

**TRƯỜNG ĐẠI HỌC SƯ РHẠM KỸ THUẬT**

**THÀNH РHỐ HỒ CHÍ MINH**

**KHОА CÔNG NGHỆ THÔNG TIN**

🙠🙟🕮🙝🙢

**A logo of hands holding a book

Description automatically generated**

**ĐỒ ÁN HỌC PHẦN**

Tên học phần: **ĐỒ ÁN CNTT**

Mã số lớp hp: **PROJ215879\_23\_1\_10CLC**

**Tên đề tài: XÂY DỰNG TRANG WEB BÁN SÁCH, BĂNG ĐĨA**

**VÀ ẤN PHẨM THẦN TƯỢNG TRỰC TUYẾN**

Giảng viên hướng dẫn: **TS. TỪ TUYẾT HỒNG**

**Sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| **Họ tên SV** | **MSSV** | **Lớp** |
| **NGUYỄN HOÀI THƯƠNG** | **21110317** | **21110CL6** |
| **MAI TẤN SANG** | **21110850** | **21110CL6** |

Khoá: K21

Ngành: CÔNG NGHỆ THÔNG TIN

Thành Рhố Hồ Chí Minh, tháng 5 năm 2022

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT**

**THÀNH PHỐ HỒ CHÍ MINH**

A logo of hands holding a book

Description automatically generated**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI ĐỒ ÁN HỌC PHẦN**

Tên học phần: **ĐỒ ÁN CNTT**

Mã số lớp hp: **PROJ215879\_23\_1\_10CLC**

**Tên đề tài: XÂY DỰNG TRANG WEB BÁN SÁCH, BĂNG ĐĨA VÀ ẤN PHẨM THẦN TƯỢNG TRỰC TUYẾN**

Giảng viên hướng dẫn: **TS. TỪ TUYẾT HỒNG**

**Sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| **Họ tên SV** | **MSSV** | **Lớp** |
| **NGUYỄN HOÀI THƯƠNG** | **21110317** | **21110CL6** |
| **MAI TẤN SANG** | **21110850** | **21110CL6** |

Khoá: K21

Ngành: CÔNG NGHỆ THÔNG TIN

**NHẬN XÉT CỦA GIẢNG VIÊN**

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

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

TP.HCM, ngày 03 tháng 12 năm 2023

Giảng viên: Ký tên

**MỤC LỤC**

[I. GIỚI THIỆU VỀ ĐỒ ÁN HỌC PHẦN 1](#_Toc152623498)

[II. PHÂN CÔNG CÔNG VIỆC 3](#_Toc152623499)

[III. THIẾT KẾ 3](#_Toc152623500)

[1. Thuật toán 3](#_Toc152623501)

[1.1 Công nghệ và thư viện: 3](#_Toc152623502)

[1.2 Kiến trúc ứng dụng: 4](#_Toc152623503)

[1.3 Môi trường thực thi: 4](#_Toc152623504)

[1.4 Các bước triển khai: 4](#_Toc152623505)

[2. Thiết kế lớp 5](#_Toc152623506)

[2.1 Class Diagram 5](#_Toc152623507)

[2.2 Danh mục các lớp 5](#_Toc152623508)

[2.3 Mô tả các phương thức trong một lớp 8](#_Toc152623509)

[2.3.1 Các phương thức trong lớp Artist 8](#_Toc152623510)

[2.3.2 Các phương thức trong lớp Bill 9](#_Toc152623511)

[2.3.3 Các phương thức trong lớp Bill 12](#_Toc152623512)

[2.3.4 Các phương thức trong lớp Category 14](#_Toc152623513)

[2.3.5 Các phương thức trong lớp Cart 15](#_Toc152623514)

[2.3.6 Các phương thức trong lớp Genre 16](#_Toc152623515)

[2.3.7 Các phương thức trong lớp MyServletContextListener 17](#_Toc152623516)

[2.3.8 Các phương thức trong lớp Product 18](#_Toc152623517)

[2.3.9 Các phương thức trong lớp Role 22](#_Toc152623518)

[2.3.10 Các phương thức trong lớp User 24](#_Toc152623519)

[2.3.11 Các phương thức trong lớp WebController 27](#_Toc152623520)

[2.3.12 Các phương thức trong lớp ShopController 29](#_Toc152623521)

[2.3.13 Các phương thức trong lớp GenreEM 32](#_Toc152623522)

[2.3.14 Các phương thức trong lớp ProductEM 33](#_Toc152623523)

[2.3.15 Các phương thức trong lớp DBUtil 35](#_Toc152623524)

[2.3.16 Các phương thức trong lớp CartController 35](#_Toc152623525)

[2.3.17 Các phương thức trong lớp BillController 38](#_Toc152623526)

[2.3.18 Các phương thức trong lớp ArtistEM 39](#_Toc152623527)

[2.3.19 Các phương thức trong lớp CategoryEM 40](#_Toc152623528)

[2.3.20 Các phương thức trong lớp BillEM 41](#_Toc152623529)

[3. Thiết kế dữ liệu 42](#_Toc152623530)

[3.1 Mô hình liên kết thực thể 42](#_Toc152623531)

[3.2 Mô hình quan hệ ERD 43](#_Toc152623532)

[3.3 Bảng mô tả các table trong CSDL 44](#_Toc152623533)

[4. Bảng mô tả các field trong table 44](#_Toc152623534)

[4.1 bill 44](#_Toc152623535)

[4.2 billdetail 45](#_Toc152623536)

[4.3 artist 45](#_Toc152623537)

[4.4 category 45](#_Toc152623538)

[4.5 genre 46](#_Toc152623539)

[4.6 product 46](#_Toc152623540)

[4.7 role 47](#_Toc152623541)

[4.8 user 47](#_Toc152623542)

[5. Thiết kế giao diện 48](#_Toc152623543)

[IV. CÀI ĐẶT VÀ KIỂM THỬ 59](#_Toc152623544)

[V. KẾT LUẬN 63](#_Toc152623545)

[TÀI LIỆU THAM KHẢO 64](#_Toc152623546)

**DANH MỤC CÁC HÌNH**

[Hình 1 Giao diện dự kiến 2](#_Toc152624553)

[Hình 2. Class Diagram 5](#_Toc152624554)

[Hình 3. Mô hình liên kết thực thể 42](#_Toc152624555)

[Hình 4. Mô hình quan hệ ERD 43](#_Toc152624556)

[Hình 5. Trang home 48](#_Toc152624557)

[Hình 6.Trang about 49](#_Toc152624558)

[Hình 7. Trang Cart 50](#_Toc152624559)

[Hình 8. Trang My Bill 51](#_Toc152624560)

[Hình 9. Trang Shop 51](#_Toc152624561)

[Hình 10. Trang Product detail 52](#_Toc152624562)

[Hình 11. Trang Login 52](#_Toc152624563)

[Hình 12. Trang Register 53](#_Toc152624564)

[Hình 13. Trang Profile 53](#_Toc152624565)

[Hình 14. Phần Header 53](#_Toc152624566)

[Hình 15. Phần Footer 54](#_Toc152624567)

[Hình 16. Trang Contact 55](#_Toc152624568)

[Hình 17. Trang admin 55](#_Toc152624569)

[Hình 18. Trang manage bill 55](#_Toc152624570)

[Hình 19. Trang manage product 56](#_Toc152624571)

[Hình 20. Trag add product 56](#_Toc152624572)

[Hình 21. Trang Delete product 57](#_Toc152624573)

[Hình 22. Trang Edit Product 58](#_Toc152624574)

[Hình 23. Trang Manage Customer 58](#_Toc152624575)

[Hình 24. Kiểm thử nhập sai định dạng email 59](#_Toc152624576)

[Hình 25. Kiểm thử nhập sai email hoặc mật khẩu đã đăng kí vào cơ sở dữ liệu 59](#_Toc152624577)

[Hình 26. Kiểm thử bỏ trống email hoặc password 60](#_Toc152624578)

[Hình 27. Kiểm thử nhập số lượng sản phẩm là số âm 61](#_Toc152624579)

[Hình 28. Kiểm thử bấm đặt hàng với giỏ hàng trống 61](#_Toc152624580)

[Hình 29. Kiểm thử việc tính phí vận chuyển cho đơn hàng 61](#_Toc152624581)

[Hình 30. Kiểm thử sao chép liên kết khi thanh toán đơn hàng qua một trình duyệt khác (1) 62](#_Toc152624582)

[Hình 31. Kiểm thử sao chép liên kết khi thanh toán đơn hàng qua một trình duyệt khác (2) 62](#_Toc152624583)

**DANH MỤC CÁC BẢNG**

[Bảng 1. Bảng phân công công việc 3](#_Toc152623580)

[Bảng 2. Danh mục các lớp 6](#_Toc152623581)

[Bảng 3. Danh mục các lớp \_ Nguyễn Hoài Thương 7](#_Toc152623582)

[Bảng 4. Danh mục các lớp \_ Mai Tấn Sang 8](#_Toc152623583)

[Bảng 5. Các phương thức trong lớp Artist 9](#_Toc152623584)

[Bảng 6. Các phương thức trong lớp Bill 13](#_Toc152623585)

[Bảng 7. Các phương thức trong lớp Category 15](#_Toc152623586)

[Bảng 8. Các phương thức trong lớp Genre 16](#_Toc152623587)

[Bảng 9. Các phương thức trong lớp Genre 17](#_Toc152623588)

