



BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO ĐỒ ÁN
CHUYÊN ĐỀ CÔNG NGHỆ PHẦN MỀM

Giảng viên hướng dẫn: **ThS.Võ Tấn Dũng**

Sinh viên thực hiện: **Nguyễn Thanh Thiên**

MSSV: 1811062304 – Lớp: 18DTHB1

Vũ Hồng Thiên

MSSV: 1811062305 – Lớp: 18DTHB2

Nguyễn Văn Thái Bảo

MSSV: 1811062068 – Lớp: 18DTHB4

Nguyễn Thị Kim Ngân

MSSV: 1811061957 – Lớp: 18DTHB4

TP. HCM, Tháng 4 năm 2022

[illegible]

Mục Lục

| | |
|------------------------------------------------|-----------|
| CHƯƠNG 1: TỔNG QUAN | 7 |
| 1.1 Giới thiệu đề tài | 8 |
| 1.2 Mô tả | 9 |
| 1.2.1 Đối tượng | 9 |
| 1.2.2 Phạm vi | 9 |
| 1.3 Các lĩnh vực được tin học hóa | 10 |
| CHƯƠNG 2: CƠ SỞ LÝ THUYẾT | 12 |
| 2.1. Tổng quan về ngôn ngữ lập trình | 12 |
| 2.1.1 HTML | 12 |
| 2.1.2 CSS | 12 |
| 2.1.3 Spring Boot | 13 |
| 2.2 Kiến thức cần thiết để học tập Spring Boot | 19 |
| 2.2.1 Java Core | 19 |
| 2.2.2 Package manager | 19 |
| 2.2.3 Spring Boot | 20 |
| 2.2.4 JPA/MongoDB/SQL | 20 |
| 2.2.5 Template engine | 20 |
| 2.2.6 Các Tool khác | 20 |
| CHƯƠNG 3: CƠ SỞ DỮ LIỆU | 21 |
| 3.1. Sơ đồ Usecase | 21 |
| 3.2. Sơ đồ Diagram | 22 |
| 3.4. Biểu đồ hoạt động (Activity) | 22 |
| CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM | 26 |

| | |
|-------------------------------------------------|-----------|
| CHƯƠNG 5: KẾT LUẬN VÀ KIẾN NGHỊ | 31 |
| 5.1 Kết luận..... | 31 |
| 5.1.1 Các chức năng đã hoàn thành..... | 31 |
| 5.1.2 Các chức năng chưa làm được..... | 31 |
| 5.2 Đánh giá website | 31 |
| 5.2.1 Ưu điểm..... | 31 |
| 5.2.2 Khuyết điểm | 31 |
| 5.3 Kiến nghị cho việc phát triển website | 31 |
| TÀI LIỆU THAM KHẢO..... | 33 |

Mục lục hình

| | |
|------------------------------------------------------------------|-----------|
| <i>Hình 2.1 Cách thức hoạt động của CSS.....</i> | <i>13</i> |
| <i>Hình 3.2 Sơ đồ Diagram.....</i> | <i>22</i> |
| <i>Hình 3.3 Biểu đồ hoạt động đăng nhập vào hệ thống.....</i> | <i>22</i> |
| <i>Hình 3.4 Biểu đồ hoạt động thoát ra khỏi hệ thống.....</i> | <i>23</i> |
| <i>Hình 3.5 Biểu đồ hoạt động xem thông tin về sản phẩm.....</i> | <i>24</i> |
| <i>Hình 3.6 Biểu đồ chức năng thêm SP vào giỏ hàng</i> | <i>25</i> |
| <i>Hình 4.8 Trang sửa sản phẩm.....</i> | <i>29</i> |
| <i>Hình 4.9 Trang thêm sản phẩm</i> | <i>30</i> |

LỜI MỞ ĐẦU

Ngày nay ứng dụng công nghệ thông tin và việc tin học hóa được xem là một trong những yếu tố mang tính quyết định trong hoạt động của các chính phủ, tổ chức, cũng như của các công ty, nó đóng vai trò hết sức quan trọng, có thể tạo ra những bước đột phá mạnh mẽ.

Việc cập nhật thông tin là rất cần thiết với nhà quản lý và nhà kinh doanh trong lĩnh vực này. Để quảng bá sản phẩm nhiều nhà kinh doanh của các thông tin đại chúng như báo, radio, truyền hình trong đó quảng bá sản phẩm trên website là một trong những giải pháp hữu hiệu nhất.

Đi đôi với sự phát triển của mạng internet thì nhu cầu chăm sóc bản thân và làm đẹp cho chính bản thân con người cũng tăng lên vì vậy mà hàng loạt những mỹ phẩm đã được cung ứng ra thị trường nhằm phục vụ khách hàng, những khách hàng có thể trực tiếp mua thông qua website trực tiếp.

Xuất phát từ những nhu cầu đó đề tài "Thiết kế website bán trang sức" được xây dựng nhằm đáp ứng cho người tiêu dùng trên toàn quốc thông qua website này. Khách hàng có thể kích vào mô hình sản phẩm để xem thông tin chi tiết về sản phẩm đó và chọn cho mình được sản phẩm ưng ý nhất.

Đồ án này là kết quả của quá trình tích lũy và vận dụng những kiến thức chúng em tiếp thu và tìm hiểu được trong quá trình học tập. Trong quá trình thực hiện đồ án em xin cảm ơn thầy Võ Tấn Dũng đã định hướng cho chúng em làm và hoàn thành đồ án này. Chúng em xin gửi tới thầy những lời cảm ơn chân thành nhất.

Mặc dù đã cố gắng trong quá trình nghiên cứu và làm đồ án nhưng do còn nhiều hạn chế về thời gian và trình độ nên đồ án của chúng em còn nhiều thiếu sót, và nhiều vấn đề chưa được giải quyết hoàn chỉnh. Vì vậy em mong nhận được những ý kiến đóng góp của thầy để có thể hoàn thiện và phát triển đề tài hơn.

LỜI CẢM ƠN

Đầu tiên, chúng em xin chân thành cảm ơn và sự tri ân sâu sắc đối với thầy cô trường Đại học Công nghệ Thành Phố Hồ Chí Minh, đặt biệt là các thầy cô khoa Công Nghệ Thông Tin của trường đã tận tình giúp đỡ cho chúng em suốt thời gian học tại trường.

Đặc biệt, chúng em xin bày tỏ lòng biết ơn chân thành và sâu sắc tới đến với **Thầy ThS.Võ Tấn Dũng** đã tận tình giúp đỡ, định hướng cách tư duy và cách làm việc khoa học. Đó là những góp ý hết sức quý báu không chỉ trong quá trình thực hiện bài đồ án chuyên ngành này mà còn là trang tiếp bước cho chúng em trong quá trình học tập sau này.

Có lẽ kiến thức là vô hạn mà sự tiếp nhận kiến thức của bản thân mỗi người luôn tồn tại những hạn chế nhất định. Do đó, trong quá trình hoàn thành đồ án chắc chắn không tránh khỏi những hạn chế, sai sót. Bản thân chúng em rất mong nhận được những đóng góp ý kiến đến từ thầy về bài đồ án chuyên ngành này của chúng em để được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

CHƯƠNG 1: TỔNG QUAN

1.1 Giới thiệu đề tài

Hiện nay, mọi vấn đề trong cuộc sống xung quanh chúng ta đang dần dần được tin học hóa, điều này diễn ra từ những quốc gia lớn mạnh hay những thứ nhỏ nhất để phục vụ cho nhu cầu đời sống của con người như: thiết bị nghe gọi bắt đầu tiên tiến và được cải thiện nhiều hơn đối với nhược điểm của cái cũ, việc buôn bán hàng, quản lý kho... Sự phát triển mạnh của công nghệ như thế giúp cho những công ty, doanh nghiệp, cửa hàng... càng có nhiều nhu cầu về mặt quảng cáo và bán hàng hơn tăng doanh thu cho cơ sở nhiều hơn. Hay việc quản lý các mặt hàng nhập, xuất kho, tính toán, lưu trữ, thống kê. Nếu như làm bằng thủ công ghi tay hoặc ghi vào giấy ghi nhớ thì dễ thất lạc và khó nhớ.

