TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KÌ MÔN CÔNG NGHỆ JAVA**

**WEBSITE BÁN ALBUM NHẠC**

*Người hướng dẫn*: **ThS NGUYỄN THANH PHONG**

*Người thực hiện*: **LÊ QUỐC HẢO - 52000331**

**NGUYỄN HỮU AN –**

**VÒONG QUANG MẠNH**

**NGÔ TƯỜNG VI-**

Lớp **: 20050401**

Khoá  **: 24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KÌ MÔN CÔNG NGHỆ JAVA**

**WEBSITE BÁN ALBUM NHẠC**

*Người hướng dẫn*: **ThS NGUYỄN THANH PHONG**

*Người thực hiện*: **LÊ QUỐC HẢO - 52000331**

**NGUYỄN HỮU AN –**

**VÒONG QUANG MẠNH**

**NGÔ TƯỜNG VI-**

Lớp **: 20050401**

Khoá  **: 24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

LỜI CẢM ƠN

Lời đầu tiên em xin được gửi lời cảm ơn đến thầy Nguyễn Thanh Phong đã nỗ lực giảng dạy chúng em trong suốt thời gian vừa qua. Cảm ơn thầy đã tiếp thêm kiến thức vào hành trang trước khi bước vào đời của chúng em qua những buổi học đầy ý nghĩa của thầy.

Đồng thời, em xin được cảm ơn phía Nhà trường đã tạo điều kiện cho chúng em có được lớp học như thế này để tiếp thu kiến thức.

Trong quá trình học tập và làm bài thì chắc chắn sẽ có sai sót, nên em rất mong thầy sẽ giúp chúng em góp ý để không chỉ giúp chúng em khắc phục mà vững vàng kiến thức cũng như kinh nghiệm làm việcsau này.

Chúng em một lần nữa xin được cảm ơn thầy và chúc thầy nhiều sức khỏe cũng như thành công trong sự nghiệp giảng dạy trong tương lai.

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của ThS Nguyễn Thanh Phong;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Lê Quốc Hảo*

*Nguyễn Hữu an*

*Ngô Tường Vi*

*Vòong Quang Mạnh*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

Trong nội dung báo cáo này chúng ta sẽ thảo luận về cách xây dựng một website bán Album nhạc online sử dụng ngôn ngữ Java với Spring Boot để xử lý dữ liệu. Ngoài ra còn dùng MySQL để lưu trữ dữ liệu; ngôn ngữ khác như HTML, CSS, JavaSript, Boostrap5 để xây dựng giao diện hệ thống các website.

Website sẽ có hai phần chính là phần quản trị website và một phần là của người dùng:

* Phần của người dùng khách hàng sẽ là tác nhân chính sử dụng để xem những sản phẩm có bán tại cửa hàng sau đó là đặt mua.
* Phần quản trị website do quản trị viên sử dụng để quản lý, kiểm tra các sản phẩm và người dùng và các đơn hàng

Website được xây dựng theo kiến trúc MVC để có thể dễ dàng quản lý và xây dựng và bảo mật.

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc133765915)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iii](#_Toc133765916)

[TÓM TẮT iv](#_Toc133765917)

[MỤC LỤC 1](#_Toc133765918)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 4](#_Toc133765919)

[CHƯƠNG 1 – GIỚI THIỆU 6](#_Toc133765920)

[1.1 Lý do chọn đề tài 6](#_Toc133765921)

[1.2 Mục tiêu và ý nghĩa đề tài 6](#_Toc133765922)

[1.3 Yêu cầu đề tài 6](#_Toc133765923)

[1.4 Phương pháp nghiên cứu và thực hiện 7](#_Toc133765924)

[1.5 Cấu trúc bài báo cáo 8](#_Toc133765925)

[CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT 9](#_Toc133765926)

[2.1. Mô hình MVC 9](#_Toc133765927)

[2.2. Spring Boot 10](#_Toc133765928)

[2.3. Hibernate 10](#_Toc133765929)

[2.4. MySQL 11](#_Toc133765930)

[2.5. Maven 11](#_Toc133765931)

[3.1. Đặc tả 12](#_Toc133765932)

[3.2. Actor 13](#_Toc133765933)

[3.3. Usecase + Sequence diagram 14](#_Toc133765934)

[3.3.1. Bảng usecase và sơ đồ usecase tổng quát 14](#_Toc133765935)

[3.3.2. Bảng usecase chi tiết và sơ đồ usecase chi tiết và Sequence diagram. 15](#_Toc133765936)

[3.4. Class diagram 31](#_Toc133765937)

[3.5. Database design 31](#_Toc133765938)

[CHƯƠNG 4: THỰC NGHIỆM 32](#_Toc133765939)

[1. Testing, Test case, Unit test 32](#_Toc133765940)

[2. Demo 32](#_Toc133765941)

**DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT**

**CÁC KÝ HIỆU**

**CÁC CHỮ VIẾT TẮT**

|  |  |
| --- | --- |
| UC | Usecase |
| CSDL | Cơ sở dữ liệu |
| CRUD | Create, Read, Update, Delete |
| MVC | Model – View - Controller |

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH**

[Hình 1: Mô hình MVC 8](#_Toc133765890)

[Hình 2: Hibernate Workflow 9](#_Toc133765891)

[Hình 3: Sơ đồ usecase tổng quát trang quản trị 13](#_Toc133765892)

[Hình 4: Sơ đồ usecase đăng nhập 14](#_Toc133765893)

[Hình 5: Sơ đồ usecase Album 14](#_Toc133765894)

[Hình 6: Bảng usecase Song 16](#_Toc133765895)

