**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**Quách Thuần Minh Triết**

**HỆ THỐNG MUA BÁN HÀNG ĐIỆN TỬ TRỰC TUYẾN**

**ĐỒ ÁN NGÀNH**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**TP. HỒ CHÍ MINH, 2024BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**Quách Thuần Minh Triết**

**HỆ THỐNG MUA BÁN HÀNG ĐIỆN TỬ TRỰC TUYẾN**

**Mã số sinh viên: 2151050473**

**ĐỒ ÁN NGÀNH**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Giảng viên hướng dẫn: Phan Trần Minh Khuê**

**TP. HỒ CHÍ MINH, 2024**

**LỜI CẢM ƠN**

Điều đầu tiên, em xin chân thành cảm ơn thầy Phan Trần Minh Khuê đã hỗ trợ em trong quá trình thực hiện đồ án. Đã cung cấp cho em những kiến thức thực tiễn trong quá trình thực hiện đồ án. Cũng như đã dành nhiều thời gian để tư vấn và cung cấp lời khuyên cho em thực hiện tốt đồ án.

Tiếp theo, em xin cảm ơn nhà trường đã giúp cho em có cơ hội học tập, đã truyền đạt những kiến thứ từ những môn học. Từ những kiến thức quý giá đó, đã tạo cho em một nền tảng kiến thức trong việc hoàn thành đồ án một cách thành công.

Tuy nhiên, do kiến thức, thời gian và sắp xếp việc học hạn chế em chưa tìm hiểu và phát huy hết toàn bộ ý tưởng đã được đưa ra và khả năng lập trình còn hạn chế nên không thể thực hiện hết những ý tưởng ban đầu. Trong quá trình xây dựng hệ thống không thể tránh khỏi những sai sót của bản thân, em rất mong nhận được sự đóng góp của thầy cô và các bạn.

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

**TÓM TẮT ĐỐ ÁN NGÀNH**

Đồ án ngành là **Hệ Thống Mua Bán Hàng Điện Tử Trực Tuyến** nhằm tạo ra một hệ thống cho doanh nghiệp và một website thu hút khách hàng. Hệ thống sẽ có những chức năng như là đăng nhập, đăng ký tài khoản, tìm kiếm các sản phẩm theo tên và xem các sản phẩm theo doanh mục của doanh nghiệp tại website. Với những khách hàng đã đăng nhập vào hệ thống sẽ có các chức năng như là quản lý thông tin cá nhân, chỉnh sửa thông tin cá nhân, thanh toán các mặt hàng đã thêm vào giỏ hàng và đánh giá các sản phẩm đã mua hoặc tham khảo các đánh giá khác. Với người quản lý hệ thống (nhân viên) sẽ có các chức năng như là thêm, sửa và xoá các sản phẩm, quản lý người dùng, kiểm tra các hoá đơn đã đặt của khách hàng, thông kê báo cáo, … Từ đó, doanh nghiệp có thể nắm bắt được tình trạng hiện tại của cửa hàng, nhằm đưa ra những chiến lược hợp lý cho công ty.

Website phải đảm bảo được thực hiện cách chức năng một cách dễ dàng và có một giao diện người dùng thân thiện dễ dàng thao tác. Để thực hiện những thứ trên, đã đưa ra những nền tảng là Django và Reactjs để thực hiện sự phát triển hệ thống. Nhằm đảm bảo trong việc lưu trữ dữ liệu và quản lý thông tin thì sẽ dùng nền tảng của MySQL để phát triển hệ thống.

Trên đây là những chức năng của hệ thống cùng với những ngôn ngữ lập trình và nền tảng để phát triển hoàn thành đồ án. Đây chỉ là một website bán hành của một doanh nghiệp nên sẽ có thể xảy ra sai sót trong quá trình phát triển. Trong tương lai, có thể sẽ bổ sung nhiều chức năng mới và hoàn thiện lại các chức năng cũ để có thể chạy mượt mà.

ABSTRACT

**MỤC LỤC**

[DANH MỤC TỪ VIẾT TẮT 9](#_Toc13182)

[DANH MỤC HÌNH VẼ 10](#_Toc9040)

[DANH MỤC BẢNG 11](#_Toc1705)

[MỞ ĐẦU 12](#_Toc17063)

[Chương 1. TỔNG QUAN ĐỒ ÁN 12](#_Toc8513)

[1.1. Lý do chọn đề tài: 12](#_Toc16294)

[1.2. Mục tiêu đề tài: 12](#_Toc8200)

[1.3. Phạm vi nghiên cứu: 12](#_Toc19869)

[1.4. Đối tượng nghiên cứu: 12](#_Toc5214)

[1.5. Phương pháp nghiên cứu: 13](#_Toc24207)

[1.5.1. Nghiên cứu thông qua phương pháp lý thuyết: 13](#_Toc15822)

[1.5.2. Nghiên cứu thông qua phương phát mô hình hoá: 13](#_Toc23307)

[1.5.3. Nghiên cứu thông qua phương pháp tiếp cận hệ thống: 13](#_Toc13593)

[1.6. Ý nghĩa thực tiễn của đề tài: 13](#_Toc10638)

[1.7. Kết cấu của báo cáo: 14](#_Toc15488)