Với tình hình dịch bệnh diễn biến khá phức tạp như hiện nay thì việc đến cửa hàng và di chuyển hoặc đến cửa hàng để chọn sản phẩm, thanh toán bằng tiền mặt và qua tiếp xúc thì có nguy cơ cao bị nhiễm bệnh và có thể khiến tình trạng lây lan của dịch bệnh càng trầm trọng hơn. Đối với việc như thế thì việc phát triển một website như thế này thì có thể đảm bảo an toàn cho người mua hàng và có thể dễ dàng chọn sản phẩm, thanh toán dễ dàng trên website mà không cần qua tiếp xúc.

Dựa vào việc giúp đỡ cho nhu cầu mua hàng của người mua hàng trong tình hình dịch bệnh như hiện nay thì chúng em muốn tin học hóa những việc như: di chuyển, lựa chọn, thanh toán, giao hàng... cũng như thay đổi các thao tác bằng thủ công sang các thao tác trên máy tính tiện lợi hơn cho người mua lẫn người bán. Chúng em quyết định chọn đề tài “Website bán trang sức”. Để giúp chủ cửa hàng lẫn người mua hàng có thể kinh doanh, quảng cáo, buôn bán dễ dàng và hạn chế tiếp xúc hay tiết kiệm thời gian cho mọi người.

1.2 Mô tả

1.2.1 Đối tượng

Phần mềm được xây dựng để hướng đến các đối tượng như:

- +Người bán hàng
- +Người sử dụng
- +Người quản lý hệ thống

1.2.2 Phạm vi

a) Thông tin tổng quan

Đơn vị sử dụng: Cửa hàng bán đồ trang sức.

Tên dự án: Web bán trang sức.

b) Phát biểu vấn đề

Hiện nay xu hướng tin học hóa toàn cầu, nhiều công ty, cửa hàng được mở ra, và cửa hàng kinh doanh đồ trang sức cần một phần mềm để dễ dàng kinh doanh, quảng cáo, lưu trữ, tìm kiếm các mặt hàng cho các khách hàng của cửa hàng. Phần mềm sẽ mang nhiều hiệu quả công việc tốt cho cửa hàng và khách mua hàng có thể tiết kiệm được thời gian mua hàng. Giải quyết những công việc lúc trước bằng thủ công sang tin học hóa.

c) Mục tiêu

- Cho phép chủ cửa hàng xem, theo dõi thông số doanh thu bán hàng.
- Cho phép khách hàng xem, tìm kiếm, chọn mặt hàng và thanh toán dễ dàng, nhanh chóng và tiện lợi.
- Hạn chế tiếp xúc trong quá trình mua hàng.
- Cho phép quảng cáo các mặt hàng mới hoặc bán chạy của cửa hàng.

d) Mô tả

Website gồm các banner ở đầu để quảng cáo đối với các sản phẩm mới hoặc các sản phẩm bán chạy.

Các thông tin của các mặt hàng được hiển thị đầy đủ như: giá cả, số lượng...

Mỗi khách hàng có tài khoản cá nhân dùng để đăng nhập và mua hàng.

e) Lợi ích

- Dễ dàng quản lý các doanh thu, doanh số theo tháng, năm cho chủ cửa hàng.
- Giúp người mua hàng tiết kiệm thời gian và chi phí.
- Hạn chế việc tiếp xúc trong tình hình dịch bệnh hiện nay.
- Có thể quảng cáo các sản phẩm.

f) Các bước thực hiện

Lập kế hoạch phát triển hệ thống

Phân tích thiết kế hệ thống

Phát thảo kế hoạch, dự án

Thiết kế theo bản thảo

Kiểm tra

Biên soạn tài liệu và hướng dẫn sử dụng

1.3 Các lĩnh vực được tin học hóa

Trong tình hình dịch bệnh, hạn chế tiếp xúc hiện nay các lĩnh vực được tin học hóa sau đây có thể giúp ích khá nhiều cho người mua hàng và chủ cửa hàng:

+Tìm kiếm cửa hàng: Với việc tìm kiếm địa chỉ của cửa hàng thì bạn chỉ cần gõ địa chỉ website thì không cần đi đến tận cửa hàng. Làm bạn mất nhiều thời gian cũng như chi phí đi lại.

+Lựa chọn mặt hàng: Với nhiều mặt hàng trên website bạn không cần phải đi loay hoay ở cửa hàng để chọn mặt hàng .

+Trao đổi với nhân viên: Với tình hình dịch bệnh thì bạn cũng không thể ra tận cửa hàng để trao đổi với nhân viên để nhận tư vấn về sản phẩm thì website có thể giúp bạn chat trực tiếp với nhân viên cửa hàng để nhận tư vấn cho sản phẩm bạn muốn mua.

+Tính toán: Việc thanh toán dựa vào máy tính sẽ giúp người mua lẫn người bán tiết kiệm thời gian tính toán và tăng độ chính xác hơn. Có thể tính toán giảm theo % đối với những khách hàng mua số lượng nhiều.

+Thanh toán: Giúp bạn có nhiều phương thức thanh toán hơn như thanh toán qua thẻ... và hạn chế tiếp xúc trong tình trạng hiện nay.

+Đặt hàng: Dễ dàng đặt hàng để nhân viên giao tận nhà mà bạn không cần chờ từ cửa hàng về đến tận nhà. Hoặc phải thuê người khác chở về nhà cho bạn.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Tổng quan về ngôn ngữ lập trình

2.1.1 HTML

Ngôn ngữ HTML (HyperText Markup Language – ngôn ngữ siêu văn bản) là một trong các loại ngôn ngữ được sử dụng trong lập trình web. Khi truy cập một trang web cụ thể là click vào các đường link, bạn sẽ được dẫn tới nhiều trang các nhau, và các trang này được gọi là một tài liệu HTML (tập tin HTML).

Một trang HTML như vậy được cấu thành bởi nhiều phần tử HTML nhỏ và được quy định bằng các thẻ tag. Bạn có thể phân biệt một trang web được viết bằng ngôn ngữ HTML hay PHP thông qua đường link của nó. Ở cuối các trang HTML thường hay có đuôi là .HTML hoặc .htm

Vai trò của HTML trong lập trình web là một loại ngôn ngữ đánh dấu siêu văn bản, thế nên các chức năng của nó cũng xoay quanh yếu tố này. Cụ thể, HTML giúp cấu thành các cấu trúc cơ bản trên một website (chia khung sườn, bố cục các thành phần trang web) và góp phần hỗ trợ khai báo các tập tin kỹ thuật số như video, nhạc, hình ảnh.

Ưu điểm nổi trội nhất và cũng là thế mạnh của HTML là khả năng xây dựng cấu trúc và khiến trang web đi vào quy củ một hệ thống hoàn chỉnh. Nếu bạn mong muốn sở hữu một website có cấu trúc tốt có mục đích sử dụng nhiều loại yếu tố trong văn bản, hãy hỏi HTML. Nhiều ý kiến cho rằng tùy theo mục đích sử dụng mà lập trình viên hay người dùng có thể lựa chọn ngôn ngữ lập trình riêng cho website của bạn, tuy nhiên thực chất HTML chứa những yếu tố cần thiết mà dù website của bạn có thuộc thể loại nào, giao tiếp với ngôn ngữ lập trình nào để xử lý dữ liệu thì nó vẫn phải cần đến *ngôn ngữ HTML* để hiển thị nội dung cho người truy cập.

2.1.2 CSS

CSS – Cascading Style Sheet là một ngôn ngữ lập trình được thiết kế vô cùng đơn giản, dễ sử dụng. Với CSS hướng tới mục tiêu chính là giúp đơn giản hóa được quá trình tạo ra các website.

Nhiệm vụ chính của CSS là thực hiện việc xử lý giao diện của một trang web cụ thể. Đó là những yếu tố như màu sắc văn bản, hay khoảng cách giữa các đoạn, hoặc kiểu font chữ, hình ảnh, bố cục, màu nền, ... đều có thể thay đổi, chỉnh sửa theo ý muốn với hỗ trợ của CSS.