[Hình 7: Bảng usecase Genre 17](#_Toc133765896)

[Hình 8: Bảng usecase Singer 19](#_Toc133765897)

[Hình 9: : Bảng usecase User 20](#_Toc133765898)

[Hình 10: : Bảng usecase Order 22](#_Toc133765899)

[Hình 11: Sơ đồ Sequence diagram cho chức năng CRUD cho các usecase trên 24](#_Toc133765900)

[Hình 12: Sơ đồ usecase phân trang 25](#_Toc133765901)

[Hình 13: Sơ đồ usecase Tìm kiếm 26](#_Toc133765902)

[Hình 14: Sơ đồ usecase chuyển trang 28](#_Toc133765903)

**DANH MỤC BẢNG**

[Bảng 1: Bảng danh sách actor chính 13](#_Toc133765904)

[Bảng 2: Bảng Usecase tổng quát 14](#_Toc133765905)

[Bảng 3: Bảng usecase Album 16](#_Toc133765906)

[Bảng 4: Bảng usecase Song 18](#_Toc133765907)

[Bảng 5: Bảng usecase Genre 19](#_Toc133765908)

[Bảng 6: Bảng usecase Singer 21](#_Toc133765909)

[Bảng 7: Bảng usecase User 22](#_Toc133765910)

[Bảng 8: Bảng usecase Order 24](#_Toc133765911)

[Bảng 9: Bảng usecase phân trang 27](#_Toc133765912)

[Bảng 10: Bảng usecase tìm kiếm 28](#_Toc133765913)

[Bảng 11: Bảng usecase chuyển trang 30](#_Toc133765914)

CHƯƠNG 1 – GIỚI THIỆU

* 1. Lý do chọn đề tài

Hiện nay, trên nền tảng Internet lĩnh vực bán hàng online ngày càng thịnh hành và phổ biến rộng rãi. Người dùng hiện nay đa số thời gian họ đều ở trên các mạng xã hội và tính chất mọi người thích ở nhà mà có thể mua sắm những thứ mình chiếm đa số hơn. Vì thế đây là lý do chính các cửa hàng Online ngày càng nhiều hơn.

Với số lượng người dùng mạng xã hội ngày càng gia tăng thì các doanh nghiệp muốn phát triển thêm về lĩnh vực kinh doanh kỹ thuật số như này sẽ có thêm nguồn thu nhập chính. Vì thế, các trang web đang coi là xương sống của doanh nghiệp, bộ mặt của doanh nghiệp và mở rộng danh tiếng.

Do đó, để hòa nhập vào vấn đề này thì yêu cầu tạo ra một trang web bán hàng sẽ khó khăn hơn và không còn đơn giản như trước. Và đây cũng là một lĩnh vực liên quan đến chương trình học môn Công nghệ Java nên chúng em quyết định tạo ra một trang web bán album nhạc. Để giúp cho những người thích sưu tầm những album nhạc và đây cũng là lĩnh vực giải trí nằm trong top hàng đầu của thế giới được nhiều người dùng. Bên cạnh đó, nó cũng giúp chúng em review và củng cố thêm về nội dung trong chương trình học.

* 1. Mục tiêu và ý nghĩa đề tài

Hiểu rõ và sử dụng các công nghệ, framework Spring Boot, JPA, Spring Security,… để xây dựng web.

Xây dựng một trang web không những thuận mắt về bề ngoài, xử lý dữ liệu trơn tru mà còn bảo mật được hệ thống.

Giúp cho cho những người sử dụng Internet có thêm sự lựa chọn mua hàng về các loại mặt hàng, sự tiện lợi khi sử dụng website của chúng em và an tâm về tính bảo mật. Giảm được chi phí quản lý nhân sự và thời gian làm việc.

* 1. Yêu cầu đề tài

Giao diện người dùng:

* Đăng nhập, đăng ký tài khoản người dùng
* Xem thông tin tài khoản và đăng xuất tài khoản khỏi trang web
* Hiển thị các sản phẩm nổi bật và giới thiệu các mặt hàng best seller tại cửa hàng
* Xem thông tin các sản phẩm một cách chi tiết
* Lọc ra và tìm kiếm những sản phẩm theo nhu cầu.
* Thêm sản phẩm và giỏ hàng
* Đặt hàng các sản phẩm (xóa, chỉnh sửa số lượng món hàng)
* Thanh toán những mặt hàng vừa đặt và xem lại các đơn hàng vừa đặt
* Bảo mật các trang web không cho người dùng sử dụng một cách tùy tiện mà phải theo trình tự được cho trong hệ thống
* Gửi email thông báo đơn hàng các người dùng

Giao diện quản trị viên:

* CRUD các sản phẩm, đơn hàng, người dùng
* Tìm kiếm các items

Đối tượng sử dụng:

- Người dùng các nhu cầu mua sản phẩm qua website.

- Nhân viên cửa hàng

- Quản trị viên quản lý website

* 1. Phương pháp nghiên cứu và thực hiện

- Tìm hiểu về framework Spring boot, JPA, Spring Web, Spring Security,…

- Quản lý các thư viện bằng Maven với các dependency trên.

- Phân tích và thiết kế hệ thống website theo quy trình rõ ràng.

- Dùng Boostrap, HTML, CS, JS để thiết kế giao diện.

- Ngôn ngữ sử dụng chính là java, xử lý CSDL là MySQL

- Sử dụng kiến trúc MVC để thiết kế web để dễ dàng xử lý luồng dữ liệu và xử lý dữ liệu trả về cho người dùng. Dễ dàng bảo mật.

