơn hàng,… (phía người quản trị). Hoặc là tìm kiếm sản phẩm, đơn hàng đặt,… (phía khách hàng).
* Thêm vào giỏ hàng: khách hàng cần đăng nhập để có thể thêm sản phẩm vào giỏ hàng
* Mua hàng: mua các sản phẩm có trong giỏ hàng.
* Quản lý địa chỉ khách hàng: khách hàng có thể tạo nhiều địa chỉ. Tuy nhiên chỉ có địa chỉ mà shop hỗ trợ mới có thể mua hàng
* Quản lý đơn hàng: Xem thông tin các đơn hàng đã mua và trạng thái từng đơn hàng
* Thay đổi thông tin cá nhân khách hàng: cho phép khách hàng thay đổi thông tin cá nhân của mình
  1. **Yêu cầu phi chức năng**
* Bảo mật: mật khẩu của người dùng + khách hàng đều được mã hoá, validate với mọi trường hợp của chức năng, trong đó đặc biệt tập trung vào việc phân quyền để không cho phép người dùng truy cập.
* Cập nhật: các thông tin của mặt hàng, danh mục, sản phẩm,… sau khi được cập nhật đều phải reset và hiển thị lại cho người dùng. Đặc biệt validate với các trường hợp thêm/xoá/sửa,… khi khách hàng cố tình truyền lên thanh địa chỉ
* Giao diện: cần yêu cầu phù hợp
* Bảo trì và mở rộng: hệ thống thiết kế cần dễ dàng trong việc bảo trì và mở rộng sau này.

1. **Phân tích thiết kế hệ thống**
   1. **Mô tả chi tiết các use case**
2. * 1. **Use case đăng nhập quản trị.**

Chức năng: Use case này cho phép người quản trị đăng nhập vào hệ thống quản trị của website bán hàng.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi người quản trị nhập đầy đủ thông tin bao gồm username và pasword và nhấn nút đăng nhập trên form đăng nhập admin. Hệ thống sẽ lấy thông tin tài khoản admin trong bảng tbl\_admin để đối chiếu và cho phép người quản trị đăng nhập nếu thông tin trùng khớp . Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại luồng cơ bản nếu khách nhập sai thông tin tài khoản hoặc không có trên trên hệ thống từ trong bảng **tbl\_admin**. Màn hình sẽ đưa ra thông báo “ Nhập sai tài khoản hoặc mật khẩu”. Use case kết thúc.
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Không có

Hậu điều kiện: Không có

* + 1. **Use case quản lý đơn đặt hàng**

Chức năng: Use case này cho phép người quản trị quản lý trạng thái đơn đặt hàng của khách hàng.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi người quản trị nhấn nút Quản lý đơn đặt hàng trên menu ngang của trang quản trị .Hệ thống sẽ lấy danh sách các đơn hàng trong bảng **tbl\_order** và hiển thị lên màn hình.
* Khi người quản trị kích vào nút xem địa chỉ, hệ thống sẽ lấy thông tin địa chỉ khách hàng từ bảng tbl\_customer và hiển thị lên màn hình.
* Khi người quản trị kích vào nút “Đang chờ xử lý”. Hệ thống sẽ tự động cập nhật lại trạng thái đơn hàng thành “Đang giao hàng”. Nếu khách hàng đã xác nhận nhận được hàng người quản trị có quyền kích nút “Xóa đơn đã giao” để xóa đơn hàng. Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Người quản trị phải đăng nhập quản trị.

Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về danh sách các đơn đặt hàng sẽ được cập nhập trong cơ sở dữ liệu trong bảng **tbl\_order**

* + 1. **Use case quản lý slider**

Chức năng: Use case này cho phép người quản trị quản lý các slider hiển thị trên trang chủ website.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi người quản trị nhấn nút Quản lý silder trên menu dọc bên trái của trang quản trị.
* Khi người quản trị kích vào nút Thêm slider, hệ thống hiển thị lên màn hình form thêm lên màn hình .Người quản trị nhập đủ các trường thông tin và nhấn nút Lưu lại. Hệ thống sẽ lưu những thông tin slider mới vào bảng **tbl\_slider.** Use case kết thúc.
* Khi người quản trị kích vào nút Danh sách slider ,hệ thống sẽ lấy thông tin danh sách slider từ bảng **tbl\_slider** và hiển thị lên màn hình.Nếu người quản trị cập nhật trạng thái slide (bật – tắt ) hoặc xóa thi hệ thống sẽ cập nhật lại thông tin trên bảng **tbl\_slider** . Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Người quản trị phải đăng nhập quản trị.

Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về danh sách slider sẽ được cập nhập trong cơ sở dữ liệu trong bảng **tbl\_slider**

* + 1. **Use case quản lý danh mục sản phẩm**

Chức năng: Use case này cho phép người quản trị quản lý các danh mục sản phẩm hiển thị trên website.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi người quản trị nhấn nút Quản lý danh mục trên menu dọc bên trái của trang quản trị.
* Khi người quản trị kích vào nút Thêm danh mục, hệ thống hiển thị lên màn hình form thêm lên màn hình .Người quản trị nhập đủ các trường thông tin và nhấn nút Lưu lại. Hệ thống sẽ lưu những thông tin danh mục mới vào bảng **tbl\_category.** Use case kết thúc.
* Khi người quản trị kích vào nút Danh sách danh mục ,hệ thống sẽ lấy thông tin danh sách danh mục từ bảng **tbl\_category** và hiển thị lên màn hình. Nếu người quản trị cập nhật hoặc xóa danh mục thì hệ thống sẽ cập nhật lại thông tin danh sách danh mục trên bảng **tbl\_category**. Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Người quản trị phải đăng nhập quản trị.

Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về danh mục sản phẩm sẽ được cập nhập trong cơ sở dữ liệu trong bảng **tbl\_category**

* + 1. **Use case quản lý xưởng sản xuất**

Chức năng: Use case này cho phép người quản trị quản lý danh sách các làng nghề/xưởng sản xuất.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi người quản trị nhấn nút Quản lý làng nghề sx trên menu dọc bên trái của trang quản trị.
* Khi người quản trị kích vào nút Thêm làng nghề sản xuất, hệ thống hiển thị lên màn hình form thêm lên màn hình .Người quản trị nhập đủ các trường thông tin và nhấn nút Lưu lại. Hệ thống sẽ lưu những thông tin làng nghề sản xuất mới vào bảng **tbl\_brand.** Use case kết thúc.
* Khi người quản trị kích vào nút Danh sách làng nghề sản xuất ,hệ thống sẽ lấy thông tin danh sách làng nghề sản xuất từ bảng **tbl\_brand** và hiển thị lên màn hình. Nếu người quản trị cập nhật hoặc xóa làng nghề sản xuất thì hệ thống sẽ cập nhật lại thông tin danh sách các làng nghề trên bảng **tbl\_brand**. Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Người quản trị phải đăng nhập quản trị.

Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về xưởng sản xuất sẽ được cập nhập trong cơ sở dữ liệu trong bảng **tbl\_brand**

* + 1. **Use case quản lý sản phẩm**

Chức năng: Use case này cho phép người quản trị quản lý các sản phẩm hiển thị trên website.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi người quản trị nhấn nút Quản lý sản phẩm trên menu dọc bên trái của trang quản trị.
* Khi người quản trị kích vào nút Thêm sản phẩm, hệ thống hiển thị lên màn hình form thêm lên màn hình .Người quản trị nhập đủ các trường thông tin và nhấn nút Lưu lại. Hệ thống sẽ lưu những thông tin sản phẩm mới vào bảng **tbl\_product.** Use case kết thúc.
* Khi người quản trị kích vào nút Danh sách sản phẩm ,hệ thống sẽ lấy thông tin danh sách sản phẩm từ bảng **tbl\_product** và hiển thị lên màn hình. Nếu người quản trị cập nhật hoặc xóa sản phẩm thì hệ thống sẽ cập nhật lại thông tin danh sách các sản phẩm trên bảng **tbl\_product.** Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Người quản trị phải đăng nhập quản trị.