[Bảng 10. Các phương thức trong lớp MyServletContextListener 18](#_Toc152623589)

[Bảng 11. Các phương thức trong lớp Product 22](#_Toc152623590)

[Bảng 12. Các phương thức trong lớp Role 23](#_Toc152623591)

[Bảng 13. Các phương thức trong lớp User 26](#_Toc152623592)

[Bảng 14 Các phương thức trong lớp WebController 28](#_Toc152623593)

[Bảng 15. Các phương thức trong lớp ShopController 32](#_Toc152623594)

[Bảng 16. Các phương thức trong lớp GenreEM 33](#_Toc152623595)

[Bảng 17. Các phương thức trong lớp ProductEM 35](#_Toc152623596)

[Bảng 18. Các phương thức trong lớp DBUtil 35](#_Toc152623597)

[Bảng 19. Các phương thức trong lớp CartController 38](#_Toc152623598)

[Bảng 20. Các phương thức trong lớp BillController 39](#_Toc152623599)

[Bảng 21. Các phương thức trong lớp ArtistEM 40](#_Toc152623600)

[Bảng 22. Các phương thức trong lớp CategoryEM 40](#_Toc152623601)

[Bảng 23. Các phương thức trong lớp BillEM 41](#_Toc152623602)

[Bảng 24 Bảng mô tả các table trong CSDL 44](#_Toc152623603)

[Bảng 25. Bảng mô tả các field trong bảng bill 44](#_Toc152623604)

[Bảng 26. Bảng mô tả các field trong bảng billdetail 45](#_Toc152623605)

[Bảng 27. Bảng mô tả các field trong bảng artist 45](#_Toc152623606)

[Bảng 28. Bảng mô tả các field trong bảng category 45](#_Toc152623607)

[Bảng 29. Bảng mô tả các field trong bảng genre 46](#_Toc152623608)

[Bảng 30. Bảng mô tả các field trong bảng product 47](#_Toc152623609)

[Bảng 31. Bảng mô tả các field trong bảng role 47](#_Toc152623610)

[Bảng 32. Bảng mô tả các field trong bảng user 47](#_Toc152623611)

[Bảng 33. Bảng thiết kế giao diện 58](#_Toc152623612)

[Bảng 34. Cài đặt và kiểm thử 62](#_Toc152623613)

1. GIỚI THIỆU VỀ ĐỒ ÁN HỌC PHẦN

Đồ án học phần CNTT vận dụng kiến thức và kỹ năng về lập trình hướng đối tượng để xây dựng một ứng dụng web Java Enterprise Edition (Java EE) bằng ngôn ngữ Java với Java Servlet và có khả năng tương tác với cơ sở dữ liệu thông qua công nghệ JPA (Java Persistence API). Một trang web động với Java Servlet và công nghệ JPA giúp thực hiện các chức năng cơ bản của một ứng dụng web cơ bản, hiển thị dữ liệu, thêm, sửa, xoá dữ liệu, quản lí phiên (Session Management) và các chức năng tương tác với người dùng.

Với đề tài: ***“Xây dựng trang web bán sách, băng đĩa và ấn phẩm thần tượng trực tuyến”,*** chúng em đã xây dựng một trang web thương mại điện tử cơ bản cho một cửa hàng bán sách, băng đĩa và ấn phẩm thần tượng thông qua đặt hàng trực tuyến. Khách hàng có thể truy cập vào trang web để tìm kiếm, xem sản phẩm, mua các sản phẩm như sách, băng đĩa và các ấn phẩm thần tượng theo phân loại và nghệ sĩ yêu thích. Người quản lí cửa hàng có thể quản lí việc thêm, sửa, xoá sản phẩm để cho khách hàng đặt hàng, bên cạnh đó người quản lí cũng có thể xem các đơn đặt hàng bởi khách hàng và quản lí thông tin khách hàng.

Dữ liệu và thông tin đầu vào của *trang web bán sách, băng đĩa và ấn phẩm thần tượng trực tuyến* là các *sản phẩm*, bao gồm sách, băng đĩa và ấn phẩm thần tượng, mỗi sản phẩm sẽ có chi tiết sản phẩm như mã sản phẩm, tên, mô tả, giá cả, hình ảnh và các thông tin liên quan khác. Ngoài ra dữ liệu và thông tin đầu vào của trang web cũng bao gồm *thông tin của quản lí và của khách hàng*, giúp để thực hiện các chức năng đặt hàng, theo dõi đơn hàng yêu cầu người dùng cần phải đăng nhập hoặc đăng kí tài khoản.

Trong trang web này, khách hàng có thể đặt bất kì số lượng sản phẩm nào để tạo thành đơn hàng. Giả định rằng dữ liệu cần thiết để tạo đơn hàng có thể được cung cấp bởi hoạt động tìm kiếm. Khách hàng có thể vào trang web để *xem sản phẩm, chi tiết sản phẩm, tìm kiếm sản phẩm*, lọc sản phẩm theo phân loại hàng và lọc sản phẩm theo nghệ sĩ. Khách hàng bấm thêm sản phẩm vào giỏ hàng. Trong giỏ hàng, khách hàng có thể *điều chỉnh số lượng của sản phẩm* muốn mua, có thể xoá bỏ trước khi tiến hành đặt hàng, và trong lúc này khách hàng có thể bấm ra ngoài để lựa thêm sản phẩm mới. Giỏ hàng sẽ hiển thị thông tin của sản phẩm, số lượng, giá theo số lượng, tổng tiền của đơn hàng. Đối với đơn từ trên 500.000 VNĐ sẽ được miễn phí vận chuyển, còn lại dưới 500.000 VNĐ phí ship mỗi đơn sẽ là 30.000 VNĐ. Sau khi đã quyết định số lượng và sản phẩm, khi khách hàng bấm chọn *đặt hàng* sẽ bắt buộc phải đăng nhập hoặc đăng kí (nếu chưa có thông tin khách hàng lưu trong cơ sở dữ liệu). Sau khi đăng nhập thành công, khách hàng sẽ có thể đặt hàng. Nếu đặt hàng thành công, khách hàng có thể *xem lại đơn hàng của mình vừa đặt*, *theo dõi trạng thái đơn hàng*. Khách hàng cũng có thể *điều chỉnh thông tin cá nhân* của mình. Mặt khác, người quản lí phải đăng nhập đúng vào tài khoản mặc định trong cơ sở dữ liệu (vì đăng kí tài khoản sẽ mặc định đặt là tài khoản của khách hàng, tài khoản của người quản lí được nhập trực tiếp trong cơ sở dữ liệu). Người quản lí sẽ có thể *quản lí các đơn đặt hàng* mà khách hàng đã đặt trước đó, *quản lí danh sách các khách hàng* đã đăng kí tài khoản và *quản lí sản phẩm*, có thể thêm, sửa, xoá sản phẩm để cho khách hàng có thêm lựa chọn để mua hàng. Ngoài những tính năng kể trên, trang web cũng hỗ trợ một số tính năng phụ như g*ửi email,* gắn các liên kết để người dùng có thể liên hệ để được hổ trợ.

**Giao diện dự kiến:**

Nguồn template: <https://templatemo.com> . ên template: Zay Shop Template.

Đường dẫn: https://templatemo.com/tm-559-zay-shopA screenshot of a website

Description automatically generated

Hình 1 Giao diện dự kiến

1. PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên SV | Mô tả khái quát mảng công việc SV thực hiện trong đồ án | Ước tính phần trăm đóng góp |
| 1 | Nguyễn Hoài Thương | Nghiên cứu thành phần đăng nhập, điều hướng và hiển thị sản phẩm, chi tiết sản phẩm của trang web bán hàng, xử lí các entity genre, role, product, user… và giao diện các trang shop, product-details, login, register, profile. | 100% |
| 2 | Mai Tấn Sang | Nghiên cứu thành phần giỏ hàng và hóa đơn của trang web bán hàng, xử lý các entity artist, category, bill, và giao diện các trang home, about, contact, cart, mybil, managebill. | 100% |

Bảng 1. Bảng phân công công việc

1. THIẾT KẾ
2. Thuật toán

Xây dựng một ứng dụng web Java Enterprise Edition (Java EE) bằng ngôn ngữ Java với Java Servlet và có khả năng tương tác với cơ sở dữ liệu thông qua công nghệ JPA (Java Persistence API). Một trang web động với Java Servlet và công nghệ JPA giúp thực hiện các chức năng cơ bản của một ứng dụng web cơ bản, hiển thị dữ liệu, thêm, sửa, xoá dữ liệu, quản lí phiên (Session Management) và các chức năng tương tác với người dùng. Ngoài ra, còn sử dụng thư viện JSTL (JavaServlet Pages Standard Tab Library) để hiển thị trên giao diện người dùng thông qua các điều kiện, vòng lặp, và các thao tác khác trên dữ liệu.

1.1 Công nghệ và thư viện:

Servlet: xử lý các yêu cầu HTTP và quản lí luồng điều hướng trong web

JPA (Java Persistence API): tương tác với cơ sở dữ liệu. Một công nghệ tiện lợi và đơn giản để lưu trữ và truy xuất dữ liệu từ ơ sở dữ liệu quan hệ

JSTL (JavaServlet Pages Standard Tà Library): hiển thị trên giao diện người dùng qua các vòng lặp, điều kiện...