- Kết nối với CSDL bằng Hibernate

- Bảo mật bằng Spring Security

* 1. Cấu trúc bài báo cáo

Chương 1: Giới thiệu

Chương 2: Cơ sở lý thuyết

Chương 3: Phân tích hệ thống

Chương 4: Thực nghiệm

CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT

1. Mô hình MVC

MVC là một mô hình thiết kế hay còn được gọi là một kiến trúc phần mềm được sử dụng ở lĩnh vực phần mềm.

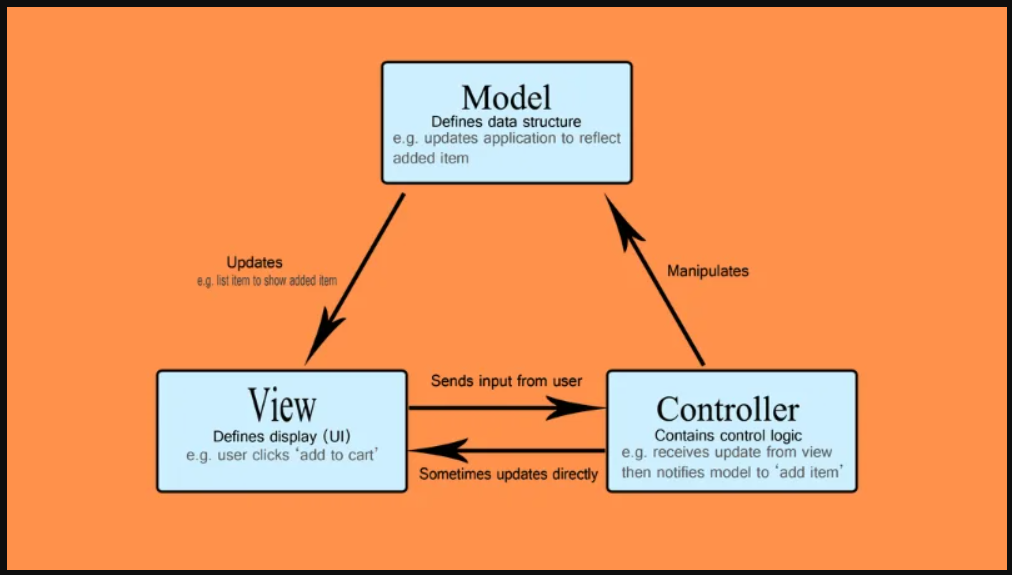
MVC phân bố source code thành 3 phần, mỗi phần có nhiệm vụ riêng biệt và độc lập với các thành phần khác. Đây làm một mô hình thiết kế mà các nhà thiết kế website hay lập trình website sử dụng để xử lý các yêu cầu kỹ thuật cùng hoàn thiện sản phẩm nhanh hơn và tiện lợi hơn.

Chúng em sử dụng ngôn ngữ java và lập trình theo hướng đối tượng nên kết hợp với MVC sẽ càng tối ưu được khả năng hoạt động hướng đối tượng trong MVC.

**Model:** Đây là thành phần được dùng để thực hiện chức năng lưu trữ toàn bộ dữ liệu. Là thành phần trung gian giữa View và Controller. Nó thường là một CSDL, file XML. Model cũng bao gồm các Class hay Function với nghiệp vụ kết nối CSDL, CRUD và truy vấn dữ liệu

**View**: Là nơi chưa các code giao diện của toàn bộ hệ thống. Thành phần tương tác chính với người dùng. Đây là tập hợp các file HTML, CSS, forms,.. để người dùng gửi dữ liệu đến **Controller**.

**Controller**: Đây là nơi xử lý các chức năng với dữ liệu nhận được từ người dùng ở View gửi đến. Nó sẽ xử lý và trả dữ liệu về lại **View**.



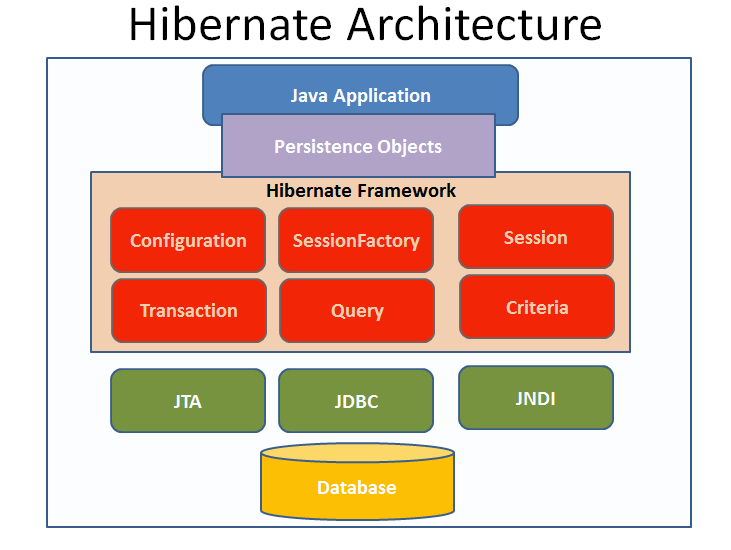
Hình 1: Mô hình MVC

1. Spring Boot

Spring Boot là một dự án phát triển bởi JAV (ngôn ngữ java) trong hệ sinh thái Spring framework. Nó giúp cho các lập trình viên chúng ta đơn giản hóa quá trình lập trình một ứng dụng với Spring, chỉ tập trung vào việc phát triển business cho ứng dụng.[[1]](#footnote-1)

1. Hibernate

Hibernate là một thư viện ORM (Object Relational Mapping) mã nguồn mở giúp lập trình viên viết ứng dụng Java có thể map các objects (pojo) với hệ quản trị cơ sở dữ liệu quan hệ,và hỗ trợ thực hiện các khái niệm lập trình hướng đối tượng với cớ dữ liệu quan hệ.[[2]](#footnote-2)



Hình 2: Hibernate Workflow

1. MySQL

MySQL là một hệ thống cơ sở dữ liệu mã nguồn mở (Relational Database Management System – được gọi tắt là RDBMS), được Tập đoàn Oracle phát triển dựa trên ngôn ngữ truy vấn có cấu trúc (SQL)[[3]](#footnote-3)

Đây được xem là hệ thống quản trị CSDL dễ dàng sử dụng, mang tính ổn định, dễ sử dụng và được các deveploper cùng các trường đại học đưa vào giảng dạy.