Hậu điều kiện: Không có

* + 1. **Use case đăng nhập khách hàng**

Chức năng: Use case này cho phép khách hàng đăng nhập vào hệ thống của website bán hàng.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi khách hàng nhấn nút đăng nhập trên góc trên bên phải màn hình, hệ thống sẽ hiển thi form đăng nhập. Khách hàng nhập đầy đủ thông tin bao gồm email và pasword và nhấn nút đăng nhập trên form đăng nhập khách hàng. Hệ thống sẽ lấy thông tin tài khoản khách hàng trong bảng **tbl\_customer** để đối chiếu và cho phép khách hàng đăng nhập nếu thông tin trùng khớp . Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại luồng cơ bản nếu khách nhập sai thông tin tài khoản hoặc không có trên trên hệ thống từ trong bảng **tbl\_customer**. Màn hình sẽ đưa ra thông báo “ Nhập sai tài khoản hoặc mật khẩu”.
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Không có

Hậu điều kiện: Không có

* + 1. **Use case**

Chức năng: Use case này cho phép khách hàng đăng kí một tài khoản mới trên hệ thống của website bán hàng.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi khách hàng nhấn nút đăng nhập trên góc trên bên phải màn hình, hệ thống sẽ hiển thị lên một form đăng kí khách hàng. Khách hàng nhập đầy đủ thông tin (Họ tên, đại chỉ nhà, số điện thoại, số CCCD, địa chỉ email, mật khẩu )và nhấn nút Tạo tài khoản trên form đăng kí khách hàng. Hệ thống sẽ lấy thông tin tài khoản khách hàng mới vừa nhập và thêm vào bảng **tbl\_customer** . Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Không có

Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về khách hàng mới sẽ được thêm trong cơ sở dữ liệu trong bảng **tbl\_customer**

* + 1. **Use case xem chi tiết sản phẩm**

Chức năng: Use case này cho phép khách hàng xem thông tin chi tiết một sản phẩm bất kì.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi khách hàng nhấn nút Xem chi tiết trên màn hình hiển thị sản phẩm, hệ thống sẽ lấy thông tin về chi tiết về sản phẩm đó từ bảng **tbl\_product** để hiển thị ra màn hình . Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Không có

Hậu điều kiện: Không có

* + 1. **Use case xem danh sách sản phẩm**

Chức năng: Use case này cho phép khách hàng xem danh sách sản phẩm hiện có của website.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi khách hàng nhấn nút Sản phẩm trên thanh menu của màn hình trang chủ website, hệ thống sẽ lấy thông tin về tất cả các sản phẩm của cửa hàng từ bảng **tbl\_product** để hiển thị ra màn hình . Use case kết thúc.
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Không có

Hậu điều kiện: Không có

* + 1. **Use case xem sản phẩm theo danh mục**

Chức năng: Use case này cho phép khách hàng xem thông tin chi tiết một sản phẩm nằm trong danh mục sản phẩm bất kì.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi khách hàng nhấn nút Xem chi tiết trên màn hình hiển thị sản phẩm, hệ thống sẽ lấy ra danh sách tên các danh mục sản phẩm từ bảng **tbl\_category** và hiển thị ra màn hình
* Người quản trị kích chọn tên một danh mục bất kì , hệ thống sẽ tự động lấy danh sách sản phẩm cảu danh mục đó hiển thị lên màn hình. Use case kết thúc
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Không có

Hậu điều kiện: Không có

* + 1. **Use case tìm kiếm sản phẩm**

Chức năng: Use case này cho phép khách hàng tìm kiếm sản phẩm với từ khóa nhập vào bất kì.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi khách hàng bắt đầu kích vào thanh tìm kiếm trên phần đầu trang web, hệ thống sẽ hiển thị một số gợi ý tên sản phẩm (trong trường hợp đã từng thực hiện tìm kiếm rồi thì lưu vết ).
* Người quản trị kích chọn tên một danh mục bất kì , hệ thống sẽ tự động lấy danh sách sản phẩm cảu danh mục đó hiển thị lên màn hình. Use case kết thúc
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Không có

Hậu điều kiện: Không có

* + 1. **Use case thêm sản phẩm vào giỏ hàng**

Chức năng: Use case này cho phép khách hàng thêm sản phẩm vào giỏ hàng.

Luồng sự kiện:

* + - * + Luồng cơ bản:
* Use case này bắt đầu khi khách hàng bắt đầu kích vào nút Mua ngay nằm ở trang thông tin chi tiết cảu sản phẩm, hệ thống sẽ lấy thông tin về sản phẩm(như tên , sô lượng, giá….) từ bảng **tbl\_product** và thêm vào trong bảng **tbl\_cart**
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Khách hàng phải đăng nhập tài khoản

Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về đơn hàng sẽ được cập nhập trong cơ sở dữ liệu trong bảng **tbl\_cart**.