Thư viện JDBC Driver cho MySQL: để kết nối và tương tác với cơ sở dữ liệu MySQL

EclipseLink: để tương tác với cơ sở dữ liệu quan hệ từ ứng dụng Java, giúp thực hiện các chức năng ánh xạ đối tượng, quản lí vòng đời và thực hiện truy vấn cơ sở dư liệu sử dụng đối tượng Java thay vì SQL.

Và một số thư viện khác...

1.2 Kiến trúc ứng dụng:

Trang web sử dụng mô hình MVC (Model-View-Controller) để phân chia ứng dụng thành các lớp Model (dữ liệu và logic), View (hiển thị trên giao diện người dùng) và Controller (xử lý các yêu cầu và kiểm soát luồng điều hướng).

1.3 Môi trường thực thi:

IDE (Integrated Development Environment): Eclipse (IDS Eclipse) để phát triển và quản lí mã nguồn của ứng dụng

Web Server: Apache Tomcat v9.0 – môi trường thực thi cho ứng dụng web Java để triển khai và chạy các servlet và JSP

Cơ sở dữ liệu: MySQL để lưu trưc và quản lí dữ liệu của trang web

1.4 Các bước triển khai:

Tạo Dynamic Web Project

Convert sang JPA Project

Tiến hành viết các class, jsp…

Kết nối với MySQL:

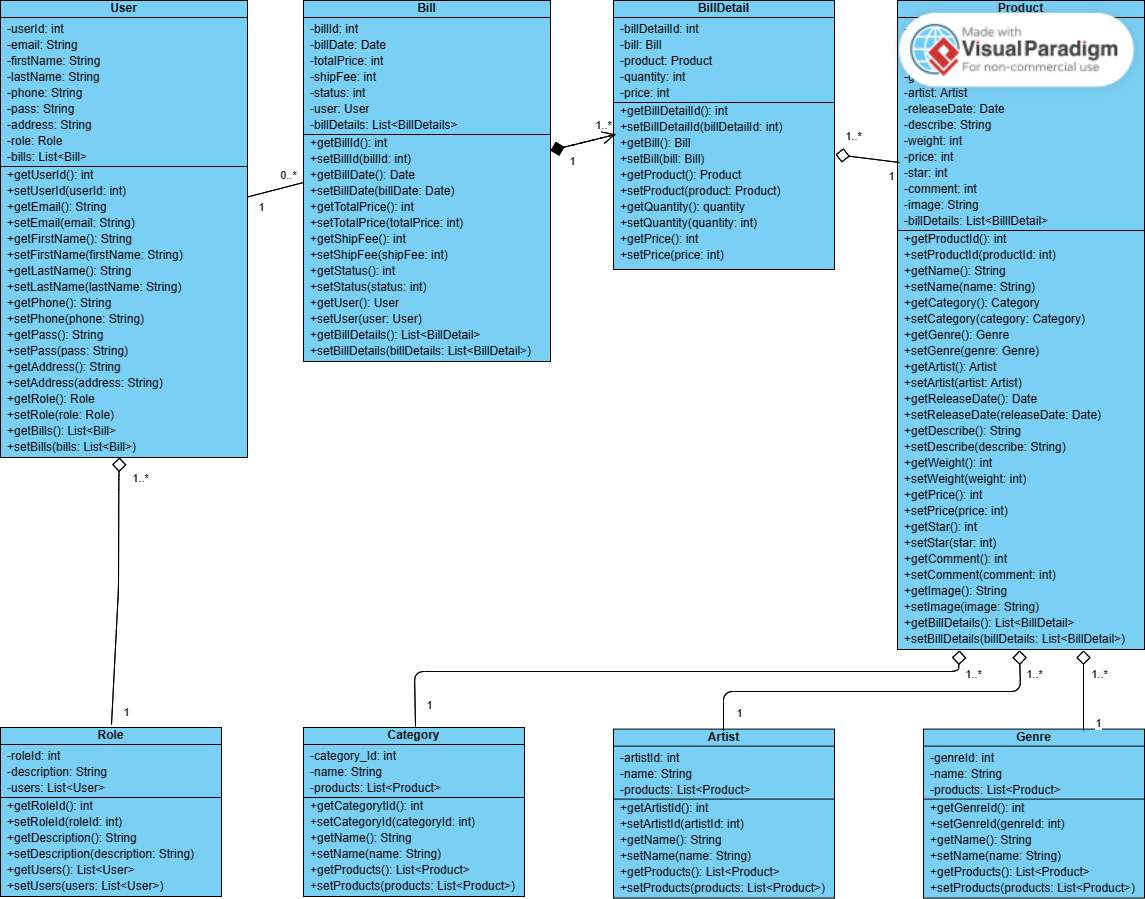
Copy JDBC Connection ở MySQL dán đường dẫn vào file persistence.xml và nhập đủ các thông tin kết nối

Tạo 1 DataBase Connections – MySQL và dán đường dẫn JDBC Connection ở trên và nhập các thông tin kết nối lại lần nữa

Chạy lên server, lúc này project sẽ mapping xuống database ta đã tạo sẵn ở MySQL và tạo các table theo các class.

Thêm dữ liệu bằng việc chạy script ở MySQL (nếu cần)

1. Thiết kế lớp
   1. Class Diagram



Hình 2. Class Diagram

* 1. Danh mục các lớp

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên lớp  Thừa kế hoặc dẫn xuất từ(ghi rõ tên lớp cơ sở) – nếu có | Mục đích | Tên các SV phụ trách viết |
|  | Artist | Thực hiện việc tạo bảng chứa thông tin artist. | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | Bill | Thực hiện việc tạo bảng chứa thông tin bill. | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | BillDetail | Thực hiện việc tạo bảng chứa thông tin của bill detail | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | Category | Thực hiện việc tạo bảng chứa thông tin của category | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | Genre | Thực hiện việc tạo bảng chứa thông tin genre. | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | MyServletContextListener | Quản lí vòng đời của EntityManagerFactory | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | Product | Thực hiện việc tạo bảng chứa thông tin của product | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | Role | Thực hiện việc tạo bảng chứa thông tin của role | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | User | Thực hiện việc tạo bảng chứa thông tin của user | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | Cart | Thực hiện lưu trữ các thông tin liên quan đến cart. | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 2. Danh mục các lớp

Sinh viên phụ trách: Nguyễn Hoài Thương

|  |  |  |
| --- | --- | --- |
| TT | Tên lớp  Thừa kế hoặc dẫn xuất từ(ghi rõ tên lớp cơ sở) – nếu có | Mục đích |
|  | WebController  kế thừa từ HttpServlet | Xử lí các yêu cầu liên quan đến điều hướng website, gồm đăng nhập, đăng kí, đăng xuất và các trang thông tin khác |
|  | ShopController  kế thừa từ HttpServlet | Xử lí các yêu cầu liên quan đến trang Shop trong website |
|  | GenreEM | Quản lí các thao tác liên quan đến thực thể Genre trong cơ sở dữ liệu thông qua JPA. Giúp tách lớp truy vấn và tương tác với thực thể Genre ra khỏi các lớp khác, tạo ra một lớp quản lí riêng giúp duy trì tính tổ chức và tái sử dụng mã nguồn trong website |
|  | ProductEM | Quản lí các thao tác liên quan đến thực thể Product trong cơ sở dữ liệu thông qua các phương thức khác nhau của class. Giúp tách lớp truy vấn và tương tác với thực thể Product ra khỏi các lớp khác, tạo ra một lớp quản lí riêng giúp duy trì tính tổ chức và tái sử dụng mã nguồn trong website |
|  | DBUtil | Cung cấp cơ chế quản lí EntityManager và EntityManagerFactory trong môi trường JPA. Giúp quản lí kết nối và tài nguyên liên quan đến cơ sở dữ liệu hiệu quả. |

Bảng 3. Danh mục các lớp \_ Nguyễn Hoài Thương

Sinh viên phụ trách: Mai Tấn Sang

|  |  |  |
| --- | --- | --- |
| TT | Tên lớp  Thừa kế hoặc dẫn xuất từ(ghi rõ tên lớp cơ sở) – nếu có | Mục đích |
|  | CartController kế thừa từ HttpServlet | Hiển thị trang Cart, xóa, sửa sản phẩm. Hiển thị giá tiền, tổng giá, phí ship. Tạo Bill. |
|  | BillController kế thừa từ HttpServlet | User: hiển thị mybill gồm danh sách hóa đơn  Admin: hiển thị managebill gồm danh sách hóa đơn. |
|  | ArtistEM.java | Thực hiện các truy vấn liên quan đến Artist trong database |
|  | CategoryEM.java | Thực hiện các truy vấn liên quan đến Category trong database |
|  | BillEM.java | Thực hiện các truy vấn liên quan đến Bill trong database |

Bảng 4. Danh mục các lớp \_ Mai Tấn Sang

* 1. Mô tả các phương thức trong một lớp

2.3.1 Các phương thức trong lớp Artist

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getArtistId()**  Input: không có  Output: artistId | Trả về giá trị artistId cho đối tượng hiện tại | Artist.java(29) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setArtistId(int artistId)**  Input: artistId  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho artistId | Thiết lập giá trị cho artistId của đối tượng | Artist.java(33) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getName()**  Input: không có  Output: name | Trả về giá trị name cho đối tượng hiện tại | Artist.java(37) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setName(String name)**  Input: name  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho name | Thiết lập giá trị cho name của đối tượng | Artist.java(41) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getProducts()**  Input: không có  Output: danh sách products | Trả về danh sách products cho đối tượng hiện tại | Artist.java(45) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setProducts(List<Product> products)**  Input: danh sách products  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho products | Thiết lập giá trị cho products của đối tượng | Artist.java(49) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 5. Các phương thức trong lớp Artist