1. Maven

Maven là một công cụ giúp quản lý và thiết lập tự động dự án phần mềm. Chủ yếu được sử dụng ở các lập trình viên java.

Maven giúp việc khởi tạo các dự án ban đầu và quản lý các thư viện hỗ trợ lập trình tốt hơn, dễ dàng hơn

CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG

1. Đặc tả

Hệ thống website bán Album nhạc là nơi để người dùng truy cập vào để mua những album nhạc.

Người dùng sẽ phải đăng ký tài khoản và dùng tài khoản đó để đăng nhập vào hệ thống để có thể đặt hàng và thanh toán sản phẩm.

Hệ thống sẽ hiện ra các giao diện sản phầm, thanh filter, search, phân trang để người dùng có thể dễ dàng thao tác tìm kiếm sản phẩm. Ở giao diện trang chủ chứa những nội dung trên thì các sản phẩm sẽ được thể hiện ra với hình ảnh, giá tiền và một số thông tin đơn giản.

Khi người dùng có hứng thú một sản phẩm bất kỳ thì người dùng sẽ chọn sản phẩm đó để đi đến trang xem chi tiết sản phẩm. Ở trang chi tiết sản phẩm này người dùng sẽ thấy được tất cả các nội dung chi tiết của sản phầm đó như thể loại, ca sĩ,…

Giỏ hàng là nơi chưa các sản phẩm mà người dùng có ý định đặt mua. Từ giỏ hàng người dùng sẽ điền thông tin các thông tin cần thiết để có thể xác nhận đặt đơn hàng và nhận được email món hàng mình đặt.

Người dùng có thể thay đổi thông tin mình đã đăng ký tại mục thông tin tài khoản.

Trang website quản lý dành cho admin thì sẽ có thể thêm, đọc, chỉnh sửa, xóa các items; tìm kiểm các items, phân trang các items và tìm kiếm có items theo từ khóa.

Admin có thể thêm, xóa, xem, chỉnh sửa các items thông qua các trang Album, Song, Singer, Genre, User, Order. Riêng User và Order sẽ không có chức năng thêm items bởi vì đây sẽ là nơi nhận các người dùng đăng ký tài khoản và các đơn hàng người dùng đặt. Với quản lý User thì khi người dùng đăng ký tài khoản thì tài khoản đó mặc định sẽ là “customer”, vì thế admin sẽ có quyền chỉnh sửa “role” của user đó.

Quy trình đơn giản khi admin muốn CRUD Album:

* Tạo

1. Admin chọn “Album” để vào trang Album

2. Chọn “Add” để thêm

3. Điền dữ liệu vào các input và nhấn “Submit”

4. Kiểm tra dữ liệu không hợp lệ thì hệ thống sẽ cho admin nhập lại

5. Sau đó hệ thống sẽ nhận dữ liệu và lưu vào database và chuyển hướng đến trang “Album” có xuất hiện items mà admin vừa tạo.

* Chỉnh sửa

1. Admin chọn “Album” để vào trang Album

2. Chọn biểu tượng bút chì màu vàng để dẫn đến trang chỉnh sửa item

3. Chỉnh sửa và nhấn “Submit”

4. Kiểm tra dữ liệu không hợp lệ thì hệ thống sẽ cho admin nhập lại

5. Sau đó hệ thống sẽ nhận dữ liệu và cập nhật lại item đó và chuyển hướng đến trang “Album” có xuất hiện items mà admin vừa tạo.