* + 1. **Use case đặt hàng**

Chức năng: Use case này cho phép khách hàng đặt mua sản phẩm có trong giỏ hàng .

Luồng sự kiện

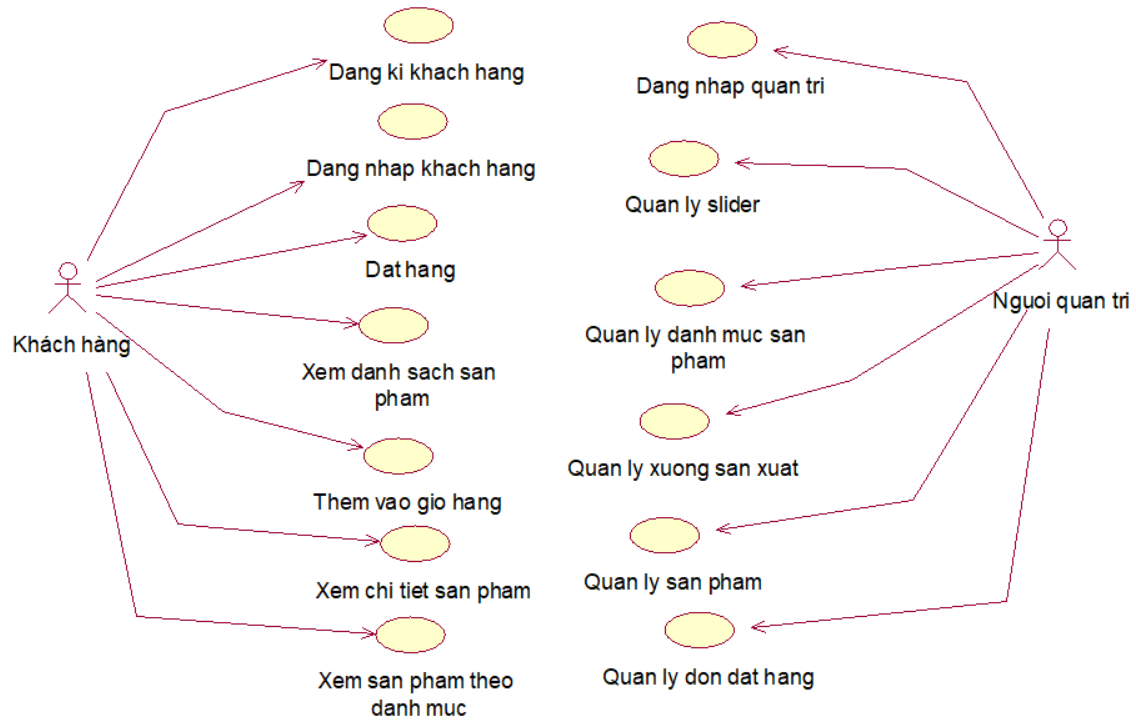
* + - * + Luồng cơ bản:
* Use case này bắt đầu khi khách hàng bắt đầu kích vào nút Check out ở trang giỏ hàng và chọn phương thức thanh toán, hệ thống sẽ lấy thông tin về đơn hàng và thông tin về khách hàng, ngày mua rồi sau đó hiển thị lên màn hình.
* Khách hàng xác nhận thông tin đơn hàng và kích vào nút Đặt hàng, hệ thống sẽ thêm thông tin đơn hàng và khách hàng vào bảng **tbl\_order** đó hiển thị lên màn hình. Use case kết thúc
  + - * + Luồng rẽ nhánh:
* Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

Yêu cầu đặc biệt: Không có

Tiền điều kiện: Khách hàng phải đăng nhập tài khoản

Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về đơn hàng sẽ được cập nhập trong cơ sở dữ liệu trong bảng **tbl\_order**

* 1. **Các biểu đồ** 
     1. **Biểu đồ Use case tổng quan**

****

Hình 2. 1 Biểu đồ Use case tổng quan

* + 1. **Các biểu đồ Use case phân rã**
    2. **Các biểu đồ trình tự**
    3. **Các biểu đồ lớp VOPC**

**Tài liệu tham khảo**

[1] Nguyễn Trung Phú, Giáo trình Thiết kế web, 2019.

[2] Trần Phương Nhung, Giáo trình Lập trình web bằng PHP, 2020.

[3] https://www.w3schools.com/