[Chương 2. CƠ SỞ LÝ THUYẾT 15](#_Toc2603)

[2.1. Django: 15](#_Toc18258)

[2.1.1. Khái niệm: 15](#_Toc12374)

[2.1.2. Tính năng nổi bậc: 15](#_Toc6930)

[2.2. Django REST Framework (DRF): 15](#_Toc5109)

[2.2.1. Khái niệm: 15](#_Toc16053)

[2.2.2. Tính năng nổi bậc: 16](#_Toc31793)

[2.3. Oauth2: 16](#_Toc20664)

[2.3.1. Khái niệm: 16](#_Toc6557)

[2.3.2. Thành phần chính: 16](#_Toc25392)

[2.3.3. Quy trình hoạt động: 16](#_Toc29346)

[2.3.4. Lợi ích mang lại: 17](#_Toc13561)

[2.4. ReactJS: 17](#_Toc15453)

[2.4.1. Khái niệm: 17](#_Toc6570)

[2.4.2. Tính năng nổi bậc: 17](#_Toc19273)

[2.5. Axios: 18](#_Toc11781)

[2.5.1. Khái niệm: 18](#_Toc25204)

[2.5.2. Tính năng nổi bậc: 18](#_Toc9458)

[2.6. MySQL: 18](#_Toc7432)

[2.6.1. Khái niệm: 18](#_Toc6472)

[2.6.2. Tính năng nổi bật: 19](#_Toc19823)

[Chương 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG: 20](#_Toc29429)

[3.1. Phân tích hệ thống: 20](#_Toc18020)

[3.1.1. Lược đồ use case: 20](#_Toc8767)

[3.1.2. Đặc tả use case: 20](#_Toc26023)

[3.1.2.1. Use case đăng ký đăng nhập: 20](#_Toc18997)

[3.1.2.2. Use case thanh toán hoá đơn: 22](#_Toc13614)

[3.1.2.3. Use case theo dõi đơn hàng: 23](#_Toc31627)

[3.1.2.4. Use case thống kê báo cáo: 24](#_Toc24331)

[3.1.2.5. Use case đánh giá sản phẩm: 25](#_Toc27510)

[3.2. Thiết kế hệ thống: 26](#_Toc20834)

[Chương 4. THẾT KẾ HỆ THỐNG: 27](#_Toc11844)

[TÀI LIỆU THAM KHẢO 27](#_Toc6943)

[PHỤ LỤC 28](#_Toc31674)

DANH MỤC TỪ VIẾT TẮT

DANH MỤC HÌNH VẼ

[Hình 1 : Lược đồ use case 20](#_Toc18225)

DANH MỤC BẢNG

[Bảng 1 : Use Case Đăng Nhập 21](#_Toc29583)

[Bảng 2 : Use Case Đăng Nhập 22](#_Toc19056)

[Bảng 3 : Use Case Thanh Toán Hoá Đơn 23](#_Toc19571)

[Bảng 4 : Use Case Theo Dõi Đơn Hàng 24](#_Toc18409)

[Bảng 5 : Use Case Thống Kê Báo Cáo 25](#_Toc19432)

[Bảng 6 : Use Case Đánh Giá Sản Phẩm 26](#_Toc16395)

MỞ ĐẦU

1. **TỔNG QUAN ĐỒ ÁN**
   1. **Lý do chọn đề tài:**

Trong Thời đại Thông tin, việc mua sắm mặt hàng đồ điện tử ngày nay đã trở thành vấn đề được quan tâm và càng ngày càng phổ biến hơn ở Việt Nam. Việc mua sắm trực tuyến trở nên cần thiết và đem lại nhiều lợi ích việc trội về thời gian, chi phí, tính đa dạng, thông tin chi tiết các sản phẩm và dịch vụ chăm sóc khách hàng đã trở thành những thứ tạo sự hấp dẫn đối với người dùng.

Từ những lợi ích và cơ hội trên, nếu muốn cạnh tranh trong môi trường thương mại hiện nay phải trang bị cho bản thân doanh nghiệp một trang web để dễ dàng tiếp xúc với người dùng. Do đó, em đã quyết định xây dựng một “Hệ thống mua bán hàng điện tử trực tuyến” nhằm tạo ra một hệ thống quản lý nhu cầu mua bán hàng có tính thực tiễn và có thể ứng dụng vào thực tế.

* 1. **Mục tiêu đề tài:**

Đề tài tập trung trong phát triển website mua bán hàng điện tử trực tuyến, nhằm đáp ứng nhu cầu mua sắm hàng trực tuyến trong thời đại phát triển hiện nay. Cũng như sẽ mang lại những trải nghiệm mua sắm mà còn sẽ được tập trung giải quyết nhu cầu trong việc quản lý kinh doanh. Nhằm nâng cao khả nâng cạnh tranh thị trường thương mại của Việt Nam.

Một số mục tiêu phải đạt được khi phát triển hệ thống, thực hiện được đăng nhập, đăng ký tài khoản vào hệ thống, tìm kiếm sản phẩm theo tên, có thể mua những mặt hàng đã đặt trong giỏ hàng và chi tiết đơn hàng. Có thể theo dõi trạng thái đơn hàng sau khi đặt.