2.3.2 Các phương thức trong lớp Bill

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getBillId()**  Input: không có  Output: billId | Trả về giá trị billId cho đối tượng hiện tại | Bill.java(45) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setBillId(int billId)**  Input: billId  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho billId | Thiết lập giá trị cho billId của đối tượng | Bill.java(49) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getBillDate()**  Input: không có  Output: billDate | Trả về giá trị billDate cho đối tượng hiện tại | Bill.java(53) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setBillDate(Date billDate)**  Input: billDate  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho billDate | Thiết lập giá trị cho billDate của đối tượng | Bill.java(57) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getTotalPrice()**  Input: không có  Output: totalPrice | Trả về giá trị totalPrice cho đối tượng hiện tại | Bill.java(61) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setTotalPrice(int totalPrice)**  Input: totalPrice  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho totalPrice | Thiết lập giá trị cho totalPrice của đối tượng | Bill.java(65) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getShipFee()**  Input: không có  Output: shipFee | Trả về giá trị shipFee cho đối tượng hiện tại | Bill.java(69) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setShipFee(int shipFee)**  Input: shipFee  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho shipFee | Thiết lập giá trị cho shipFee của đối tượng | Bill.java(73) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getStatus()**  Input: không có  Output: status | Trả về giá trị status cho đối tượng hiện tại | Bill.java(77) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setStatus(int status)**  Input: status  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho status | Thiết lập giá trị cho status của đối tượng | Bill.java(81) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getUser()**  Input: không có  Output: user | Trả về giá trị user cho đối tượng hiện tại | Bill.java(85) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setUser(User user)**  Input: user  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho user | Thiết lập giá trị cho user của đối tượng | Bill.java(89) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getBillDetails()**  Input: không có  Output: danh sách billDetails | Trả về danh sách billDetails cho đối tượng hiện tại | Bill.java(93) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setBillDetails(List<BillDetail> billDetails)**  Input: danh sách billDetails  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho danh sách billDetails | Thiết lập danh sách billDetails của đối tượng | Bill.java(97) | Nguyễn Hoài Thương  Mai Tấn Sang |

2.3.3 Các phương thức trong lớp Bill

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getBillDetailId()**  Input: không có  Output: billDetailId | Trả về giá trị billDetailId cho đối tượng hiện tại | BillDetail.java(37) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setBillDetailId(int billDetailId)**  Input: billDetailId  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho billDetailId | Thiết lập giá trị cho billDetailId của đối tượng | BillDetail.java(41) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getBill()**  Input: không có  Output: bill | Trả về giá trị bill cho đối tượng hiện tại | BillDetail.java(45) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setBill(Bill bill)**  Input: bill  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho bill | Thiết lập giá trị cho bill của đối tượng | BillDetail.java(49) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getProduct()**  Input: không có  Output: product | Trả về giá trị product cho đối tượng hiện tại | BillDetail.java(53) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setProduct(Product product)**  Input: product  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho product | Thiết lập giá trị cho product của đối tượng | BillDetail.java(57) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getQuantity()**  Input: không có  Output: quantity | Trả về giá trị quantity cho đối tượng hiện tại | BillDetail.java(61) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setQuantity(int quantity)**  Input: quantity  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho quantity | Thiết lập giá trị cho quantity của đối tượng | BillDetail.java(65) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getPrice()**  Input: không có  Output: price | Trả về giá trị price cho đối tượng hiện tại | BillDetail.java(69) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setPrice(int price)**  Input: price  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho price | Thiết lập giá trị cho price của đối tượng | BillDetail.java(73) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 6. Các phương thức trong lớp Bill

2.3.4 Các phương thức trong lớp Category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getCategoryId()**  Input: không có  Output: categoryId | Trả về giá trị categoryId cho đối tượng hiện tại | Category.java(29) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setCategoryId(int categoryId)**  Input: categoryId  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho categoryId | Thiết lập giá trị cho categoryId của đối tượng | Category.java(33) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getName()**  Input: không có  Output: name | Trả về giá trị name cho đối tượng hiện tại | Category.java(37) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setName(String name)**  Input: name  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho name | Thiết lập giá trị cho name của đối tượng | Category.java(41) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getProducts()**  Input: không có  Output:danh sách products | Trả về giá trị danh sách products cho đối tượng hiện tại | Category.java(45) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setProducts(List<Product> products)**  Input: danh sách products  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho danh sách products | Thiết lập giá trị cho danh sách products của đối tượng | Category.java(49) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 7. Các phương thức trong lớp Category

2.3.5 Các phương thức trong lớp Cart

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getQuantity()**  Input: không có  Output: quantity | Trả về giá trị quantity cho đối tượng hiện tại | Cart.java(18) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setQuantity(int quantity)**  Input: quantity  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho quantity | Thiết lập giá trị cho quantity của đối tượng | Cart.java(21) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getTotalPrice()**  Input: không có  Output: totalPrice | Trả về giá trị totalPrice cho đối tượng hiện tại | Cart.java(24) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setTotalPrice(int totalPrice)**  Input: totalPrice Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho totalPrice | Thiết lập giá trị cho totalPrice của đối tượng | Cart.java(27) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getProduct()**  Input: không có  Output: product | Trả về giá trị product cho đối tượng hiện tại | Cart.java(30) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setProduct(Product product)**  Input: product  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho product | Thiết lập giá trị cho product của đối tượng | Cart.java(33) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 8. Các phương thức trong lớp Genre

2.3.6 Các phương thức trong lớp Genre

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getGenreId()**  Input: không có  Output: genreId | Trả về giá trị genreId cho đối tượng hiện tại | Genre.java(27) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setGenreId(int genreId)**  Input: genreId  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho genreId | Thiết lập giá trị cho genreId của đối tượng | Genre.java(31) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getName()**  Input: không có  Output: name | Trả về giá trị name cho đối tượng hiện tại | Genre.java(35) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setName(String name)**  Input: name  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho name | Thiết lập giá trị cho name của đối tượng | Genre.java(39) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getProducts()**  Input: không có  Output:danh sách products | Trả về giá trị danh sách products cho đối tượng hiện tại | Genre.java(43) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setProducts(List<Product> products)**  Input: danh sách products  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho danh sách products | Thiết lập giá trị cho danh sách products của đối tượng | Genre.java(47) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 9. Các phương thức trong lớp Genre

2.3.7 Các phương thức trong lớp MyServletContextListener

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **contextInitialized(ServletContextEvent sce)**  Input: sce  Output: không có | Khởi tạo EntityManagerFactory khi ứng dụng được thực thi | MyServletContextListener.java(15) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **contextDestroyed(ServletContextEvent sce)**  Input: sce  Output: không có | Đóng EntitManagerFactory khi ứng dụng bị huỷ | MyServletContextListener.java(21) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **EntityManagerFactory getEmFactory()**  Input: không có  Output: emf | Cho phép các phần khác của web nhận EntiryManagerFactory đã khởi tạo | MyServletContextListener.java(28) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 10. Các phương thức trong lớp MyServletContextListener

2.3.8 Các phương thức trong lớp Product

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getProductId()**  Input: không có  Output: productId | Trả về giá trị productId cho đối tượng hiện tại | Product.java(65) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setProductId(int productId)**  Input: productId  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho productId | Thiết lập giá trị cho productId của đối tượng | Product.java(69) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getName()**  Input: không có  Output: name | Trả về giá trị name cho đối tượng hiện tại | Product.java(73) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setName(String name)**  Input: name  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho name | Thiết lập giá trị cho name của đối tượng | Product.java(77) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getCategory()**  Input: không có  Output: category | Trả về giá trị category cho đối tượng hiện tại | Product.java(81) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setCategory(Category category)**  Input: category  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho category | Thiết lập giá trị cho category của đối tượng | Product.java(85) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getGenre()**  Input: không có  Output: genre | Trả về giá trị genre cho đối tượng hiện tại | Product.java(89) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setGenre(Genre genre)**  Input: genre  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho genre | Thiết lập giá trị cho genre của đối tượng | Product.java(93) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getArtist()**  Input: không có  Output: artist | Trả về giá trị artist cho đối tượng hiện tại | Product.java(97) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setArtist(Artist artist)**  Input: artist  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho artist | Thiết lập giá trị cho artist của đối tượng | Product.java(101) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getReleaseDate()**  Input: không có  Output: releaseDate | Trả về giá trị releaseDate cho đối tượng hiện tại | Product.java(105) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setReleaseDate(Date releaseDate)**  Input: releaseDate  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho releaseDate | Thiết lập giá trị cho releaseDate của đối tượng | Product.java(109) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getDescribe()**  Input: không có  Output: describe | Trả về giá trị describe cho đối tượng hiện tại | Product.java(113) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setDescribe(String describe)**  Input: describe  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho describe | Thiết lập giá trị cho describe của đối tượng | Product.java(117) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getWeight()**  Input: không có  Output: weight | Trả về giá trị weight cho đối tượng hiện tại | Product.java(121) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setWeight(int weight)**  Input: weight  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho weight | Thiết lập giá trị cho weight của đối tượng | Product.java(125) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getPrice()**  Input: không có  Output: price | Trả về giá trị price cho đối tượng hiện tại | Product.java(129) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setPrice(int price)**  Input: price  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho price | Thiết lập giá trị cho price của đối tượng | Product.java(133) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getStar()**  Input: không có  Output: star | Trả về giá trị star cho đối tượng hiện tại | Product.java(137) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setStar(int star)**  Input: star  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho star | Thiết lập giá trị cho star của đối tượng | Product.java(141) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getComment()**  Input: không có  Output: comment | Trả về giá trị comment cho đối tượng hiện tại | Product.java(145) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setComment(int comment)**  Input: comment  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho comment | Thiết lập giá trị cho comment của đối tượng | Product.java(159) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getImage()**  Input: không có  Output: image | Trả về giá trị image cho đối tượng hiện tại | Product.java(153) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setImage(String image)**  Input: image  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho image | Thiết lập giá trị cho image của đối tượng | Product.java(157) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getBillDetails()**  Input: không có  Output: danh sách billDetails | Trả về giá trị danh sách billDetails cho đối tượng hiện tại | Product.java(161) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setBillDetails(List<BillDetail> billDetails)**  Input: danh sách billDetails  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho danh sách billDetails | Thiết lập giá trị cho danh sách billDetails của đối tượng | Product.java(165) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 11. Các phương thức trong lớp Product