Sử dụng CSS chính là một công cụ hỗ trợ hữu ích và cần thiết cho quá trình phát triển, thiết kế website. Những công việc mà HTML không xử lý được sẽ được giải quyết tốt với CSS. Nhờ có CSS giúp mang lại khả năng kiểm soát mạnh mẽ,

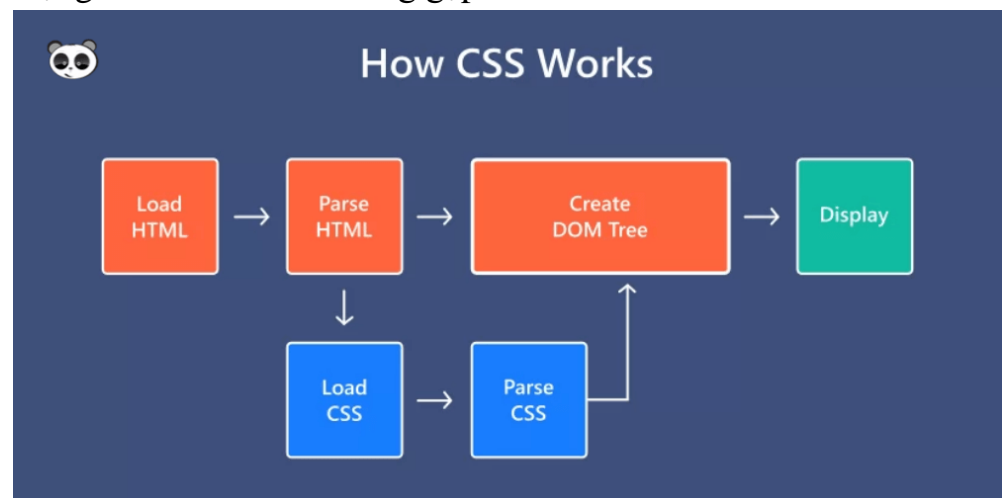
hiệu quả với quá trình trình bày một tài liệu HTML cụ thể. Bởi thế mà nó là công cụ thường được kết hợp với các ngôn ngữ như HTML hay XHTML.

Ưu điểm:

- Khả năng tiết kiệm thời gian
- Dễ dàng khi thực hiện bảo trì
- Khả năng tải trang nhanh chóng

Nhược điểm:

- CSS hoạt động khác biệt cho từng trình duyệt
- Khá khó khăn cho người mới
- Định dạng của web có khả năng gặp rủi ro



Hình 2.1 Cách thức hoạt động của CSS

2.1.3 Spring Boot

Là một phần cực kỳ quan trọng của Spring framework, do vậy trước khi tìm hiểu Spring Boot là gì, chúng ta cần có kiến thức cơ bản về Spring framework.

2.1.3.1 Spring framework là gì?

Spring là một framework được ra đời để giúp các nhà phát triển có thể xây dựng hệ thống và chạy ứng dụng trên JVM một cách thuận tiện, đơn giản và nhanh chóng. Đây là một mã nguồn mở được phát triển và rất nhiều người sử dụng.

Trên thực tế, Spring framework là tập hợp gồm rất nhiều các dự án nhỏ khác nhau như: Spring MVC (sử dụng để xây dựng các ứng dụng trên nền tảng web), Spring Data, Spring Boot,...

Để phát triển một ứng dụng web cơ bản sử dụng Spring framework, bạn cần trải qua ít nhất 5 công đoạn sau:

- Tạo một project sử dụng Maven với các dependency cần thiết.
- Tạo một tập tin web đuôi (.xml) để khai báo DispatcherServlet (thuộc Spring MVC).
- Một tập tin có cấu hình của Spring MVC.
- Trả về một class Controller khi có request đến.
- Cuối cùng là phải có một web server phục vụ triển khai ứng dụng lên chạy.

Để khởi tạo một dự án Spring khá là vất vả và tốn nhiều công sức. Người lập trình phải khai báo những dependency trong pom.xml, cấu hình XML hoặc các annotation cực kỳ phức tạp. Giờ đây, việc tạo ra các ứng dụng này sẽ được thực hiện một cách đơn giản, nhanh chóng hơn rất nhiều chỉ với Spring Boot. Vậy Spring Boot là gì?

2.1.3.2 Khái niệm về Spring Boot

Từ những thông tin ở trên, ta có thể khẳng định Spring Boot chính là một Java framework siêu to và khổng lồ và có nhiều khả năng hữu ích vì nó có thể giúp lập trình viên giải quyết rất nhiều vấn đề. So với framework Spring thông thường, Spring Boot tỏ ra những lợi thế vượt trội. Khi sử dụng Spring Boot, rất nhiều thứ được cải tiến hỗ trợ lập trình viên như:

- Auto config: tự động cấu hình thay lập trình viên, bạn chỉ cần viết code và tiến hành chạy hệ thống là được.
- Dựa trên các Annotation để tạo lập các bean thay vì XML.
- Server Tomcat có thể được nhúng ngay trong file JAR build ra và có thể chạy ở bất kì đâu mà java chạy được.

Khi sử dụng Spring Boot, lập trình viên chỉ cần:

- Sử dụng Spring Initializr: nhập các thông tin của dự án (project), chọn thư viện (Library) rồi tải code về máy.
- Mở mã nguồn (source code) và bắt đầu viết code.
- Có thể chạy ngay trong IDE, hoặc build thành file JAR mà không cần cấu hình config cho server nữa.

2.1.3.3 Tại sao bạn nên học Spring Boot?

Việc ứng dụng Spring Boot giúp các lập trình viên tập trung tốt hơn đến mục tiêu viết code của mình thay vì phải lo nghĩ nhiều đến các kỹ thuật code như với Node.js.

Bản thân Spring Boot chứa đựng nhiều các library bên trong, cấu trúc code cũng trở nên chuẩn mực hơn, bạn không cần phải quá bận tâm đến việc code làm sao cho thật tốt. Do đó, bạn có nhiều thời gian hơn để tập trung vào sự logic của sản phẩm.

Spring Boot là sự cải tiến tuyệt vời của Spring, cải thiện được rất nhiều khâu rườm rà đến từ Spring. Hơn thế, quá trình học hỏi và tiếp thu Spring Boot cũng trở nên dễ dàng hơn rất nhiều.

Đừng cố giữ những quan điểm cổ hủ cho rằng muốn biết về Spring Boot trước tiên phải học Spring. Thực chất đây là điều thực sự không cần thiết, bạn hoàn toàn có thể lựa chọn học Spring Boot ngay từ đầu vì nó giúp bạn hiểu và ứng dụng các vấn đề nhanh hơn rất nhiều.

2.1.3.4 Đặc điểm và tính năng của Spring Boot

Ưu điểm nổi bật của Spring Boot

Những tính năng nổi bật của Spring Boot mà bạn không thể bỏ qua như:

- Phát triển web một cách đơn giản và nhanh chóng.
- Config an toàn
- Có thể hỗ trợ YAML
- Có tính quản trị cao
- Dễ dàng ứng dụng Spring và các sự kiện.
- Có thể cấu hình ở bên ngoài và tạo ra những tệp thuộc tính.
- Tính bảo mật cao.
- Ghi log

Với những ưu điểm mà Spring Boot mang lại, nó cũng mang theo rất nhiều đặc điểm ưu việt. Cùng tìm hiểu và tận dụng những chương trình này một cách hiệu quả nhé.

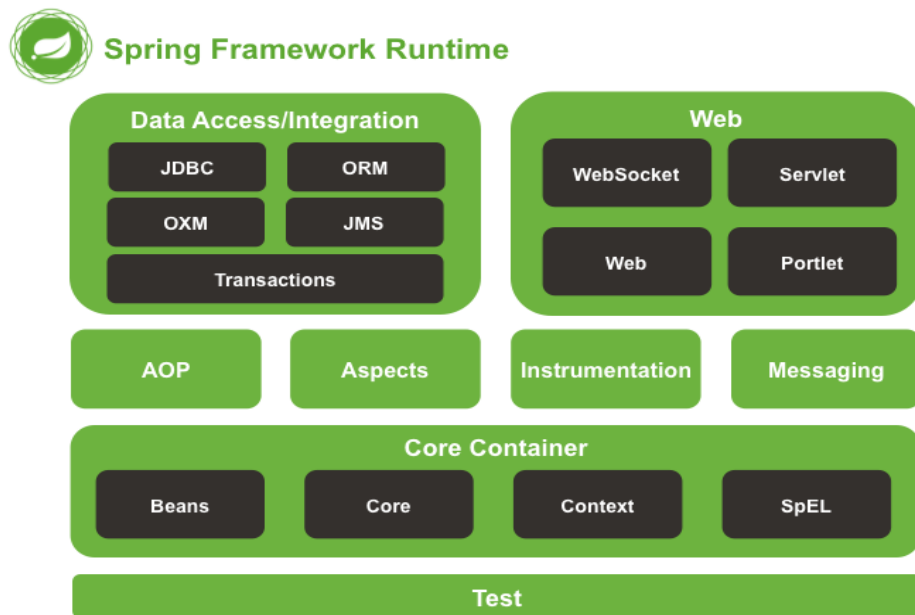
Đặc điểm nổi bật của Spring Boot

- **SpringApplication:** Đây là class được dùng để khởi chạy ứng dụng từ hàm `main()`. Vì thế, khi khởi chạy ứng dụng, các lập trình viên chỉ cần gọi `method run()` là được.
- **Externalized Configuration:** Spring Boot cho phép người dùng có thể sử dụng cấu hình (config) từ bên ngoài. Chính vì vậy mà mỗi ứng dụng được tạo ra có thể chạy được trên nhiều môi trường khác nhau.
- **Profiles:** Dùng để phân chia các loại cấu hình cho các môi trường khác nhau.