* Xóa

1. Admin chọn “Album” để vào trang Album

2. Chọn biểu tượng thùng rác màu đỏ

3. Một khung cảnh báo sẽ xác nhận có muốn xóa hay không

4. Nhấn “Delete” để xác nhận xóa

5. Sau đó hệ thống sẽ gửi “id” mà admin muốn xóa đến database và xóa sản phẩm đó.

1. Actor

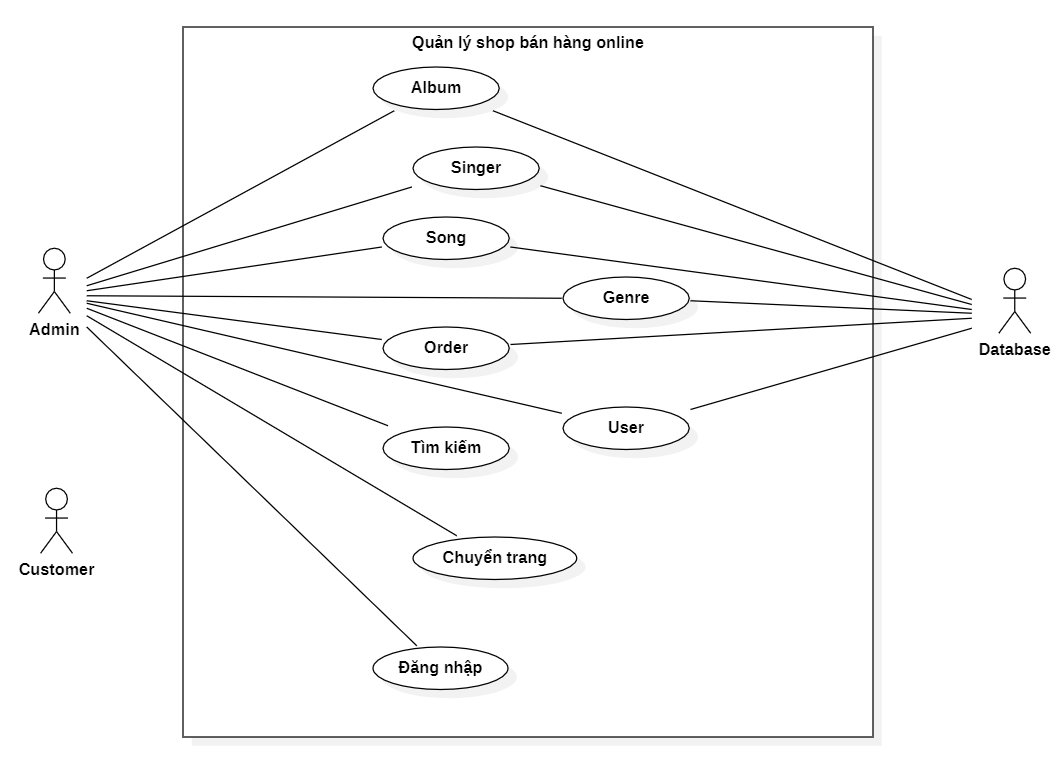
|  |  |
| --- | --- |
| Actor | Chức năng |
| Admin | Thêm, chỉnh sửa, xóa , xem:   * Album * Singer * Song * Genre * Order * User   Tìm kiếm  Xem thông tin sản phẩm  Chuyển trang  Đăng nhập |
| Khách hàng | Xem thông tin sản phẩm  Filter, tìm kiếm sản phẩm  Đặt hàng  Thêm sản phẩm vào giỏ hàng  Thay đổi thông tin tài khoản |
| Cơ sở dữ liệu | Nơi lưu trữ dữ liệu để xử lý, truy vấn |

Bảng 1: Bảng danh sách actor chính

1. Usecase + Sequence diagram
2. Bảng usecase và sơ đồ usecase tổng quát

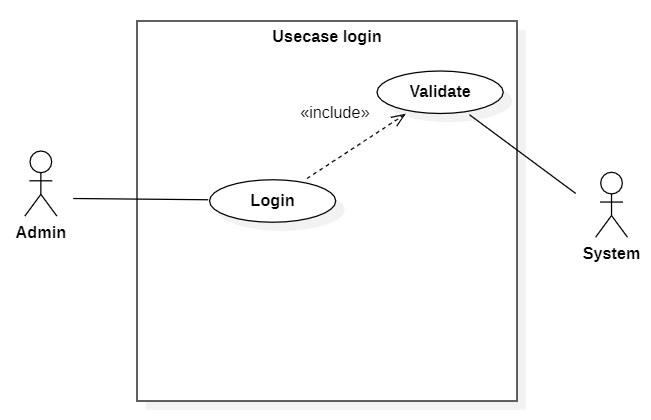
|  |  |  |
| --- | --- | --- |
| Usecase ID | Usecase | Mô tả |
|  | Usecase album | Thêm, xóa, sửa, cập nhật album |
|  | Usecase singer | Thêm, xóa, sửa, cập nhật singer |
|  | Usecase song | Thêm, xóa, sửa, cập nhật song |
|  | Usecase genre | Thêm, xóa, sửa, cập nhật genre |
|  | Usecase order | Thêm, xóa, sửa, cập nhật order |
|  | Usecase user | Thêm, xóa, sửa, cập nhật user |
|  | Usecase phân trang | Chia nhỏ số lượng sản phẩm xuất hiện trong một trang |
|  | Usecase tìm kiếm | Tìm kiếm các từ khóa có trong các danh mục |
|  | Usecase chuyển trang | Di chuyển giữa các trang quản lý |