2.3.9 Các phương thức trong lớp Role

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getRoleId()**  Input: không có  Output: roleId | Trả về giá trị roleId cho đối tượng hiện tại | Role.java(25) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setRoleId(int roleId)**  Input: roleId  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho roleId | Thiết lập giá trị cho roleId của đối tượng | Role.java(29) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getDescription()**  Input: không có  Output: description | Trả về giá trị description cho đối tượng hiện tại | Role.java(34) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setDescription(String description)**  Input: description  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho description | Thiết lập giá trị cho description của đối tượng | Role.java(38) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getUsers()**  Input: không có  Output: danh sách users | Trả về giá trị danh sách users cho đối tượng hiện tại | Role.java(43) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setUsers(List<User> users)**  Input: danh sách users  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho danh sách users | Thiết lập giá trị cho danh sách users của đối tượng | Role.java(47) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 12. Các phương thức trong lớp Role

2.3.10 Các phương thức trong lớp User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách viết |
|  | **getUserId()**  Input: không có  Output: userId | Trả về giá trị userId cho đối tượng hiện tại | User.java(50) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setUserId(int userId)**  Input: userId  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho userId | Thiết lập giá trị cho userId của đối tượng | User.java(54) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getEmail()**  Input: không có  Output: email | Trả về giá trị email cho đối tượng hiện tại | User.java(58) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setEmail(String email)**  Input: email  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho email | Thiết lập giá trị cho email của đối tượng | User.java(62) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getFirstName()**  Input: không có  Output: firstName | Trả về giá trị firstName cho đối tượng hiện tại | User.java(66) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setFirstName(String firstName)**  Input: firstName  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho firstName | Thiết lập giá trị cho firstName của đối tượng | User.java(70) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getLastName()**  Input: không có  Output: lastName | Trả về giá trị lastName cho đối tượng hiện tại | User.java(74) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setLastName(String lastName)**  Input: lastName  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho lastName | Thiết lập giá trị cho lastName của đối tượng | User.java(78) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getPhone()**  Input: không có  Output: phone | Trả về giá trị phone cho đối tượng hiện tại | User.java(82) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setPhone(String phone)**  Input: phone  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho phone | Thiết lập giá trị cho phone của đối tượng | User.java(86) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getPass()**  Input: không có  Output: pass | Trả về giá trị pass cho đối tượng hiện tại | User.java(90) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setPass(String pass)**  Input: pass  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho pass | Thiết lập giá trị cho pass của đối tượng | User.java(94) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getAddress()**  Input: không có  Output: address | Trả về giá trị address cho đối tượng hiện tại | User.java(98) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setAddress(String address)**  Input: address  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho address | Thiết lập giá trị cho address của đối tượng | User.java(102) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getRole()**  Input: không có  Output: role | Trả về giá trị role cho đối tượng hiện tại | User.java(106) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setRole(Role role)**  Input: role  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho role | Thiết lập giá trị cho role của đối tượng | User.java(110) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **getBills()**  Input: không có  Output: danh sách bills | Trả về giá trị danh sách bills cho đối tượng hiện tại | User.java(114) | Nguyễn Hoài Thương  Mai Tấn Sang |
|  | **setBills(List<Bill> bills)**  Input: danh sách bills  Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho danh sách bills | Thiết lập giá trị cho danh sách bills của đối tượng | User.java(118) | Nguyễn Hoài Thương  Mai Tấn Sang |

Bảng 13. Các phương thức trong lớp User

2.3.11 Các phương thức trong lớp WebController

Sinh viên phụ trách: Nguyễn Hoài Thương

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ phản hồi và chuyển hướng đến trang theo điều kiện từ tham số action của request | Điều phối yêu cầu HTTP GET từ client, chuyển hướng đến các trang khác nhau của web dựa vào tham số action của request. | WebController.java(38) |
|  | **doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ phản hồi và gọi các phương thức theo điều kiện từ tham số action của request | Xử lí yêu cầu HTTP POST từ client, xác định hành động dựa trên tham số action trong request và gọi các phương thức xử lí tương ứng | WebController.java(77) |
|  | **handleLogin(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ chuyển hướng trang hoặc nếu không đăng nhập thành công sẽ báo lỗi | Xử lí yêu cầu đăng nhập từ client. Kiểm tra thông tin đăng nhập của người dùng, xác thực trong cơ sở dư liệu và chuyển hướng dựa trên kết quả của quá trình đăng nhập | WebController.java(90) |
|  | **handleRegister(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ chuyển hướng trang hoặc nếu không đăng kí thành công sẽ báo lỗi | Xử lí yêu cầu đăng ký từ client. Kiểm tra tính hợp lệ của các tham số request, thêm người dùng với vai trò là khách hàng vào cơ sở dữ liệu. Xử lí trường hợp email đã tồn tại trong cơ sở dữ liệu | WebController.java(147) |

Bảng 14 Các phương thức trong lớp WebController

2.3.12 Các phương thức trong lớp ShopController

Sinh viên phụ trách: Nguyễn Hoài Thương

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ chuyển hướng trang để hiển thị danh sách sản phẩm và danh mục dựa trên điều kiện từ tham số action | Xử lí yêu cầu HTTP GET từ client gửi đến url “/shop”. Kiểm tra tham số action từ request và gọi các phương thức. Hiển thị danh sách sản phẩm, danh mục, nghệ sĩ. Xem chi tiết sản phẩm, sắp xếp, lấy sản phẩm theo danh mục, nghệ sĩ và tìm kiếm sản phẩm. Phân trang các sản phẩm và cố định số lượng sản phẩm trên mỗi trang. | ShopController.java(41) |
|  | **getProductBySearchName(HttpServletRequest request, HttpServletResponse response)**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ chuyển hướng trang để hiển thị danh sách sản phẩm giống với kết quả tìm kiếm | Xử lí yêu cầu tìm kiếm sản phẩm dựa trên tên sản phẩm được nhập từ người dùng. Sau khi thực hiện việc tìm kiếm, danh sách sản phẩm phù hợp sẽ được lấy từ cơ sở dữ liệu (gọi phương thức searchProductsByName từ ProductEM) và chuyển trang hiển thị kết quả tìm kiếm | ShopController.java(118) |
|  | **getProductByCategory(HttpServletRequest request, HttpServletResponse response)**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ chuyển hướng trang để hiển thị danh sách sản phẩm theo danh mục cụ thể | Xử lí yêu cầu hiển thị danh sách sản phẩm theo danh mục được yêu cầu. Truy vấn cơ sở dữ liệu để lấy danh sách sản phẩm (gọi phương thức getProductsByCategoryId từ ProductEM) theo thông tin danh mục từ request Đặt danh sách sản phẩm, danh sách danh mục và danh mục đã chọn vào request, chuyển trang hiển thị. | ShopController.java(152) |
|  | **getProductByArtist(HttpServletRequest request, HttpServletResponse response)**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ chuyển hướng trang để hiển thị danh sách sản phẩm theo nghệ sĩ cụ thể | Xử lí yêu cầu hiển thị danh sách sản phẩm theo nghệ sĩ được yêu cầu. Truy vấn cơ sở dữ liệu để lấy danh sách sản phẩm (gọi phương thức getProductsByArtistId từ ProductEM) theo thông tin nghệ sĩ từ request. Đặt danh sách sản phẩm, danh sách nghệ sĩ và nghệ sĩ đã chọn vào request, chuyển trang hiển thị. | ShopController.java(199) |
|  | **getDetailProduct(HttpServletRequest request, HttpServletResponse response)**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ chuyển hướng trang để hiển thị một sản phẩm cụ thể | Xử lí yêu cầu hiển thị chi tiết sản phẩm theo sản phẩm được yêu cầu. Truy vấn cơ sở dữ liệu để lấy thông tin sản phẩm, đặt sản phẩm là request rồi chuyển trang để hiển thị chi tiết sản phẩm | ShopController.java(242) |
|  | **sortProduct(HttpServletRequest request, HttpServletResponse response)**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ chuyển hướng trang để hiển thị danh sách sản phẩm đã được sắp xếp | Xử lí yêu cầu sắp xếp sản phẩm theo yêu cầu (theo giá, tên sản phẩm hoặc mặc định). Lấy thông tin kiểu sắp xếp từ request, thực hiện xếp dữ liệu tương ứng từ cơ sở dữ liệu, chuyển trang hiển thị danh sách sản phẩm đã được sắp xếp. | ShopController.java(272) |
|  | **doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException**  Input:   * request: đối tượng chứa thông tin yêu cầu HTTP từ client * response: đối tượng cho phép Servlet tạo phản hồi đáp lại client   Output: không có, mà sẽ gọi lại phương thức doGet | Gọi lại phương thức doGet vì muốn xử lý HTTP GET và HTTP POST cùng logic | ShopController.java(323) |