- Login: Sử dụng phục vụ cho toàn bộ chức năng log trong phạm vi nội bộ. Những logging này sẽ được quản lý một cách mặc định.

Ngoài ra còn rất nhiều các tính năng tương tự khác của Spring Boot như: Developing web Applications, Working with SQL Technologies, Security, Messaging, Caching, Calling rest Services with RestTemplate/WebClient, Sending Email, Validation,....

2.1.3.5. Mô hình Spring MVC

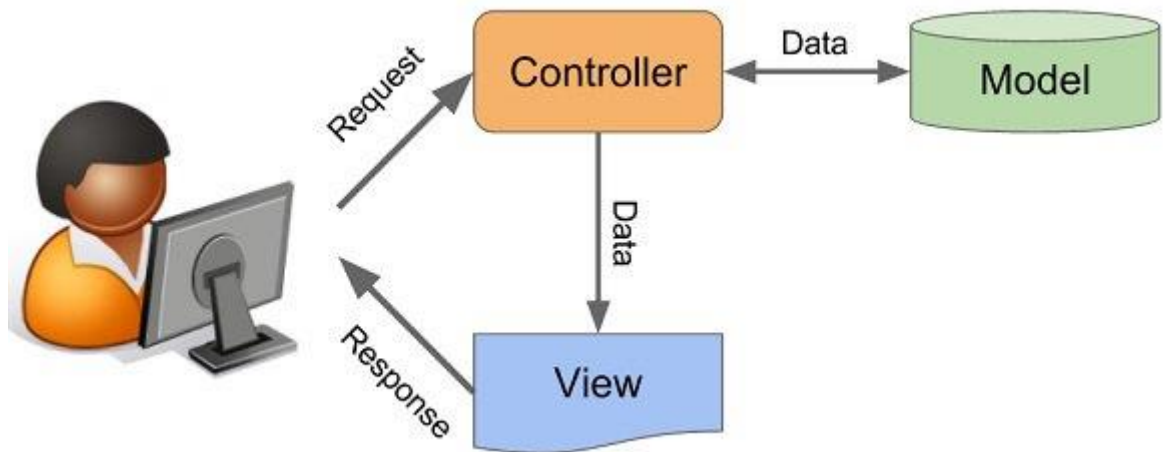


Hình 2.1.3 Các module của Spring MVC frameword thuộc tầng wed

- **Servlet** module chứa implemetation của Spring MVC trên web applications.
- **Web** module cung cấp các features và sự khởi tạo của IoC container sử dụng **servlet listeners** và **web application context**
- Web framework được build trên servlet API.
- Cung cấp kiến trúc Model-View-Controller (MVC)

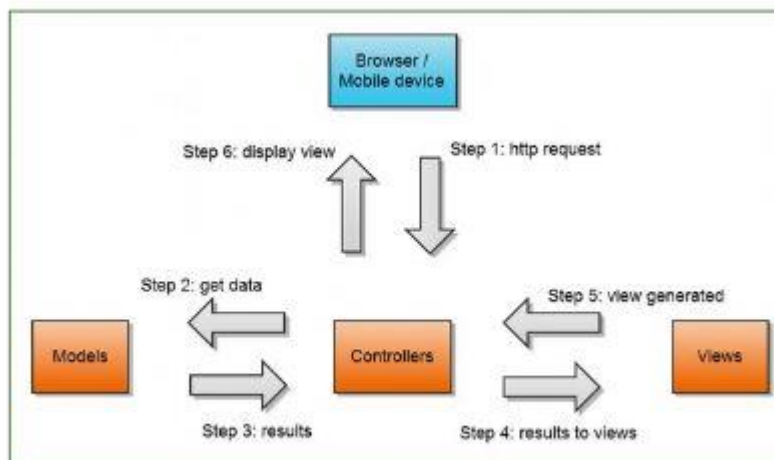
MVC (Model-View-Controller)

- MVC là 1 architectural pattern, được sử dụng phổ biến trong việc phát triển user interfaces
- 1 application được chia làm 3 phần: model, controller, view.



Hình 2.1.3 Mô hình MVC

- **Model:** Chịu trách nhiệm quản lý data
- **View:** Chịu trách nhiệm display data của model tới người dùng.
- **Controller:** Chịu trách nhiệm cho việc xử lý các requests của người dùng và build model thích hợp và truyền tới view



Bước 1: Người dùng sử dụng browser hoặc các thiết bị mobile để gửi http request tới **Controller**.

Bước 2: Controller sẽ yêu cầu lấy data từ **model**.

Bước 3: Controller nhận data từ **model**

Bước 4: Kết quả sẽ chuyển đến **view**

Bước 5: **View** sẽ được tạo ra và chuyển về **controller**

Bước 6: **Controller** sẽ display **view** tới người dùng.

Request Processing Workflow (High Level)

- **DispatcherServlet:** Spring MVC được thiết kế bao quanh 1 servlet trung tâm được gọi là **DispatcherServlet**. Nó hoạt động như 1 entry trung tâm của ứng dụng Spring MVC. Mỗi 1 request được xử lý bởi **DispatcherServlet**. Trong hình trên Front Controller chính là DispatcherServlet.
- **Controller:** Công việc của front controller là quyết định handler thích hợp. Có thể hiểu handler là những method trong Controller. Controller nào được chọn sẽ tương tác với tầng service, data tương đương sẽ được lưu trữ trong **model** và chuyển cho front controller. Khi controller kết thúc quá trình xử lý, Front controller sẽ quyết định **view** nào gửi đến render.
- **View:** Front controller sẽ truyền model tới view cái mà được chuyển tới render.

Request Processing Workflow (Details Level)

- **Chú thích:** Màu tím: Là những thành phần mà chúng ta tạo ra. Màu xanh dương: Là những thành phần được cung cấp bởi Spring Source. Màu xanh nõn chuối: Là những thành phần được cung cấp bởi Spring Source, nhưng có thể chỉnh sửa được.
- Quá trình xử lý request như sau:
 - **DispatcherServlet** sẽ nhận tất cả các request từ user.
 - Sau đó **dispatcherServlet** sẽ gửi task của **controller** thích hợp tới **HandlerMapping**. **HandlerMapping** sẽ chọn **controller**, cái mà map với với request url được nhận và trả về **Handler** và **controller** tới **dispatcherServlet**.
 - **DispatcherServlet** sẽ gửi task thực thi business logic của **controller** tới **HandlerAdapter**. (Ở đây, có thể hiểu **HandlerAdapter** như 1 người biên dịch translator, mục đích để **DispatcherServlet** và **Controller** có thể giao tiếp với nhau).
 - **HandlerAdapter** sẽ gọi các phương thức của **Controller**.
 - **Controller** sẽ thực thi business logic bằng cách thông qua tầng service, set processing result trong **model** và trả về tên của view cho **HandlerAdapter**
 - **DispatcherServlet** sẽ nhận tên view từ **HandlerAdapter** và gửi task giải quyết view cùng với tên view tới **ViewResolver**. **ViewResolver** sẽ trả về View tương ứng với tên view.
 - **DispatcherServlet** sẽ gửi kết quả nhận được từ **ViewResolver**, và gửi tới View.
 - View sẽ trả về model data và return response cho **DispatcherServlet** (Trong hình vẽ đang bị nhầm chút) và **DispatcherServlet** sẽ trả response cho người dùng.