* 1. **Phạm vi nghiên cứu:**

Đồ án được tập trung xây dựng dựa trên các mặt hàng ở Cellphone S (cellphones.com.vn), FPT (fptshop.com.vn), Phong Vũ (phongvu.vn), …

Giao diện của hệ thống được tập trung phát triển trên website bán hàng của Phong Vũ.

* 1. **Đối tượng nghiên cứu:**

Các website nổi tiếng như là FPT, CellphoneS, phongvu,… nhằm để phát triển từ UX/UI đến các chức năng của hệ thống.

* 1. **Phương pháp nghiên cứu:**
     1. **Nghiên cứu thông qua phương pháp lý thuyết:**

Phương pháp lý thuyết là phương pháp thu thập ý kiến từ những bài báo thông qua việc đọc, các tài liệu liên quan đến tình trạng hiện nay. Từ đó, sẽ chọn ra các khái niệm và tư tưởng nhằm tạo tăng hiệu quả trong việc thực hiện đồ án. Thông qua những thông tin thu thập được có thể hỗ trợ trong việc pháp triển hệ thông và thực hiện bài báo cáo đồ án.

* + 1. **Nghiên cứu thông qua phương phát mô hình hoá:**

Phương pháp mô hình hoá hệ thống là phương pháp nghiên cứu cho phép chúng ta xây dựng một mô hình gần giống với hệ thống. Quá trình sẽ chia ra làm hai giai đoạn chính là mô hình hoá hệ thống và thực nghiệm những tác vụ trên mô hình ấy. Từ đó, cho chúng ta hiểu rõ hơn về hệ thông bán hàng điện tử thông qua một số hình ảnh giả lập sẽ hoạt động như thế nào, cũng như từ đó đưa ra những vấn đề bất cập cần được giải quyết trong quá trình thực hiện hệ thống một cách thực tế và hiệu quả.

* + 1. **Nghiên cứu thông qua phương pháp tiếp cận hệ thống:**

Phương pháp tiếp cận hệ thống là phương pháp tiếp cận nghiên cứu các đối tượng đang dùng hệ thống để tìm ra được những vấn đề nên được giải quyết. Cũng như, sẽ nổ lực giải quyết các vấn đề liên quan đến chất lượng hệ thống từ giao diện cho đến cơ sở dữ liệu. Để thoả mãn những nhu cầu đó, cần được sự chấp thuận của khách hàng cũng như là các doanh nghiệp sở hữu hệ thống nhằm giải quyết chính xác yêu cầu và sự chấp thuận từ các bên liên quan.

* 1. **Ý nghĩa thực tiễn của đề tài:**

Đối với khách hàng: Tạo ra một website giúp cho những khách hàng có nhu cầu cấp thiết về việc mua hàng nhanh chóng tại cửa hàng và không khó khăn so với những người có độ tuổi khách nhau. Giúp cho khách hàng dễ dàng so sánh giá cả những sản phẩm mình muốn mua. Tiết kiệm nhiều chi phí và thời gian cho khách hàng.

Đối với người sở hữu hệ thống: Tạo ra một website giúp cho người sở hữu có thể dễ dàng quản lý doanh thu, bán những món hàng khó chưng bày tại tiệm, quản lý những mặt hàng người dùng và quản lý những khách hàng tiềm năng của của hàng. Từ đó, dễ dàng đặt ra những kế hoạch cho tương lai.

* 1. **Kết cấu của báo cáo:**

1. **CƠ SỞ LÝ THUYẾT**
   1. **Django:**
      1. **Khái niệm:**

Django là một trong những framework web nổi tiếng bậc cao, mã nguồn mở, được lập trình trên nền tảng ngôn ngữ Python. Django hỗ trợ các nhà lập trình xây dựng hệ thống ứng dụng web một cách nhanh và hiệu quả nhờ vào việc cung cấp rõ các module có thể tái sử dụng nhiều lần. Django sử dụng mô hình MTV (Model-Template-Views) để tổ chức và phát triển các ứng dụng web phức tạp và không cần viết đi viết lại nhiều lần.

* + 1. **Tính năng nổi bậc:**

**- ORM (Object-Relational Mapping)**: Django cung cấp một hệ thống ORM mạnh mẽ, cho phát người lập trình làm việc dễ dàng với cơ sở dữ liệu bằng Python thay vì dùng những câu lệnh thủ công của MySQL.

**- Admin Interface:** Django sẽ hỗ trợ người dùng bằng cách tự tạo ra một website admin dễ nhìn và thao thác cho người dùng. Từ đó, giúp người dùng dễ dàng quản lý dữ liệu mà không cần viết thêm mã.

**- Authentication:** Để quản lý các tính năng như đăng ký, đăng nhập và quản lý phiên làm việc của người dùng hệ thống đã được Django tích hợp sẵn một mô hình nhầm giải quyết.

**- Middleware:** Django sử dụng middleware để xử lý các yêu cầu và phản hồi của HTTP, cho phép người lập trình sử dụng thêm nhiều chức năng bảo mật và quản lý dễ dàng.

**- Security:** Đã được Django tích hợp sẵn những mô hình tính năng bảo mật nhằm giúp người sử dụng hệ thống và người lập trình tránh khỏi những lỗ hỏng phổ biến hiện nay.