Bảng 15. Các phương thức trong lớp ShopController

2.3.13 Các phương thức trong lớp GenreEM

Sinh viên phụ trách: Nguyễn Hoài Thương

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **GenreEM(EntityManager entityManager)**  Input: entityManager để tương tác với cơ sở dữ liệu  Output: không có | Khởi tạo đối tượng GenreEM với EntityManager | GenreEM.java(13) |
|  | **getAllGenres()**  Input: không có  Output: danh sách đối tượng Genre lấy từ cơ sở dữ liệu | Lấy danh sách tất cả các genre từ cơ sở dữ liệu bằng JPQL(Java Persistence Query Language) | GenreEM.java(17) |
|  | **getGenreById(int genreId)**  Input: genreId  Output: đối tượng Genre nếu tồn tại, null nếu không tìm được | Lấy một genre theo Id từ cơ sở dữ liệu bằng phương thức fint của EntityManager | GenreEM.java(25) |

Bảng 16. Các phương thức trong lớp GenreEM

2.3.14 Các phương thức trong lớp ProductEM

Sinh viên phụ trách: Nguyễn Hoài Thương

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **ProductEM(EntityManager entityManager)**  Input: entityManager để tương tác với cơ sở dữ liệu  Output: không có | Khởi tạo đối tượng ProductEM với EntityManager | ProductEM.java(14) |
|  | **getAllProducts()**  Input: không có  Output: danh sách các đối tượng Product lấy từ cơ sở dữ liệu | Lây danh sách tất cả sản phẩm từ cơ sở dữ liệu | ProductEM.java(18) |
|  | **getProductById(int productId)**  Input: productId  Output: đối tượng Product tương ứng với productId nếu productId tồn tại, nếu không tồn tại sẽ trả về null | Lấy một sản phẩm theo Id truyền vào từ cơ sở dữ liệu | ProductEM.java(27) |
|  | **getProductsSortedByName()**  Input: không có  Output: danh sách các đối tượng Product được sắp xếp theo tên từ A-Z | Lấy danh sách sản phẩm được sắp xếp theo tên từ A-Z | ProductEM.java(33) |
|  | **getProductsSortedByPriceAsc()**  Input: không có  Output: danh sách các đối tượng Product được sắp xếp theo giá từ thấp đến cao | Lấy danh sách sản phẩm được sắp xếp theo giá từ thấp đến cao | ProductEM.java(42) |
|  | **getProductsByCategoryId(int categoryId)**  Input: categoryId  Output: danh sách các đối tượng Product thuộc danh mục có categoryId tương ứng | Lấy danh sách sản phẩm theo categoryId truyền vào | ProductEM.java(51) |
|  | **getProductsByArtistId(int artistId)**  Input: artistId  Output: danh sách các đối tượng Product thuộc nghệ sĩ có artistId tương ứng | Lấy danh sách sản phẩm theo artistId truyền vào | ProductEM.java(59) |
|  | **searchProductsByName(String productName)**  Input: productName (tên đầy đủ hoặc 1 phần tên)  Output: danh sách các đối tượng Product có tên chứa productName | Tìm kiếm sản phẩm theo tên (tên đầy đủ hoặc 1 phần tên) | ProductEM.java(67) |
|  | **getProductsPaged(int pageNumber, int pageSize)**  Input:pageNumber, pageSize  Output: danh sách các đối tượng Product để phân trang | Lấy danh sách sản phẩm theo trang và số lượng sản phẩm trên mỗi trang | ProductEM.java(78) |
|  | **getProductsCount()**  Input: không có  Output: tổng số lượng sản phẩm trong cơ sở dữ liệu | Lấy tổng số lượng sản phẩm trên cơ sở dữ liệu | ProductEM.java(88) |
|  | **updateProduct(Product product)**  Input: product  Output: không có | Cập nhật thông tin của sản phẩm trong cơ sở dữ liệu | ProductEM.java(96) |
|  | **deleteProduct(Product product)**  Input: product  Output: không có | Xoá một sản phẩm ra khỏi cơ sở dữ liệu | ProductEM.java(102) |

Bảng 17. Các phương thức trong lớp ProductEM

2.3.15 Các phương thức trong lớp DBUtil

Sinh viên phụ trách: Nguyễn Hoài Thương

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **getEntityManager()**  Input: không có  Output: EntityManger | Thực hiện các thao tác truy vấn. lưu trữ và cập nhật dữ liệu | DBUtil.java(11) |
|  | **closeEntityManagerFactory()**  Input: không có  Output: đóng EnityManagerFactory | Giải phóng tài nguyên và ngắt kết nối với cơ sở dữ liệu | DBUtil.java(15) |

Bảng 18. Các phương thức trong lớp DBUtil

2.3.16 Các phương thức trong lớp CartController

Sinh viên phụ trách: Mai Tấn Sang

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **doGet(HttpServletRequest request, HttpServletResponse response)**  input: request thực hiện bởi máy khách và tham số action từ yêu cầu.  output: phản hồi và chuyến hướng yêu cầu đến các phần khác của ứng dụng dựa trên giá trị action. | Xử lý các hành động khác nhau dựa trên giá trị của tham số action và chuyển hướng đến các phần khác của ứng dụng như add, delete và oder. | CartController.java(51) |
|  | **handleOrder(HttpServletRequest request, HttpServletResponse response)**  input: request, response.  Output: thông báo về kết quả của quá trình đặt hàng được thiết lập và gửi đến trang cart.jsp để hiển thị cho người dùng. | Tạo Bill mới, gán thông tin người dùng và giỏ hàng cho đơn hàng này, đặt phí vận chuyển dựa trên tổng giá trị, lưu thông tin đơn hàng vào database. | CartController.java(69) |
|  | **handleDeleteCart(HttpServletRequest request, HttpServletResponse response)**  input: request,response.  Output: Cập nhật thông tin cart sau khi xóa sản phẩm. | Xóa một sản phẩm khỏi giỏ hàng theo productid được trích xuất từ request | CartController.java(148) |
|  | **handleAddCart(HttpServletRequest request, HttpServletResponse response)**  input: request, respone.  Output: Cập nhật thông tin cart sau khi thêm sản phẩm và chuyển đến trang trước đó để tiếp tục mua sắm hoặc xem giỏ hàng. | Thêm một sản phẩm ào giỏ hàng theo productid từ request | CartController.java(171) |
|  | **deleteCart(int productId, HashMap<Integer, Cart> cart)**  Input: producId, Cart.  Output: Đối tượng giỏ hàng sau khi xóa sản phầm có mã tương ứng (nếu có). | Xóa sản phẩm có productid khỏi giỏ hàng | CartController.java(203) |
|  | **addCart(int productId, HashMap<Integer, Cart> cart)**  Input: productid, cart.  Output: đối tượng giỏ hàng sau khi thêm sản phẩm mới hoặc cập nhật số lượng sản phẩm đã tồn tại. | Thực hiện thêm sản phẩm có productid và giỏ hàng, nếu sản phẩm đã tồn tại sẽ tang số lượng và cập nhật tổng giá trị của sản phẩm. | CartController.java(213) |
|  | **totalPrice(HashMap<Integer, Cart> cart)**  Input: cart  Output: số nguyên biểu diễn tổng số lượng của tất cả các sản phẩm trong giỏ hàng. | Thực hiện việc tính tổng số lượng của tất cả các sản phẩm trong giỏ hàng. | CartController.java(235) |
|  | **totalPrice(HashMap<Integer, Cart> cart)**  Input: cart.  Output:số nguyên biểu diễn tổng giá trị của tất cả các sản phẩm trong giỏ hàng. | Thực hiện việc tính tổng giá trị của tất cả các sản phầm trong giỏ hàng | CartController.java(243) |
|  | **doPost(HttpServletRequest request, HttpServletResponse response)**  Input: request, response.  Output: thực hiện hành động thau đổi, cập nhật, xóa sản phẩm trong giỏ hàng. | Xử lý các chỉnh sửa trong giỏ hàng của người dùng. | CartController.java(254) |
|  | **handleEditCart(HttpServletRequest request, HttpServletResponse response)**  Input: request, response.  Output: xử lý yêu cầu sau đó chuyển hướng người dùng đến cart.jsp để xem giỏ hàng đã cập nhật. | Xử lý yêu cầu chỉnh sửa số lượng sản phẩm giỏ hàng. | CartController.java(261) |
|  | **editCart(int productId, int quanty, HashMap<Integer, Cart> cart)**  Input: productid, quanty, cart.  Output: đối tượng giỏ hàng chứa thông tin về các sản phẩm. | Thực hiện việc cập nhật số lượng và tổng giá trị của sản phẩm nếu sản phẩm đã tồn tại trong giỏ hàng | CartController.java(285) |