Bảng 2: Bảng Usecase tổng quát

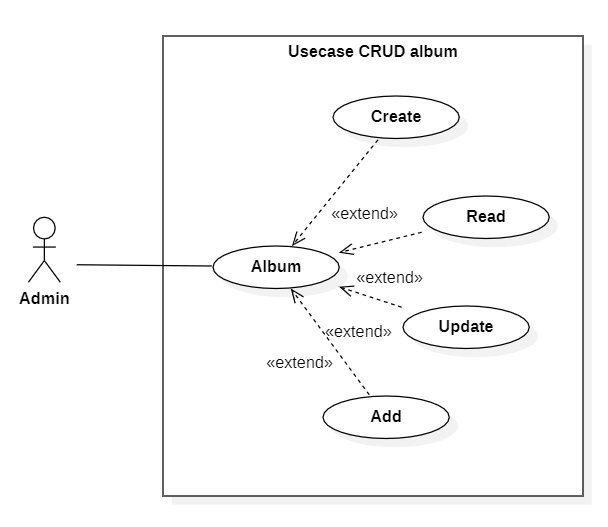


Hình 3: Sơ đồ usecase tổng quát trang quản trị

1. Bảng usecase chi tiết và sơ đồ usecase chi tiết và Sequence diagram.



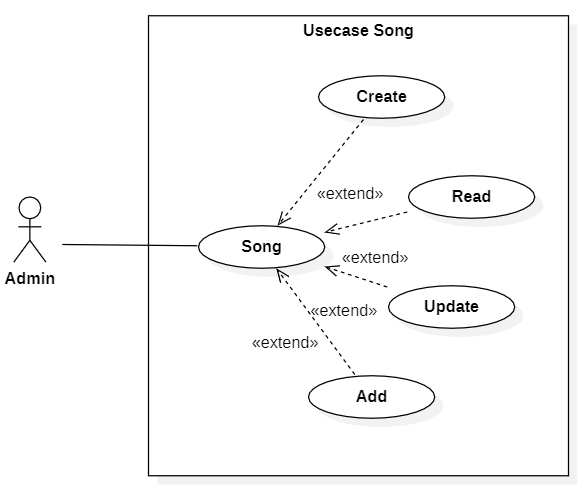
Hình 4: Sơ đồ usecase đăng nhập



Hình 5: Sơ đồ usecase Album

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | Album | |
| **Description** | Admin sẽ vào để tạo, đọc, chỉnh sửa, xóa các Album theo nhu cầu | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Thao tác được với các trang Album | |
| **Flow** | Actor | System |
| 1. Người dùng truy cập vào mục “Album”  2. Thao tác với trang “Album” | 1. Hiển thị mục “Album”  2. Hiển thị giao diện thực hiện chức năng |
| **Exception** | Người dùng chưa được phân quyền | |

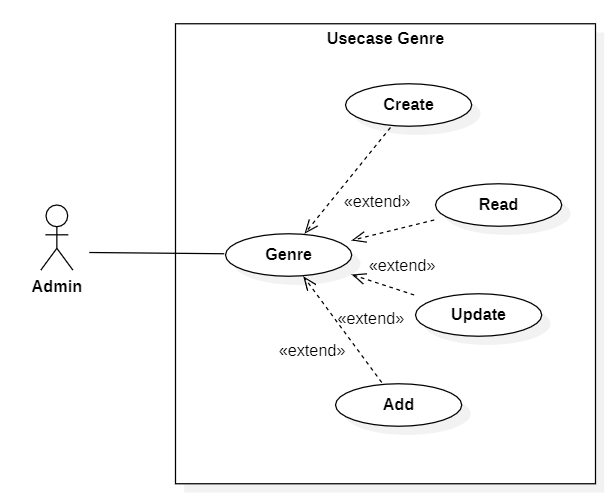
Bảng 3: Bảng usecase Album



Hình 6: Bảng usecase Song

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | Song | |
| **Description** | Admin sẽ vào để tạo, đọc, chỉnh sửa, xóa các Song theo nhu cầu | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Thao tác được với các trang Song | |
| **Flow** | Actor | System |
| 1. Người dùng truy cập vào mục “Song”  2. Thao tác với trang “Song” | 1. Hiển thị mục “Song”  2. Hiển thị giao diện thực hiện chức năng |
| **Exception** | Người dùng chưa được phân quyền | |

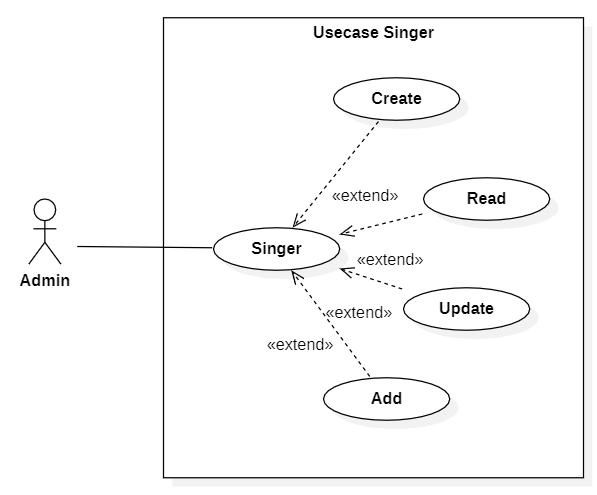
Bảng 4: Bảng usecase Song



Hình 7: Bảng usecase Genre

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | Genre | |
| **Description** | Admin sẽ vào để tạo, đọc, chỉnh sửa, xóa các Genre theo nhu cầu | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Thao tác được với các trang Genre | |
| **Flow** | Actor | System |
| 1. Người dùng truy cập vào mục “Genre”  2. Thao tác với trang “Genre” | 1. Hiển thị mục “Genre”  2. Hiển thị giao diện thực hiện chức năng |
| **Exception** | Người dùng chưa được phân quyền | |

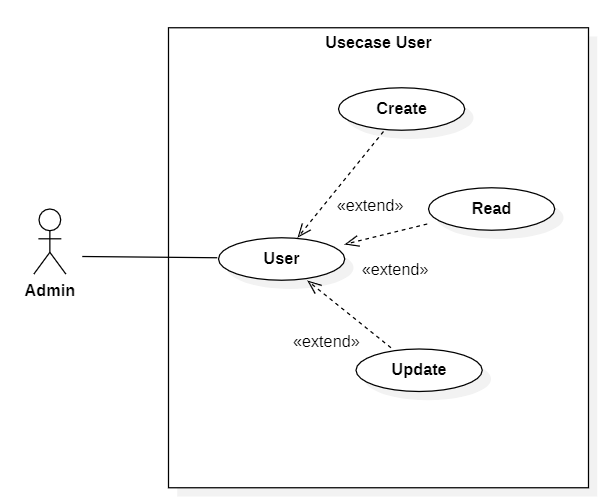
Bảng 5: Bảng usecase Genre



Hình 8: Bảng usecase Singer

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | Singer | |
| **Description** | Admin sẽ vào để tạo, đọc, chỉnh sửa, xóa các Singer theo nhu cầu | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Thao tác được với các trang Singer | |
| **Flow** | Actor | System |
| 1. Người dùng truy cập vào mục “Singer”  2. Thao tác với trang “Singer” | 1. Hiển thị mục “Singer”  2. Hiển thị giao diện thực hiện chức năng |
| **Exception** | Người dùng chưa được phân quyền | |