**- Form Handling:** Django cung cấp những công cụ vô cùng mạnh mẽ để xử lý các biểu mẫu, xác thực và xử lý các lỗi xảy ra trong hệ thống của người dùng.

* 1. **Django REST Framework (DRF):**
     1. **Khái niệm:**

DRF là một thư viện mạnh mẽ của Django mang nhiều tính chất linh hoạt. Nhằm hỗ trợ người lập trình phát triển hệ thống ứng dụng web API nhanh chóng. DRF cung cấp những công cụ và tính năng để xây dựng các RESTful API, cho phép sự giao tiếp giữa các client khách hàng và chủ sở hữu thông qua các định dạng dữ liệu như JSON và XML.

* + 1. **Tính năng nổi bậc:**

**- Serializer:** Chuyển đổi dữ liệu giữa các kiểu Python và JSON, giúp dễ dàng kiểm soát trong việc gửi và nhận dữ liệu của hệ thống.

**- Viewsets:** Cung cấp cho hệ thống các phương thức xử lý HTTP (GET, POST, PUT, DELETE) hiệu quả.

**- Authentication và Permissions:** Hỗ trợ nhiều phương thức xác thực khác nhau và kiểm soát các quyền truy cập, hỗ trợ mạnh mẽ trong việc bảo vệ API để đảm bảo được an toàn.

**- Pagination:** Hỗ trợ phân trang đối với những kết quả trả về nhằm tối ưu hiệu quả khi làm việc với những tập tin và cơ sở dữ liệu lớn.

- **Giao diện thử nghiệm:** Hệ thống cung cấp sẵn cho lập trình viên một giao diện website nhằm thử nghiệm các API, giúp cho việc kiểm tra và phát triển một cách dễ dàng.

* 1. **Oauth2:**
     1. **Khái niệm:**

Oauth2 là giao thức giúp cho việc truy cập vào ứng dụng web thông qua tài khoản người dùng mà không cần thiết phải chia sẻ mật khẩu. Được áp dụng nhiều vào trong các ứng dụng web và di động nhằm tăng tính bảo mật thông tin của người dùng trong quá trình sử dụng.

* + 1. **Thành phần chính:**

- **Resource Owner:** Người dùng sở hữu dữ liệu mà hệ thống muốn truy cập.

- **Resource Server:** Máy chủ có thể lưu lại toàn bộ những dữ liệu cần thiết.

- **Client:** Giao diện hệ thống muốn truy cập vào dữ liệu.

- **Authorization Server:** Máy chủ xử các tác vụ như uỷ quyền, cấp mã truy cập cho client.

* + 1. **Quy trình hoạt động:**

**- Authorization Request:** Client yêu cầu quyền truy cập thông qua Authorization Server từ Resource Owner lên hệ thống.

**- Authorization Response:** Resource Owner phải đồng ý yêu cầu Authorization Server mới có thể cung cấp một mã uỷ quyền (authorization code). Resource Owner có thể từ chối yêu cầu.

**- Token Request:** Client sẽ dùng mã uỷ quyền đã được cấp để yêu cầu mã truy cập từ Authorization Server.

**- Token Response:** Authorization Server sẽ trả về mã truy cập sau khi xác thực mã uỷ quyền mà client đã gửi.

**- Access Resource:** Client sử dụng mã truy cập được cấp để sử dụng tài nguyên của hệ thống trên Resource Server.

* + 1. **Lợi ích mang lại:**

**- Bảo Mật:** Người dùng không cần thiết phải chia sẻ tài khoản với client.

**- Quyền Kiểm Soát:** Người dùng có thể kiểm soát và quản lý được các hoạt động truy cập dễ dàng.

**- Tính linh hoạt:** Hỗ trợ nhiều phương thức xác thực khác nhau trên nhiều ứng dụng khác nhau.

* 1. **ReactJS:**
     1. **Khái niệm:**

ReactJS có tên gọi chính thức là React theo như trang web chính thức thông báo. Tuy nhiên, ReactJS là thuật ngữ được cộng đồng sử dụng nhằm phần biệt với React Native (dùng để phát triển mobile). ReactJS là một thư viện Javascipt mã nguồn mở, được xây dựng và phát triển bởi Facebook vào 2013, được xây dựng giúp đỡ người lập trình phát triển toàn diện về giao diện (front-end) cho website. Không can thiệp vào các logic và cấu trúc đã được lập trình của hệ thống, nhằm tạo ra tính dễ dàng trong việc tích hợp vào các ứng dụng web.

* + 1. **Tính năng nổi bậc:**

**- Component:** Để dễ dàng trong việc xử lý và tái sử dụng mã code nhiều lần khi cần. ReactJS đã tổ chức một thống mà các giao diện thành phần của website trở thành từng thành phần độc lập với nhau.

- **Virtual DOM** **(Document Object Model ảo):** Tuy là bản sao chép nhỏ của DOM, nhưng vẫn đảm bảo tối ưu hiệu suất thay vì cập nhật toàn bộ DOM cũng như thay đổi vài thành phần cần thiết, giúp hệ thống tăng tốc độ thực thi.