WebApplicationContext

- DispatcherServlet mong chờ 1 WebApplicationContext, cái mà extend từ ApplicationContext (là 1 IoC container, có tác dụng tạo ra bean và quản lý vòng đời của chúng)
- Trong 1 Spring MVC Application, DispatcherServlet sẽ chứa tối thiểu 2 loại ApplicationContext là: Servlet và Root WebApplicationContext.
- **Trong đó:** Root WebApplicationContext sẽ nên chứa các Services và Repositories. Còn Servlet WebApplicationContext nên chứa các Controllers, ViewResolver, HandlerMapping.
- **Note:** Các thành phần trong Servlet WebApplicationContext có thể gọi các thành phần trong Root WebApplicationContext, nhưng ngược lại thì không cho phép. Ngoài ra, chúng ta có thể gộp các thành phần trong Servlet WebApplicationContext vào Root WebApplicationContext. Tuy nhiên, Servlet WebApplicationContext vẫn tồn tại nhưng ở trạng thái empty. Cách này, không được recommend do sẽ làm giảm quá trình xử lý.

Spring MVC Configurations

Spring MVC hỗ trợ 2 loại cấu hình là:

XML Configuration

- Java-based Configuration

2.2 Kiến thức cần thiết để học tập Spring Boot

2.2.1 Java Core

Trước khi học về Spring Boot, bạn cần biết về một số kiến thức cần biết của Java

- Java cơ bản: hàm, biến, vòng lặp,...
- Lập trình hướng đối tượng (OOP)
- Các tính năng của Java
- Collections API: biết cách sử dụng các collection thông dụng.

2.2.2 Package manager

Package manager được cài thêm để quản lý các thư viện khi code dự án Spring Boot. Bạn nên tìm hiểu cách dùng Maven và Gradle cơ bản.

Nếu bạn là người mới bắt đầu không nên tìm hiểu quá sâu, chỉ cần biết cách cài đặt, xóa thư viện, chỉnh sửa thông tin của project, các build-in tasks là được. Sau này khi sử dụng nhiều thì bạn tự khắc quen thôi.

2.2.3 *Spring Boot*

Cách học Spring Boot đúng đắn nhất chính là học thực hành, làm project. Nhiều thứ trong Spring Boot rất dễ, hoặc thành chuẩn mực rồi, chủ yếu là bạn đã từng đụng tới và thử áp dụng chưa thôi.

Bạn nên tìm hiểu nên học Spring Boot bắt đầu từ đâu, tạo dự án Spring Boot đầu tiên như thế nào, các luồng đi trong Spring Boot, Dependency injection áp dụng vào Spring Boot như thế nào, cấu trúc một dự án Spring Boot như thế nào cho chuẩn,...

2.2.4 *JPA/MongoDB/SQL*

Để tạo nên một chương trình, ứng dụng không thể thiếu được Database, do vậy các bạn cần tìm hiểu về:

- Cách cấu hình database trong Spring Boot
- Cấu trúc lớp và interface của JPA
- Tìm hiểu các thao tác cơ bản CRUD.
- Query creation
- Thực hiện các câu query phức tạp bằng MongoTemplate.
- Sắp xếp và phân trang data query được.
- Thực hiện aggregation

2.2.5 *Template engine*

Template engine đảm nhiệm xử lý phần View trong ứng dụng MVC. Trong Spring Boot chúng ta sử dụng template engine để paste dữ liệu vào View và trả về một trang HTML. Những kiến thức về cách trả về trang HTML, sử dụng Thymeleaf, JSP trong Spring Boot bạn cũng nên tìm hiểu trước.

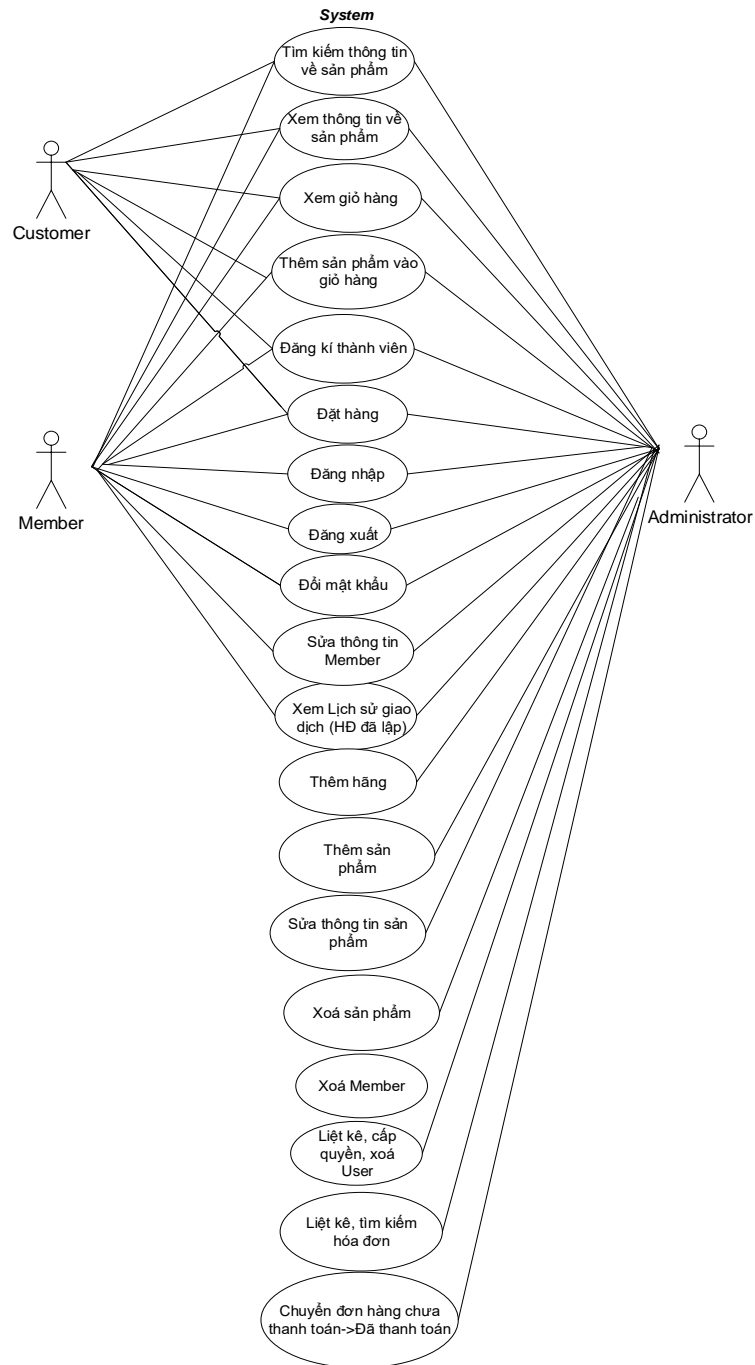
2.2.6 *Các Tool khác*

Ngoài ra, trong một dự án Spring Boot còn sử dụng các Tool khác các bạn có thể tìm hiểu về cấu hình và cách sử dụng để nâng cao hiệu suất viết code như:

- Cấu hình và sử dụng Swagger
- Thiết lập Logger cho project
- Lombok – viết code Java ngắn
- Cấu hình Spring Actuator
- Cấu hình Firebase
- Cấu hình Spring Boot devtools