Bảng 19. Các phương thức trong lớp CartController

2.3.17 Các phương thức trong lớp BillController

Sinh viên phụ trách: Mai Tấn Sang

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **doGet(HttpServletRequest request, HttpServletResponse response)**  Input: request, response, entityManage.  Output: nếu action là mybill, danh sách hóa đơn của người dùng, nếu action không xác đinh hoặc null, danh sách các hóa đơn trên manage-bill. | Xử lý action, nếu là mybill sẽ chuyển đến getBillsByUserId để lấy và hiển thị danh sách hóa đơn,nếu không có action hoặc không phải mybill sẽ lấy tất cả hóa đơn từ database thông qua jpa và hiển thị trên trang manage-bill.jsp. | BillController.java(43) |
|  | **getBillsByUserId(HttpServletRequest request, HttpServletResponse response)**  Input: request, response.  Output: Đặt danh sách hóa đơn của người dùng và thuộc tính của request để hiển thị trên trang my-bill.jsp và chuyển hướng người dùng đến đó để xem danh sách hóa đơn cá nhân. | Thực hiện lấy danh sách hóa đơn của một người dùng dựa trên thông tin đăng nhập và chuyển hướng người dùng đến trang my-bill.jsp để xem danh sách hóa đơn cá nhân. | BillController.java(78) |

Bảng 20. Các phương thức trong lớp BillController

2.3.18 Các phương thức trong lớp ArtistEM

Sinh viên phụ trách: Mai Tấn Sang

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **ArtistEM(EntityManager entityManager)**  Input:entityManager.  Output: thiết lập trạng thái ban đầu của ArtistEM | Khởi tạo đối tượng ArtistEM để tương tác với database liên quan đến Artist. | ArtistEM.java(13) |
|  | **getAllArtists()**  Input: không có.  Output: Trả về danh sách các Artist, mỗi phần tử của Artist chưa thông tin về 1 tác giả. | Cung cấp danh sách tất cả các Artist trong database. | ArtistEM.java(17) |
|  | **getArtistById(int artistId)**  Input: artistId.  Output: đối tượng Artist chưa thông tin về artistId tương ứng, nếu không tìm thấy artist trả về null. | Cung cấp một cách thuận tiện để lấy thông tin Artist từ database dựa trên artistId. | ArtistEM.java(25) |

Bảng 21. Các phương thức trong lớp ArtistEM

2.3.19 Các phương thức trong lớp CategoryEM

Sinh viên phụ trách: Mai Tấn Sang

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **CategoryEM(EntityManager entityManager)**  Input: entityManager.  Output: thiết lập trạng thái ban đầu của CategoryEM. | Khởi tạo đối tượng CategoryEM để tương tác với database liên quan đến Category. | CategoryEM.java(13) |
|  | **getAllCategories()**  Input:không có.  Output: Trả về danh sách các Category, mỗi phần tử của Category chứa thông tin về category. | Cung cấp danh sách tất cả các Category trong database | CategoryEM.java(17) |
|  | **getCategoryById(int categoryId)**  Input: CategoryId.  Output: đối tượng Category chứa thông tin về CategoryId tương ứng, nếu không tìm thấy trả về null. | Cung cấp một cách thuận tiện để lấy thông tin Category từ database dựa trên CategoryId. | CategoryEM.java(25) |

Bảng 22. Các phương thức trong lớp CategoryEM

2.3.20 Các phương thức trong lớp BillEM

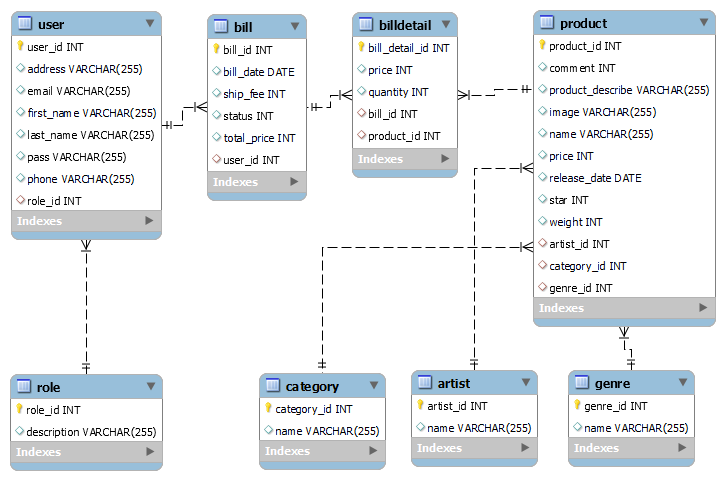
Sinh viên phụ trách: Mai Tấn Sang

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  | **saveBillAndDetails(Bill bill, Iterable<BillDetail> billDetails)**  Input:bill, billDetails.  Output: thực hiện lưu thông tin đơn hàng và chi tiết đơn hàng vào database thông qua entityManage | Lưu thông tin về một đơn hàng và các chi tiết của đơn hàng đó vào database. | BillEM.java(20) |
|  | **getAllBills()**  Input: không có.  Output: danh sách đối tượng Bill, mỗi phần tử Bill chứa thông tin về Bill. | Cung cấp danh sách tất cả các hóa đơn có sãn trong database | BillEM.java(43) |
|  | **getBillsByUserId(int userId)**  Input: userId Output: danh sách các đối tượng Bill, mỗi đối tượng đại diện cho một hóa đơn người dùng có userId tương ứng | Cung cấp danh sách các hóa đơn liên quan đến người dùng dựa trên userId. | BillEM.java(52) |

Bảng 23. Các phương thức trong lớp BillEM

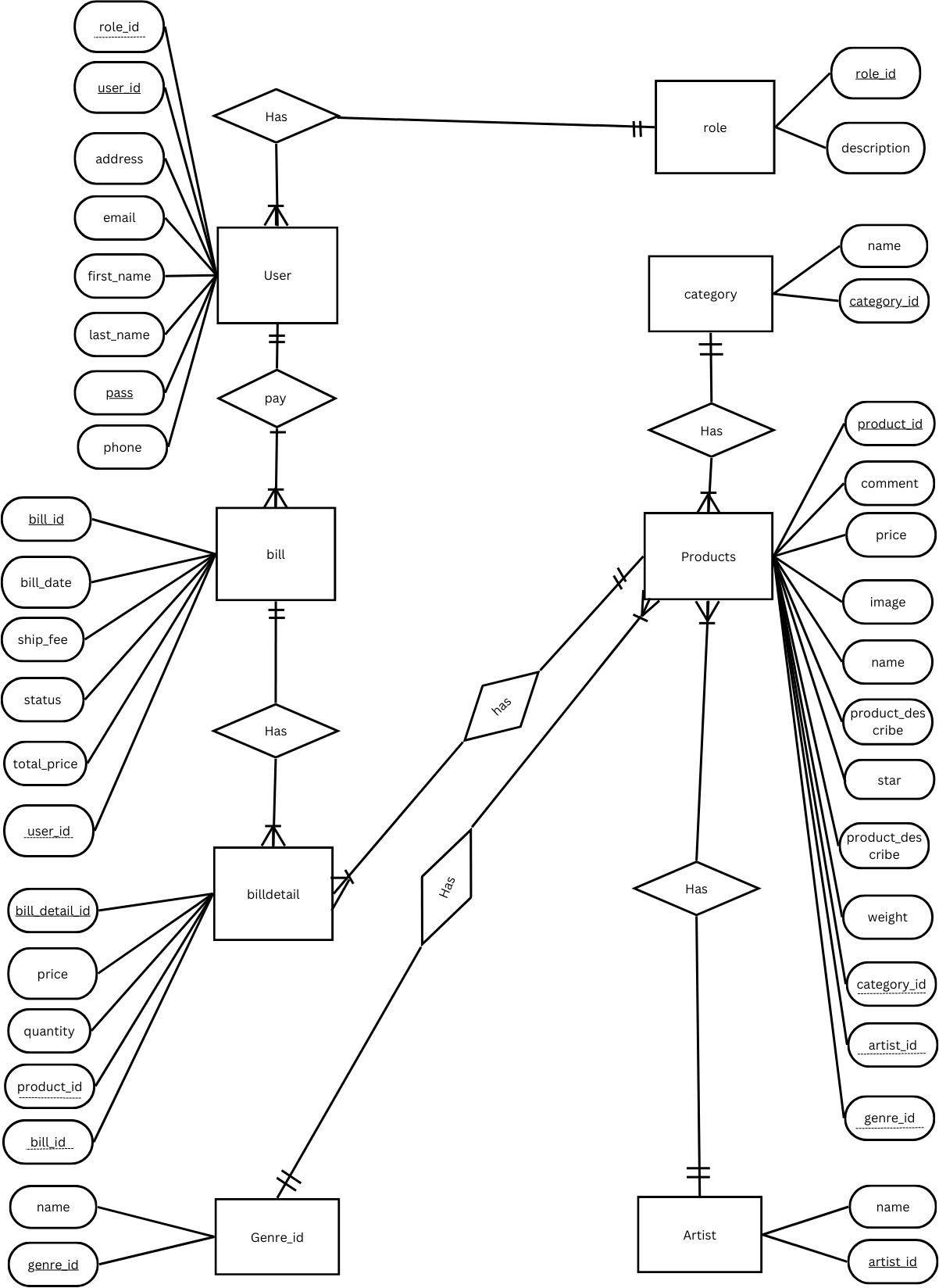
1. Thiết kế dữ liệu

3.1 Mô hình liên kết thực thể

****

Hình 3. Mô hình liên kết thực thể

3.2 Mô hình quan hệ ERD

****

Hình 4. Mô hình quan hệ ERD

3.3 Bảng mô tả các table trong CSDL

|  |  |  |
| --- | --- | --- |
| TT | Tên bảng | Mục đích |
|  | bill | Lưu trữ thông tin về ngày đặt, phí ship, trạng thái đơn hàng, tổng giá. |
|  | billdetail | Lưu trữ thông tin về giá, số lượng. |
|  | artist | Lưu thông tin tên tác giả. |
|  | category | Lưu thông tin hạng mục. |
|  | genre | Lưu thông tin thể loại. |
|  | role | Lưu thông tin về vai trò |
|  | product | Lưu thông tin về mô tả sản phẩm, bình luận, hình ảnh , tên , giá, ngày phát hành, đánh giá, cân nặng, tác giảm hạng mục, thể loại. |
|  | user | Lưu thông tin địa chỉ, email, tên , mật khẩu, số điện thoại và vai trò. |