**- JSX (Javascript XML):** là một cú pháp cho phép người lập trình viết HTML trong Javascript. Giúp việc xây dựng và phát triển đơn giản dễ hiểu. Không chỉ vậy mà còn tối ưu code khi được viết giúp cho JSX chạy nhanh hơn so với code Javascript thông thường.

- **Single-Way Data Flow:** Thông qua các props có trong ReactJS giúp cho ReactJS có thể nhận được các dữ liệu từ cha truyền xuống con, cho phép thực hiện các chức năng và các thành phần dễ dàng trao đổi dữ liệu với nhau.

* 1. **Axios:**
     1. **Khái niệm:**

Axios là một thư viện của Javascript được xây dựng dựa trên nền tảng Javascipt Promise để thực hiện yêu cầu HTTP gửi đến. Được sử dụng phổ biến trong các ứng dụng front-end như React, Vue,… Nhằm sử dụng để thực hiện các request HTTP bất đồng bộ của các REST endpoint. Dễ dàng sử dụng vì có nhiều tính năng hữu ích, giúp thực hiện các yêu cầu một cách dễ dàng.

* + 1. **Tính năng nổi bậc:**

**- Hỗ Trợ Promise:** việc xử lý các yêu cầu bất đồng bộ trên hệ thống trở nên dễ dàng và hiệu quả hơn thông qua axios. Bằng cách đó, có thể xử lý các yêu cầu và phản hồi từ server trở nên linh hoạt, không gây ra vấn đề về luồng của ứng dụng website.

**- Xử lý dữ liệu:** Có thể gửi và nhận nhiều loại dữ liệu khác nhau. Không những vậy còn hỗ trợ xử lý dữ liệu định dạng khác nhau thành kiểu dữ liệu Javascript như là đối tượng hoặc mảng.

**- Interceptors:** Là một bức tường chắn các request, response của ứng dụng web nhằm thêm hoặc thay đổi các logic tuỳ chỉnh trên header hoặc param.

**- Error Handling:** Khi xảy ra lỗi axios sẽ cung cấp phương thức để dễ dàng nhận biết vã xử lý lỗi từ server.

**- Cấu hình tuỳ chỉnh:** Cho phép sửa chữa và điều chỉnh các tham số của yêu cầu HTTP theo tuỳ yêu cầu của ứng dụng như params, data, headers,…

* 1. **MySQL:**
     1. **Khái niệm:**

MySQL là một hệ thống quản lý dữ liệu mã nguồn mở, cho phép người lập trình có thể xây dựng và phát triển dữ liệu dễ dàng. Sử dụng ngôn ngữ truy vấn của SQL tương tác với dữ liệu. Nó thường được ứng dụng vào một số hệ quản trị cơ sở dữ liệu phổ biến trên thế giới. Được ưa chuộng sử dụng do tính linh hoạt, hiệu suất cao và sự nhất quán.

* + 1. **Tính năng nổi bật:**

**- Mã nguồn mở:** MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở, cho phép người dùng tải về và phát triển theo nhu cầu của bản thân một các tự do. Giúp cho các chi phí sử dụng giảm bớt.

**- Hiệu suất cao:** MySQL có thể xử lấy hàng triệu câu lệnh truy vấn khác nhau một cách nhanh chóng và hiệu quả nhằm đem lại sự tối ưu cao trong hiệu suất sử dụng. Giúp cho việc trải nghiệm của nhiều khách hàng trong quá trình sử dụng cùng một lúc sẽ không gặp phải sự chậm trễ .