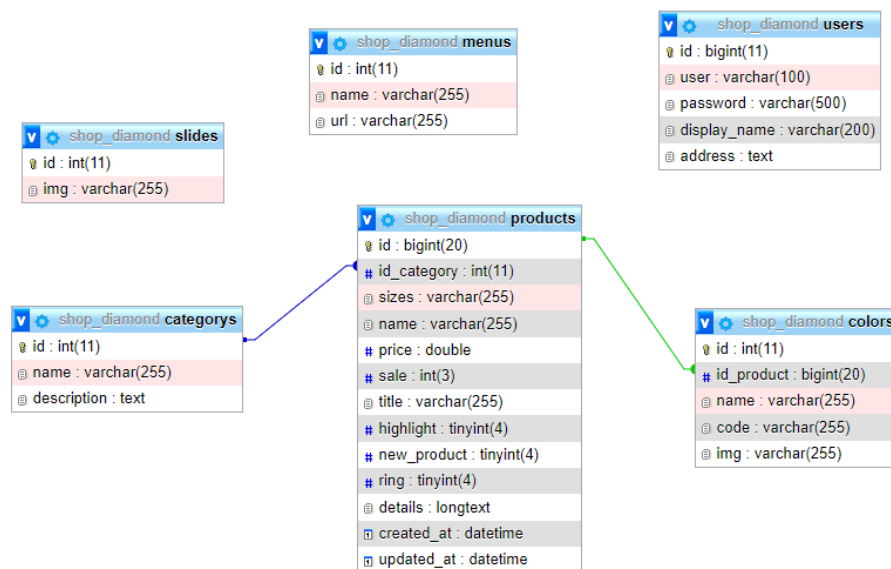
CHƯƠNG 3: CƠ SỞ DỮ LIỆU

3.1. Sơ đồ Usecase



Hình 3.1 Mô hình use-case

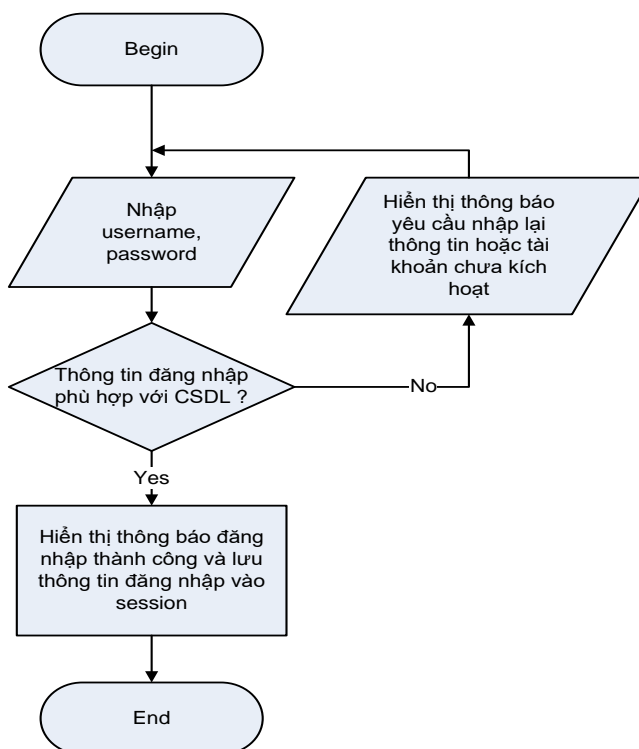
3.2. Sơ đồ Diagram



Hình 3.0.2 Sơ đồ Diagram

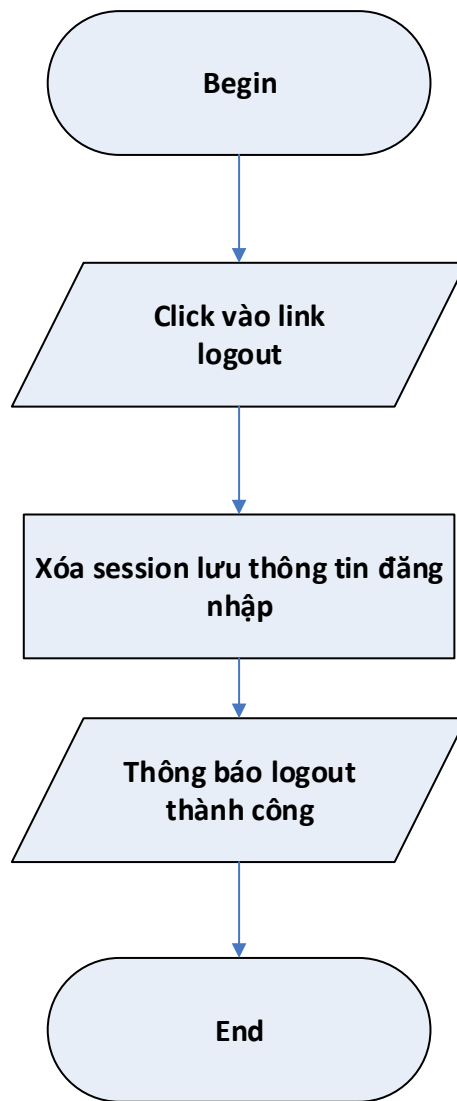
3.3. Biểu đồ hoạt động (Activity)