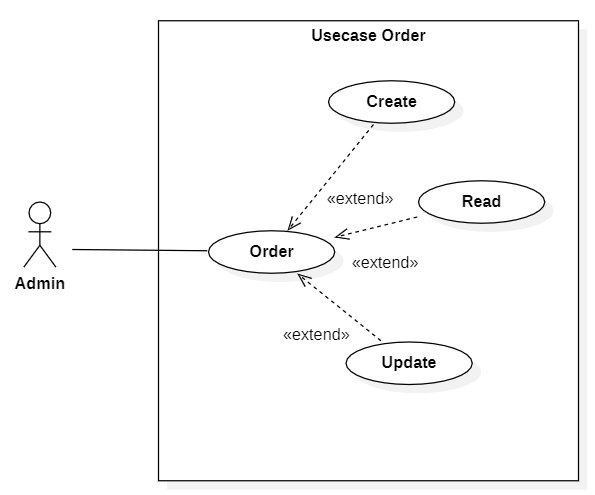
Bảng 6: Bảng usecase Singer



Hình 9: : Bảng usecase User

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | User | |
| **Description** | Admin sẽ vào để đọc, chỉnh sửa, xóa các User theo nhu cầu | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Thao tác được với các trang User | |
| **Flow** | Actor | System |
| 1. Người dùng truy cập vào mục “User”  2. Thao tác với trang “User” | 1. Hiển thị mục “User”  2. Hiển thị giao diện thực hiện chức năng |
| **Exception** | Người dùng chưa được phân quyền | |

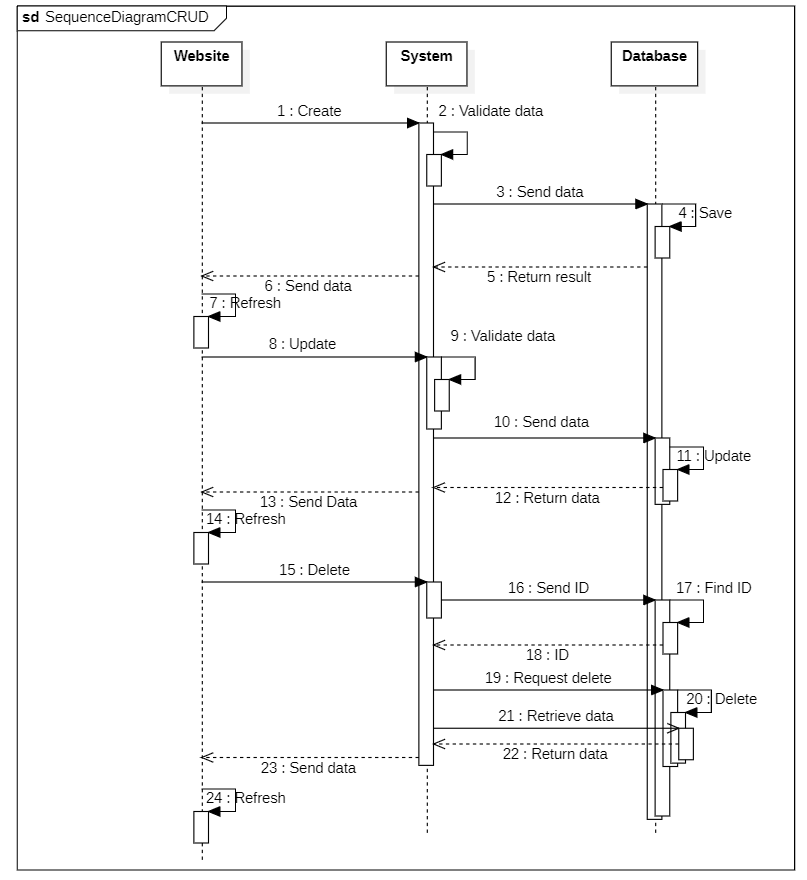
Bảng 7: Bảng usecase User



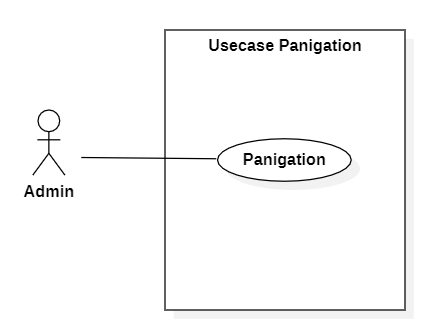
Hình 10: : Bảng usecase Order

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | Order | |
| **Description** | Admin sẽ vào để đọc, chỉnh sửa, xóa các Order theo nhu cầu | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Thao tác được với các trang Order | |
| **Flow** | Actor | System |
| 1. Người dùng truy cập vào mục “Order”  2. Thao tác với trang “Order” | 1. Hiển thị mục “Order”  2. Hiển thị giao diện thực hiện chức năng |
| **Exception** | Người dùng chưa được phân quyền | |

Bảng 8: Bảng usecase Order



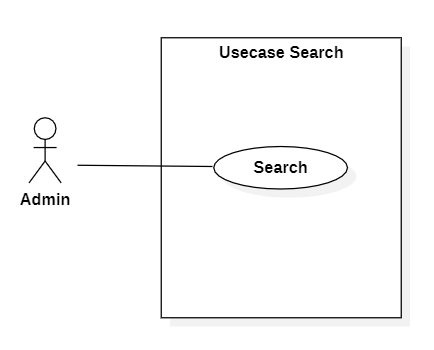
Hình 11: Sơ đồ Sequence diagram cho chức năng CRUD cho các usecase trên