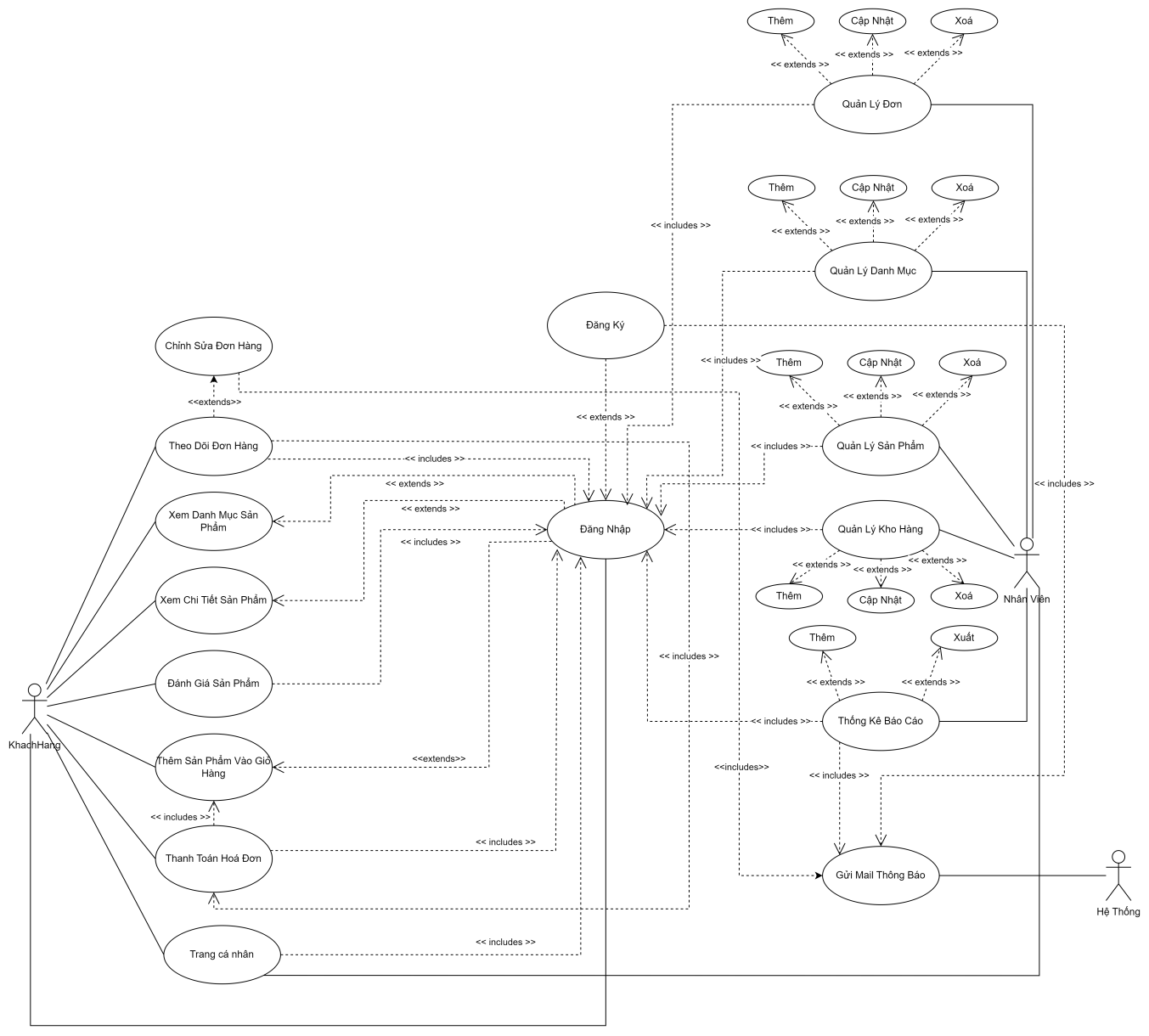
**- Khả năng mở rộng:** Cho phép người dùng xây dựng một cơ sở dữ liệu đơn giản sau đó phát triển lên cơ sở dữ liệu trở nên phức tạp trong hệ thống nếu cần thiết. Để tránh việc thay đổi nền tảng cơ sở dữ liệu MySQL đã hỗ trợ việc quản lý dễ dàng cùng với việc xử lý một khối lượng lớn dữ liệu.

**- Hỗ trợ nhiều loại dữ liệu:** MySQL cho phép người lập trình có thể làm việc hoặc sử dụng đầy đủ các kiểu dữ liệu trong JSON một cách thuận tiện và dễ dàng. Từ đó, có thể xây dựng và phát triển hệ thống dễ dàng trong việc thiết kế dữ liệu và lưu trữ thông tin theo nhu cầu khác nhau trong việc sử dụng.

**- Bảo mật:** MySQL có tích hợp nhiều chức năng xác thực người dùng nhằm nâng cao sự bảo mật của dữ liệu. Phân quyền trong việc truy cập chi tiết và mã hoá dữ liệu nhằm tránh khỏi các đe doạ và truy cập trái phép để có thể lấy những thông tin cá nhân hoặc thông tin nhạy cảm, tạo ra một môi trường an toàn và bảo mật.

**- Hỗ trợ giao dịch:** Được hỗ trợ theo nguyên tắc ACID (Atomicity, Consistency, Isolation, Durability), đảm bảo chắc chắn rằng mọi giao dịch được thực hiện hoặc không thực hiện gì cả. Giúp người lập trình có thể đảm bảo toàn vẹn thông tin dữ liệu, nhằm tránh ra khỏi đa số sai sót thường mắc phải trong các tác vụ phức tạp.

1. **PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG:**
   1. **Phân tích hệ thống:**
      1. **Lược đồ use case:**



Hình 1: Lược đồ use case

* + 1. **Đặc tả use case:**
       1. **Use case đăng ký đăng nhập:**

|  |  |
| --- | --- |
| **Use case ID** | UC001 |
| **Tên use case** | Đăng ký |
| **Mô tả vắn tắt Use Case** | - Use case cho phép đăng ký tài khoản mới thông qua giao diện để sử dụng các tính năng khác. |
| **Actor chính** | Khách Hàng |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | - Người dùng truy cập vào hệ thống và vào trang đăng ký. |
| **Hậu điều kiện** | - Người dùng đăng ký thành công và đăng nhập vào hệ thống. |
| **Luồng hoạt động** | **Bước 1:** Người dùng truy cập vào hệ thống trang đăng ký.  **Bước 2:** Điền thông tin ở trang đăng ký tài khoản.  **Bước 3:** Người dùng bấm nút đăng ký.  **Bước 4:** Hệ thống xác thực thông tin và lưu trữ vào cơ sở dữ liệu.  **Bước 5:** Hệ thống hiển thị đã đăng ký thành công và chuyển sang trang đăng nhập. |
| **Luồng thay thế** | - Người dùng huỷ đăng ký tài khoản và chuyển về trang chủ của hệ thống hoặc trang đăng nhập của hệ thống.  - Tại bước 2 nếu người dùng không điền đủ thông tin. Hệ thống sẽ thông báo trên giao những thông tin bị thiếu.  - Tại bước 4 sẽ xác thực những thông tin đã tồn tại trong hệ thống như tên tài khoản và gmail nếu trùng lặp thì hệ thống sẽ thông báo những thứ bị trùng trên giao diện. |
| **Luồng ngoại lệ** | - Tại bước 4 nếu xảy ra lỗi trong quá trình lưu trữ và xác thực thông tin hệ thống sẽ thông báo lỗi và chuyển về trang chủ. |