➤ Đăng nhập

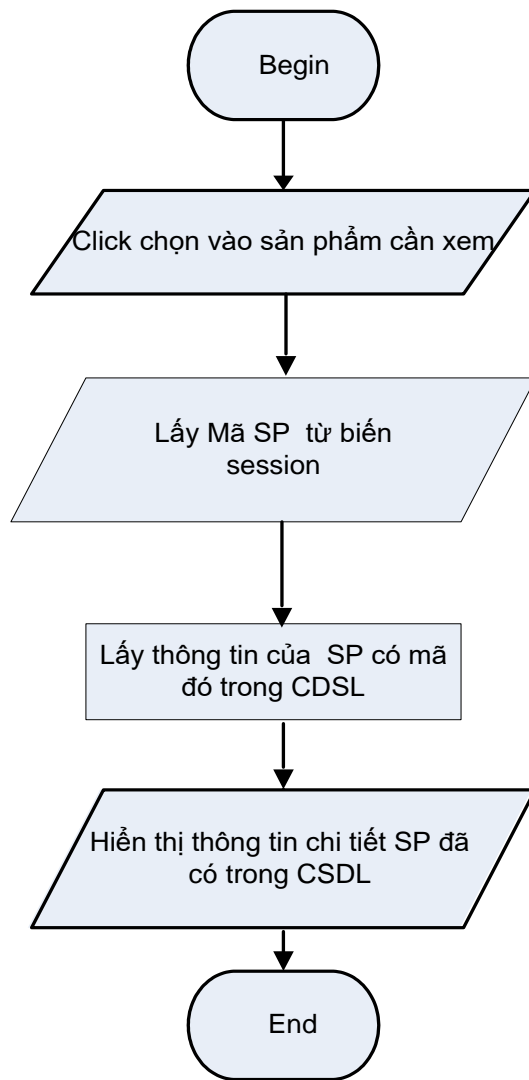


Hình 3.0.3 Biểu đồ hoạt động đăng nhập vào hệ thống

➤ Đăng xuất

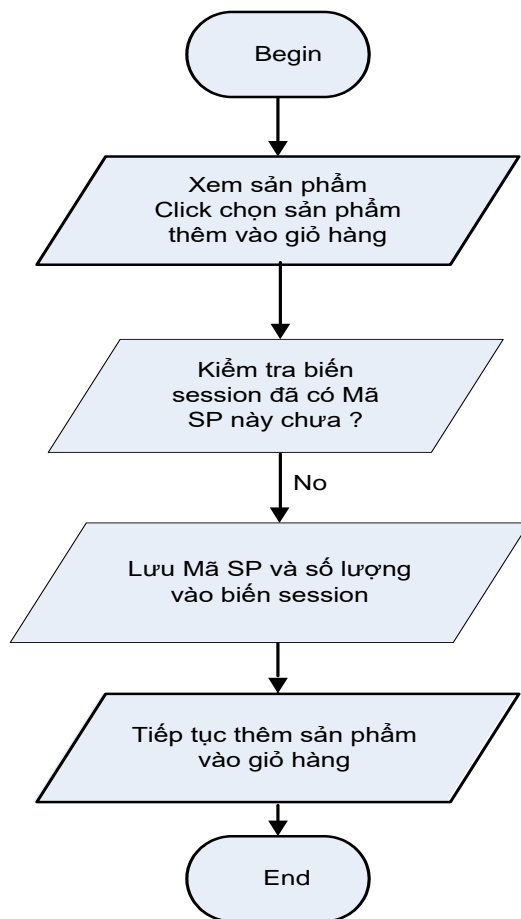


Hình 3.0.4 Biểu đồ hoạt động thoát ra khỏi hệ thống
Xem thông tin về sản phẩm



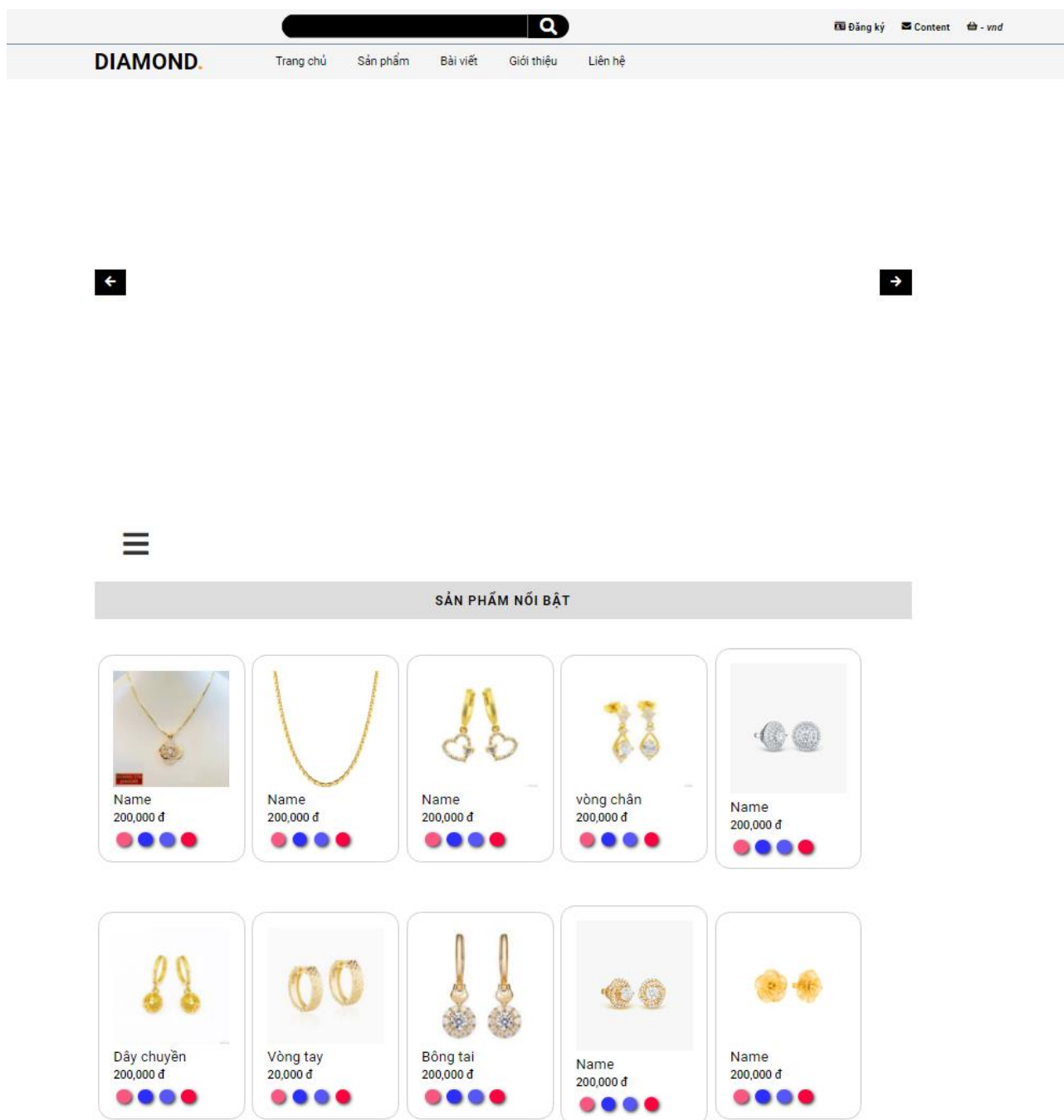
Hình 3.0.5 Biểu đồ hoạt động xem thông tin về sản phẩm