Hình 12: Sơ đồ usecase phân trang

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | Phân trang | |
| **Description** | Admin sẽ di chuyển các trang sản phẩm được chia nhỏ ra của một danh mục | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Xem được các items chia nhỏ ra có trong một danh mục | |
| **Flow** | Actor | System |
| 1. Người dùng chọn phần panigation ở cuối trang  2. Chọn thứ trang muốn đi đến | 1. Hiển thị mục các trang  2. Di chuyển đến page người dùng vừa chọn |
| **Exception** | Trang không có sản phẩm  Người dùng thao tác vượt quá số trang hiện có | |

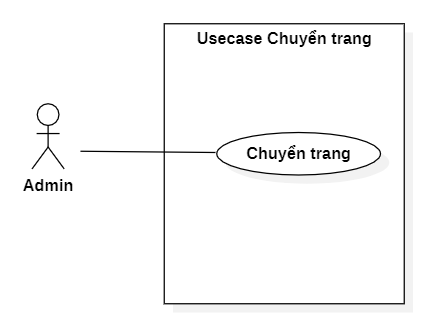
Bảng 9: Bảng usecase phân trang



Hình 13: Sơ đồ usecase Tìm kiếm

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | Tìm kiếm | |
| **Description** | Admin sẽ nhập từ khóa bất kỳ vào “Search” và hệ thống sẽ tìm kiếm và trả kết quả lại | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Tìm kiếm được items theo ý muốn | |
| **Flow** | Actor | System |
| 1. Người dùng điền từ khóa vào trong thanh “Search” đầu trang | 1. Hiển thị mục các mục “Search”  2. Trả về các kết quả dựa theo từ khóa của người dùng vừa nhập vào |
| **Exception** | Trang không có items mà người dùng nhập | |

Bảng 10: Bảng usecase tìm kiếm



Hình 14: Sơ đồ usecase chuyển trang

|  |  |  |
| --- | --- | --- |
| **Use Case ID** |  | |
| **Use Case Name** | Chuyển trang | |
| **Description** | Người dùng chọn các mục muốn đến ở bên trái màn hình và hệ thống sẽ dẫn đến trang theo người dùng vừa chọn | |
| **Actor(s)** | Admin | |
| **Stack Holder** | Admin, Website, Database | |
| **Trigger** | Tài khoản phải có “role” là “admin”  Người dùng đăng nhập thành công vào trang web | |
| **Pre-Condition(s):** | Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập  Người dùng đăng nhập thành công | |
| **Post-Condition(s):** | Chuyển đến các danh mục theo ý muốn | |
| **Flow** | Actor | System |
| 1. Chọn danh mục bất kỳ bên thanh navbar bên trái màn hình | 1. Hiển thị các danh mục  2. Di chuyển đến các danh mục tương ứng |
| **Exception** | Người dùng tự trỏ đến trang không có trong hệ thống | |

Bảng 11: Bảng usecase chuyển trang

1. Class diagram
2. Database design

CHƯƠNG 4: THỰC NGHIỆM

1. Testing, Test case, Unit test

2. Demo

**TÀI LIỆU THAM KHẢO**

**BẢNG PHÂN CÔNG CÔNG VIỆC**

|  |  |
| --- | --- |
| 1 | Nguyễn Hữu An |
| 2 | Lê Quốc Hảo |
| 3 | Voòng Quang Mạnh |
| 4 | Ngô Tường Vi |

|  |  |  |  |
| --- | --- | --- | --- |
| **TUẦN 1** | **CÔNG VIỆC** | **WHO** | **%** |
|  | **Trang trước khi đăng nhập và trang settings** | **4** | **100%** |
|  | **Trang login** | **2** | **100%** |
|  | **Trang sign up** | **1** | **100%** |
|  | **Trang chủ** | **3** | **100%** |
| **TUẦN 2** | **CÔNG VIỆC** |  |  |
|  | **Giao diện người dùng** | **3** | **100%** |
|  | **Giao diện người dùng** | **4** | **100%** |
|  | **Giao diện trang quản trị viên** | **2** | **100%** |
|  | **Thiết kế CSDL và dữ liệu** | **1** | **100%** |
| **TUẦN 3** | **CÔNG VIỆC** |  |  |
|  | **Chức năng trang quản trị viên** | **2** | **100%** |
|  | **Viết API cho user và order** | **3** | **100%** |
|  | **Bảo mật hệ thống, gửi email đơn hàng, tổng hợp code** | **1** | **100%** |
|  | **Chức năng trang người dùng** | **4** | **100%** |

1. Topdev. (2021).*Giới thiệu về Spring Boot. Spring Boot là gì? | TopDev.* Retrieved April 30, 2023 from https://topdev.vn/blog/gioi-thieu-ve-spring-boot-s pring-boot-la-gi/ [↑](#footnote-ref-1)
2. Topdev. (2021).*Hibernate là gì? Sao phải dùng nó thay JDBC? | TopDev.* TopDev. Retrieved April 30, 2023 from https://topdev.vn/blog/hibernate-la-gi-sao-phai-du ng-no-thay-jdbc/ [↑](#footnote-ref-2)
3. *MySQL là gì? Chi tiết kiến thức MySQL cho người mới 2022.* (n.d.). Retrieved April 30, 2023 from https://www.hostify.vn/blog/mysql-la-gi-kien-thuc- cho-nguoi-moi-bat-dau/#:~:text=(data,array\_type)-, 2.4 Mysql workbench là gì?,Linux và Mac OS X [↑](#footnote-ref-3)