Bảng 1: Use Case Đăng Nhập

|  |  |
| --- | --- |
| **Use case ID** | UC002 |
| **Tên use case** | Đăng nhập |
| **Mô tả vắn tắt Use Case** | - Use case cho phép đăng nhập tài khoản của bản thân thông qua giao diện để sử dụng các tính năng khác. |
| **Actor chính** | Khách Hàng |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | - Người dùng truy cập vào hệ thống và vào trang đăng nhập. |
| **Hậu điều kiện** | - Người dùng đăng nhập thành công vào hệ thống. |
| **Luồng hoạt động** | **Bước 1:** Người dùng truy cập vào hệ thống trang đăng nhập.  **Bước 2:** Điền thông tin ở trang đăng nhập tài khoản.  **Bước 3:** Người dùng bấm nút đăng nhập.  **Bước 4:** Hệ thống xác thực thông tin tại cơ sở dữ liệu.  **Bước 5:** Hệ thống hiển thị đã đăng nhập thành công và chuyển sang trang chủ. |
| **Luồng thay thế** | - Tại bước 2 nếu người dùng không điền đủ thông tin. Hệ thống sẽ thông báo trên giao những thông tin bị thiếu.  - Tại bước 4 sẽ xác thực thông tin đăng nhập có đúng với hệ thống hay không nếu không sẽ báo lỗi đăng nhập không thành công. |
| **Luồng ngoại lệ** | - Tại bước 4 nếu xảy ra lỗi trong quá trình lưu trữ và xác thực thông tin hệ thống sẽ thông báo lỗi và chuyển về trang chủ. |

Bảng 2: Use Case Đăng Nhập

* + - 1. **Use case thanh toán hoá đơn:**

|  |  |
| --- | --- |
| **Use case ID** | UC003 |
| **Tên use case** | Thanh toán hoá đơn |
| **Mô tả vắn tắt Use Case** | - Use case này hỗ trợ người dùng thanh toán hoá đơn trong giỏi hàng thông qua 2 hình thực online và trực tiếp. |
| **Actor chính** | Khách Hàng |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | - Người dùng phải đăng nhập vào hệ thống và truy cập trang giỏ hàng. |
| **Hậu điều kiện** | - Người dùng thanh toán thành công hoá đơn. |
| **Luồng hoạt động** | **Bước 1:** Người dùng truy cập vào hệ thống trang giỏ hàng.  **Bước 2:** Hệ thống hiển thị danh sách sản phẩm trong giỏ hàng chưa thanh toán.  **Bước 3:** Người dùng bấm nút thanh toán.  **Bước 4:** Hệ thống sẽ chuyển về những phương thức có sẵn.  **Bước 5:** Hệ thống hiển thị đã thanh toán thành công hoá đơn của bản thân. |
| **Luồng thay thế** | - Khách hàng muốn xoá sản phẩm trong giỏ hàng hệ thống sẽ xoá sản phẩm đó ra giỏ hàng. |
| **Luồng ngoại lệ** | - Tại bước 4 nếu xảy ra lỗi trong quá trình thanh toán và chuyển giao phương thức hệ thống sẽ thông báo lỗi và chuyển về trang chủ. |