- Thêm sản phẩm vào giỏ hàng

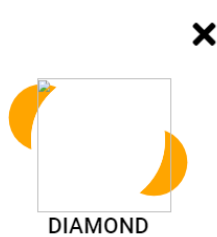


Hình 3.0.6 Biểu đồ chức năng thêm SP vào giỏ hàng

CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM



Hình 4.1 Trang chủ



Dây chuyền vàng

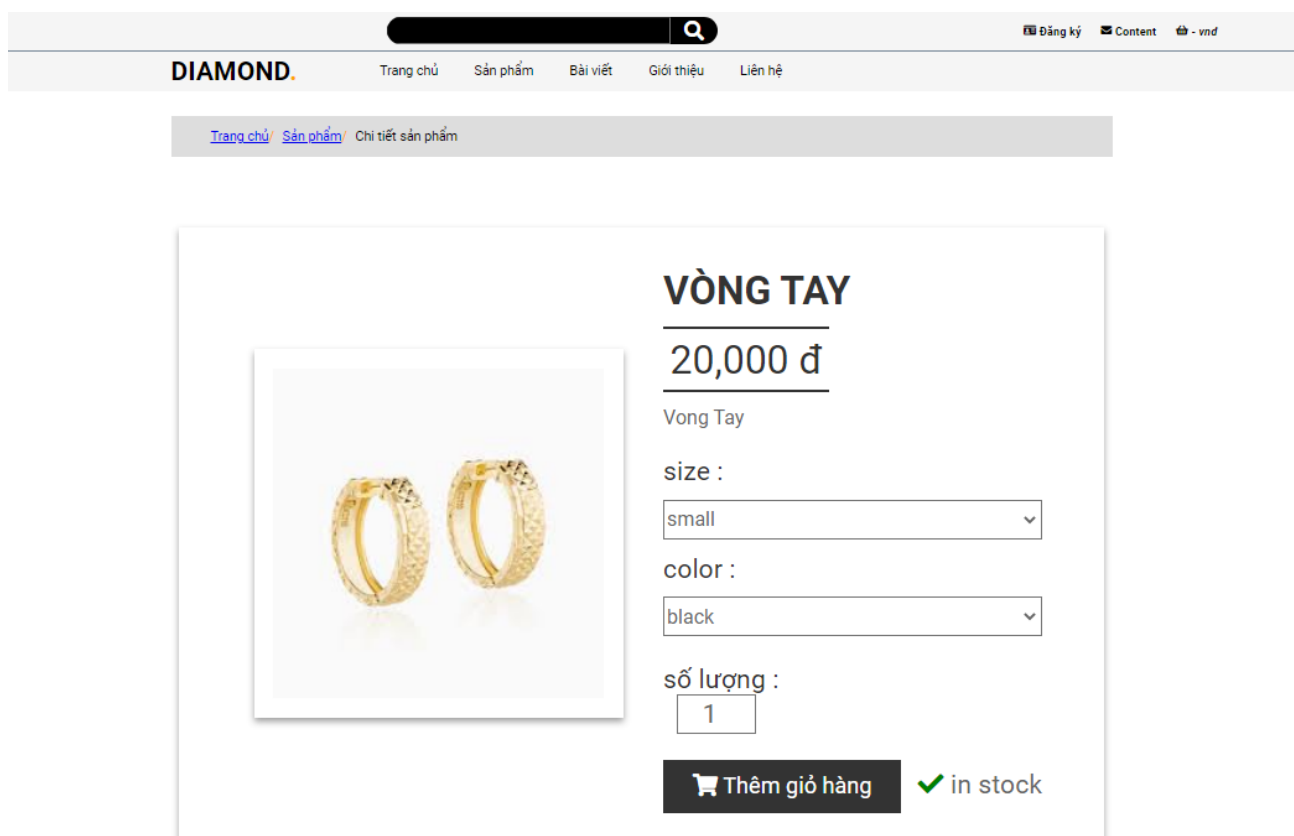
Nhẫn cưới

Bông tai

Vòng tay



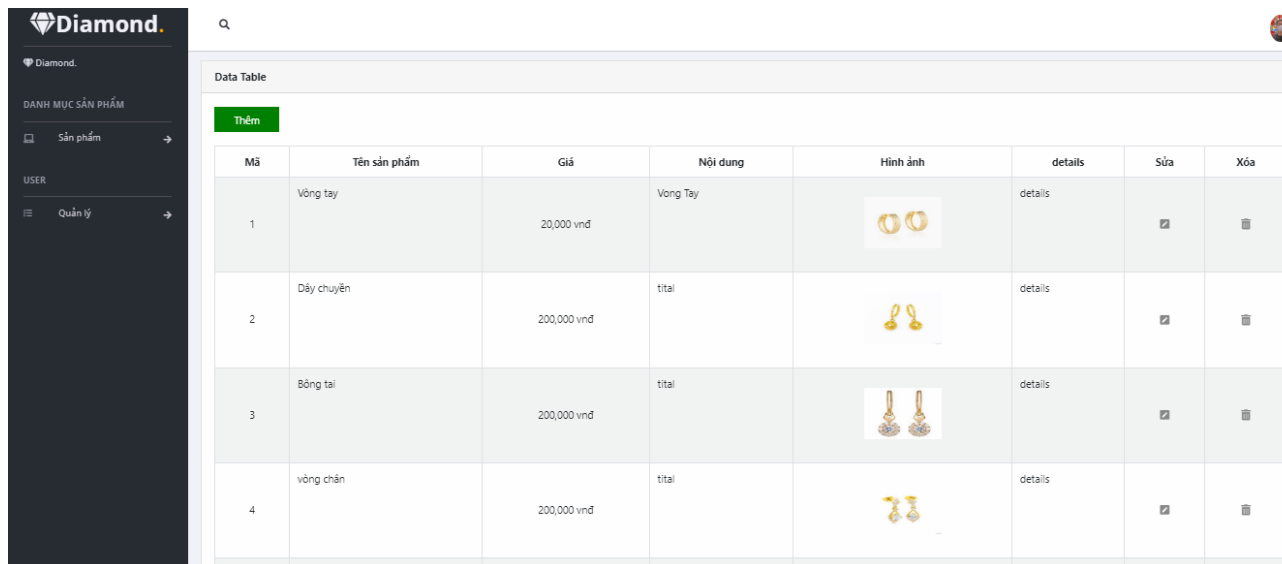
Hình 4.2 Danh mục



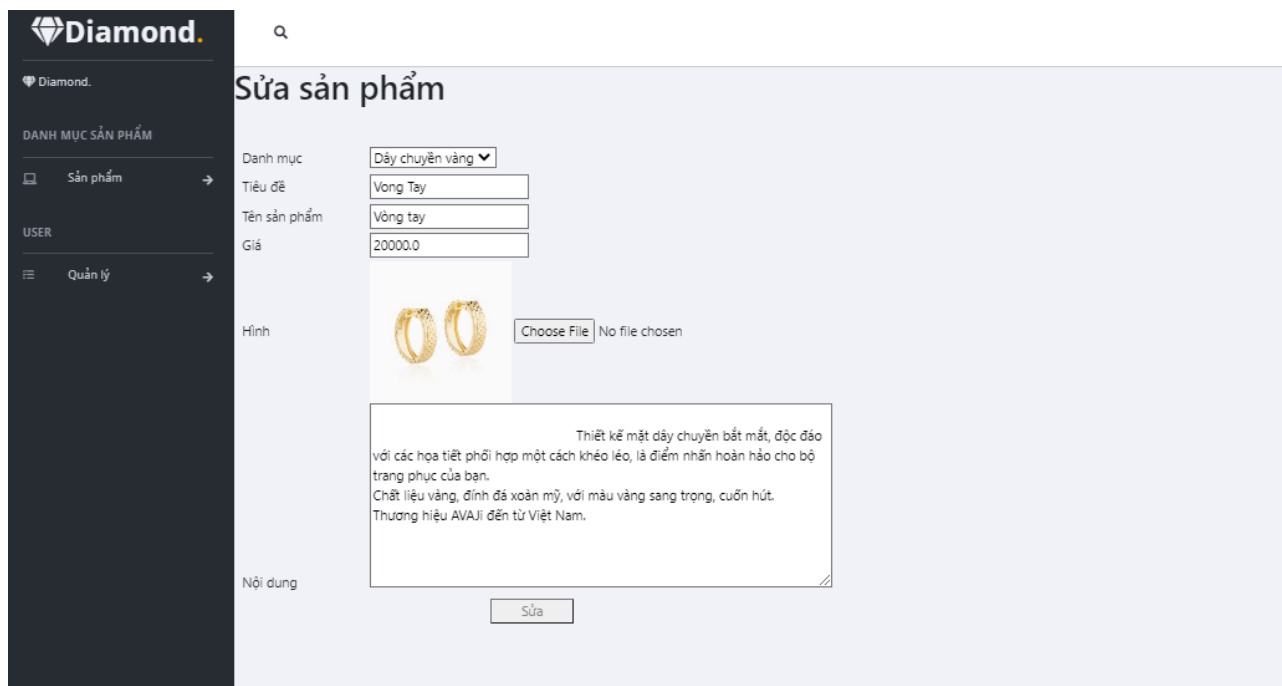
Hình 4.3 Chi tiết sản phẩm

Hình 4.4 Thanh toán


Hình 4.5 Đăng kí



Hình 4.6 Trang quản lý sản phẩm




Hình 4.0.7 Trang sửa sản phẩm


Diamond.

♥ Diamond.

DANH MỤC SẢN PHẨM

 Sản phẩm →

USER

 Quản lý →

🔍

Thêm sản phẩm

Danh mục

Dây chuyền vàng ▾

Tiêu đề

Tên sản phẩm

Giá

Hình

Choose File

No file chosen

Nội dung

Thêm

Hình 4.0.8 Trang thêm sản phẩm

CHƯƠNG 5: KẾT LUẬN VÀ KIẾN NGHỊ

5.1 Kết luận

5.1.1 Các chức năng đã hoàn thành

Hiện tại website đã hoàn thành được 70% so với kế hoạch đề ra. Phần mềm có thể đưa vào vận hành.

Nhóm vẫn còn nhiều thiếu sót, chưa thể hoàn thiện được hết các chức năng đề ra, phát triển thêm một số chức năng khác. Nhóm sẽ cố gắng hoàn thiện tốt nhất các phần đã làm.

Vận dụng được những kiến thức đã được học.

5.1.2 Các chức năng chưa làm được

Vì trong thời gian có giới hạn nên mức độ hoàn thiện của đồ án này không thể hoàn thành tốt 100%. Vì cần nhiều thời gian tìm hiểu một số chức năng khác.

Còn một số chức năng như phân loại khách hàng,.... dùng một số thuật toán đề gợi ý sản phẩm khách hàng muốn mua theo gợi ý của khách hàng.

5.2 Đánh giá website

5.2.1 Ưu điểm

Giao diện website thân thiện và ưa nhìn cho người sử dụng.

Chức năng đăng nhập nhanh chóng và dễ dàng, có bảo mật cho người sử dụng.

Hiển thị được các sản phẩm của cửa hàng.

Chức năng tìm kiếm sản phẩm theo phân loại hoặc theo tên.

Dễ dàng thanh toán và đặt hàng cho người sử dụng.

5.2.2 Nhược điểm

Việc nhập mã giảm giá để giảm giá sản phẩm chưa được.

Chỉ tích hợp web trên máy tính, chưa tích hợp trên các thiết bị khác.

Tính bảo mật chưa cao.

Chưa hiểu rõ nghiệp vụ của quá trình nhập kho của một cửa hàng.

5.3 Kiến nghị cho việc phát triển website

Xây dựng thêm một số tính năng thanh toán online khác.

Thêm các chức năng đăng nhập bằng các tài khoản mạng xã hội khác như Facebook, Instagram, Twitter, Zalo...

Thêm chức năng gửi mail tự động các thông tin khuyến mãi, sản phẩm mới hay sự kiện ưu đãi của cửa hàng.

Thêm chức năng tặng các voucher giảm giá cho khách hàng.

Tích hợp cho điện thoại và một số thiết bị khác.

Chức năng gợi ý sản phẩm cho khách hàng theo yêu thích, màu sắc, theo thói quen bằng cách tích hợp AI vào website.

Thêm một số chức năng cấp độ thân thiết cho khách hàng như: khách hàng thân thiết, khách hàng VIP...

TÀI LIỆU THAM KHẢO

- {1}[\(48\) SiVi CODE | Xây dựng Website bán hàng Spring MCV 22. Lấy dữ liệu chi tiết sản phẩm. - YouTube](#)
- {2}[GitHub - HuynhVanThuy/DiamonShopSpringMVC at Bai22](#)
- {3}[\(48\) Java Spring MVC and JDBC CRUD Tutorial \(Web App using Eclipse, Tomcat, MySQL and JUnit\) - YouTube](#)
- {4}[Java Spring MVC with JdbcTemplate Tutorial \(codejava.net\)](#)