Bảng 3: Use Case Thanh Toán Hoá Đơn

* + - 1. **Use case theo dõi đơn hàng:**

|  |  |
| --- | --- |
| **Use case ID** | UC004 |
| **Tên use case** | Theo dõi đơn hàng |
| **Mô tả vắn tắt Use Case** | - Use case cho phép người dùng theo dõi những đơn hàng đã đặt và những sản phẩm bên trong. |
| **Actor chính** | Khách Hàng |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | - Khách hàng đã đăng nhập vào hệ thống.  - Khách hàng đã có ít nhất một đơn hàng. |
| **Hậu điều kiện** | - Khách hàng có thể xem được thông tin chi tiết về đơn hàng đã chọn.  - Khách hàng có thể theo dõi sự thay đổi trạng thái của đơn hàng.  - Khách hàng có thể trả hàng và huỷ đơn hàng. |
| **Luồng hoạt động** | **Bước 1:** Người dùng truy cập vào hệ thống trang theo dõi đơn hàng.  **Bước 2:** Hệ thống hiển thị những đơn hàng đã đặt.  **Bước 3:** Người dùng bấm nút xem chi tiết.  **Bước 3.1:** Người dùng bấm nút huỷ đơn hàng trong giai đoạn xử lý.  **Bước 3.2:** Hệ thống sẽ gửi mail về cho nhân viên xác nhận huỷ đơn.  **Bước 3.3:** Người dùng bấm nút hoàn trả sau khi nhận hàng.  **Bước 3.4:** Hệ thông sẽ gửi mail về cho người dùng đã gửi yêu cầu cho nhân viên và nhân viên sau khi nhận mail gửi trả thì xác định thông tin và chấp nhận hoàn trả.  **Bước 3.5:** Hệ thống sẽ gửi mail về cho khách hàng đã xác nhận hoàn trả và gửi shipper đến lấy hàng.  **Bước 4:** Hệ thống hiển thị toàn bộ thông tin sản phẩm đã được mua trong đơn hàng.  **Bước 5:** Đơn hàng sẽ được hiển thị dưới dạng thời gian thực như là đang xử lý, đang giao, đã nhận và đã hoàn trả. |
| **Luồng thay thế** | - Tại bước 3 các chức năng không thực hiện đúng giai đoạn sẽ không được hiển thị trên hệ thống giao diện. |
| **Luồng ngoại lệ** | - Tại bước 2 không tìm thấy thông tin sản phẩm sẽ hiển thị không có đơn hàng.  - Tại bước 4 nếu không tìm thấy thông tin chi tiết của sản phẩm trong đơn hàng sẽ hiển thị không tìm thấy và sẽ báo lỗi.  - Tại bước 3 hệ thống sẽ không thực hiện được các chức năng và hiển thị lỗi trên hệ thống. |

Bảng 4: Use Case Theo Dõi Đơn Hàng

* + - 1. **Use case thống kê báo cáo:**

|  |  |
| --- | --- |
| **Use case ID** | UC005 |
| **Tên use case** | Thống kê báo cáo |
| **Mô tả vắn tắt Use Case** | - Use case cho phép nhân viên thực hiện thống kê doanh thu theo tháng trong năm. |
| **Actor chính** | Nhân viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | - Nhân viên đã đăng nhập vào hệ thống.  - Hệ thống có số liệu cần được thống kê. |
| **Hậu điều kiện** | - Nhân viên có thể xem các báo cáo thống kê.  - Nhân viên có thể xuất báo cáo dưới dạng file (ví dụ: Excel, PDF). |
| **Luồng hoạt động** | **Bước 1:** Nhân viên truy cập vào hệ thống trang thống kê báo cáo.  **Bước 2:** Hệ thống hiển thị biểu đồ thống kê báo cáo theo dữ liệu của hệ thống.  **Bước 3:** Nhân viên bấm nút xuất báo cáo.  **Bước 4**: Hệ thống sẽ kiểm tra dữ liệu và thực hiện xuất báo cáo.  **Bước 5:** Hệ thống thông báo thành công và gửi sang mail cho nhân viên. |
| **Luồng thay thế** | - Hệ thống có thể theo dõi thống kê báo cáo theo tháng và năm. |
| **Luồng ngoại lệ** | - Nếu xảy ra lỗi hệ thống sẽ kiểm trả và thông báo lỗi lên trên giao diện người dùng. |

Bảng 5: Use Case Thống Kê Báo Cáo

* + - 1. **Use case đánh giá sản phẩm:**

|  |  |
| --- | --- |
| **Use case ID** | UC006 |
| **Tên use case** | Đánh giá sản phẩm |
| **Mô tả vắn tắt Use Case** | - Use case cho phép người dùng đánh giá sản phẩm của hệ thống. |
| **Actor chính** | Khách Hàng |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | - Khách hàng đã đăng nhập vào hệ thống.  - Khách hàng phải truy cập vào đúng sản phẩm cần đánh giá. |
| **Hậu điều kiện** | - Hiển thị được các đánh giá khác.  - Thực hiện được thành công đánh giá sản phẩm. |
| **Luồng hoạt động** | **Bước 1:** Người dùng truy cập vào hệ thống trang chi tiết sản phẩm cần đánh giá.  **Bước 2:** Hệ thống hiển thị những đánh giá trước đó và form đánh giá cho người dùng.  **Bước 3:** Người dùng điền thông tin đánh giá và nhấn nút đánh giá.  **Bước 4:** Hệ thống hiển thị đánh giá thành công và lưu bài đánh giá vào cơ sở dữ liệu. |
| **Luồng thay thế** | - Tại bước 3 nếu người dùng chưa truy cập sẽ chuyển sang trang login để thực hiện việc đánh giá.  - Tại bước 3 người dùng không điền đầy đủ thông tin cần điền sẽ hiển thị lỗi. |
| **Luồng ngoại lệ** | - Tại bước 4 hệ thống không lưu được dữ liệu của bài đánh giá sẽ thông báo lỗi cho người dùng.  - Nếu xảy ra lỗi ngoài ý muốn sẽ hiển thị lỗi lên hệ thống và thực hiện việc trở lại trang chủ. |

Bảng 6: Use Case Đánh Giá Sản Phẩm

* 1. **Thiết kế hệ thống:**

1. **THẾT KẾ HỆ THỐNG:**

TÀI LIỆU THAM KHẢO

(Theo chuẩn IEEE – *bỏ dòng này khi viết báo cáo*)

PHỤ LỤC