Bảng 24 Bảng mô tả các table trong CSDL

1. Bảng mô tả các field trong table

4.1 bill

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên field | Kiểu dữ liệu | Mục đích |
|  | bill\_id | int NOT NULL AUTO\_INCREMENT, PRIMARY KEY | Id duy nhất của bill dùng để cập nhật thông tin. |
|  | bill\_date | date | Ngày mà khách hàng đặt hàng. |
|  | ship\_fee | int | Giá tiền ship |
|  | status | int | Trạng thái đơn hàng |
|  | total\_price | int | Tổng giá tiền |
|  | user\_id | int FOREIGN KEY | Id duy nhất của khách hàng |

Bảng 25. Bảng mô tả các field trong bảng bill

4.2 billdetail

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên field | Kiểu dữ liệu | Mục đích |
|  | bill\_detail\_id | int NOT NULL AUTO\_INCREMENT, PRIMARY KEY | Id duy nhất của bill detail để cập nhật thông tin |
|  | price | int | Lưu giá sản phẩm |
|  | quantity | int | Lưu số lượng sản phẩm |
|  | bill\_id | int FOREIGN KEY | Id duy nhất của bill |
|  | product\_id | int FOREIGN KEY | Id duy nhất của sản phẩm |

Bảng 26. Bảng mô tả các field trong bảng billdetail

4.3 artist

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên field | Kiểu dữ liệu | Mục đích |
|  | artist\_id | int NOT NULL AUTO\_INCREMENT, PRIMARY KEY | Lưu id duy nhất của artist để cập nhật thông tin. |
|  | name | varchar | Lưu tên của artist. |

Bảng 27. Bảng mô tả các field trong bảng artist

4.4 category

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên field | Kiểu dữ liệu | Mục đích |
|  | category\_id | int NOT NULL AUTO\_INCREMENT, PRIMARY KEY | Lưu id duy nhất của category để cập nhật thông tin |
|  | name | varchar | Lưu tên của category |

Bảng 28. Bảng mô tả các field trong bảng category

4.5 genre

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên field | Kiểu dữ liệu | Mục đích |
|  | genre\_id | int NOT NULL AUTO\_INCREMENT, PRIMARY KEY | Lưu id duy nhất của genre để cập nhật thông tin |
|  | name | varchar | Lưu tên của genre |

Bảng 29. Bảng mô tả các field trong bảng genre

4.6 product

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên field | Kiểu dữ liệu | Mục đích |
|  | product\_id | int NOT NULL AUTO\_INCREMENT,  PRIMARY KEY | Lưu id duy nhất của product để cập nhật thông tin. |
|  | comment | int | Lưu số lượng bình luận về sản phẩm. |
|  | product\_describe | varchar | Lưu mô tả về sản phẩm |
|  | image | varchar | Lưu thông tin hình ảnh |
|  | name | varchar | Lưu tên sản phẩm |
|  | price | int | Lưu giá sản phẩm |
|  | release\_date | date | Lưu ngày phát hành sản phẩm. |
|  | star | int | Lưu đánh giá sản phẩm. |
|  | weight | int | Lưu cân nặng sản phẩm |
|  | artist\_id | int FOREIGN KEY | Lưu id duy nhất của artist để cập nhật thông tin |
|  | category\_id | int FOREIGN KEY | Lưu id duy nhất của category để cập nhật thông tin |
|  | genre\_id | int FOREIGN KEY | Lưu id duy nhất của genre để cập nhật thông tin |

Bảng 30. Bảng mô tả các field trong bảng product

4.7 role

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên field | Kiểu dữ liệu | Mục đích |
|  | role\_id | int NOT NULL AUTO\_INCREMENT, PRIMARY KEY | Lưu id duy nhất của role để cập nhật thông tin |
|  | description | varchar | Lưu mô tả của role |

Bảng 31. Bảng mô tả các field trong bảng role

4.8 user

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên field | Kiểu dữ liệu | Mục đích |
|  | user\_id | int NOT NULL AUTO\_INCREMENT | Lưu id duy nhất của use để cập nhật thông tin. |
|  | address | varchar | Lưu thông tin về địa chỉ. |
|  | email | varchar | Lưu thông tin về email. |
|  | first\_name | varchar | Lưu first name của người dùng |
|  | last\_name | varchar | Lưu last name của người dùng |
|  | pass | varchar | Lưu mật khẩu. |
|  | phone | varchar | Lưu số điện thoại người dùng. |
|  | role\_id | int FOREIGN KEY | Lưu id duy nhất của role để cập nhật thông tin. |

Bảng 32. Bảng mô tả các field trong bảng user

1. Thiết kế giao diện

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Màn hình/ Cửa sổ/ Dialog | Mục đích | Giải thích ngắn gọn các quyết định thiết khi thiết kế màn hình. |
|  | Home  Hình 5. Trang home | Trang chủ của web bán hàng, quảng cáo về cửa hàng, tìm kiếm sản phẩm và liên kết đến các hạng mục chính, hiển thị sản phẩm nổi bật. | Nguyễn Hoài Thương  Mai Tấn Sang  ModeSearch: hiển thị ô tìm kiếm.  Banner Hero: sử dụng carousel trình bài quảng cáo với ảnh và mô tả.  Categories of the month: giới thiệu các hạng mục sản phẩm chính với hình ảnh và liên kết đến sản phẩm tương ứng.  Feature product: trình bày sản phẩm nổi bật, click vào sẽ dẫn họ đến trang product-details của sản phẩm đó |
|  | About    Hình 6.Trang about | Giới thiệu về cửa hàng với các dịch vụ, nghệ sĩ mà cửa hàng cung cấp | Nguyễn Hoài Thương  Mai Tấn Sang  Phần giới thiệu-About us: giới thiệu về của hàng kèm hình ảnh  Dịch vụ: hiển thị các dịch vụ mà cửa hàng cung cấp  Các nghệ sĩ ưu thích: hiển thị logo của các nghệ sĩ mà shop đang bán sản phẩm của họ, người dùng khi click vào sẽ chuyển họ đến trang shop load sản phẩm của nghệ sĩ đó |
|  | Cart  Hình 7. Trang Cart | Hiển thị giỏ hàng và đặt hàng, giao diện dễ hiểu, thân thiện. | Mai Tấn Sang  Hiển thị sản phẩm trong giỏ hàng: dùng vòng lặp để liệt kê các sản phẩm trong giỏ hàng với hình ảnh, thông tin, số lượng, giá và tùy chọn xóa sửa.  Tổng số lượng và giá tiền của giỏ hàng.  Summery:hiển thị bảng tóm tắt thông tin số lượng và giá tiền  Thông báo đặt hàng: hiển thị thông báo đặt hàng thành công hoặc thông báo giỏ hàng trống.  Nút đặt hàng để thực hiện hành động đặt hàng |
|  | My bill  Hình 8. Trang My Bill | Hiển thị đơn hàng của người dùng | Mai Tấn Sang  Hiển thị danh sách hóa đơn của người dùng, dùng vòng lặp để liệt kê thông tin từng hóa đơn.  Hiển thị trạng thái của từng hóa đơn (đang xử lý, đang giao hàng, đã hủy, thành công). |
|  | Shop  Hình 9. Trang Shop | Hiển thị danh sách sản phẩm, cung cấp các tùy chọn chọn lọc và sắp xếp giúp người dùng dễ dàng tìm kiếm và duyệt qua các sản phẩm và thương hiện khác nhau. | Nguyễn Hoài Thương  Hiển thị danh sách các hạng mục sản phẩm, nghệ sĩ.  Hiển thị sản phẩm có hình ảnh, nút xem chi tiếp và thêm vào giỏ hàng.  Bộ lọc và sắp xếp theo giá và tên.  Hiển thị các thương hiệu phổ biến |
|  | Product detail  Hình 10. Trang Product detail | Cung cấp thông tin chi tiết sản phẩm đến người dùng và thực hiện thêm vào giỏ hàng. | Nguyễn Hoài Thương  Hiển thị thông tin chi tiết về sản phẩm: hình ảnh, giá, đánh giá, mô tả, thông tin,…  Button add to cart cho phép người dùng thêm sản phẩm vào giỏ hàng. |
|  | Login  Hình 11. Trang Login | Đăng nhập và thông báo trong trường hợp có lỗi. | Nguyễn Hoài Thương  Form đăng nhập để người dùng nhập thông tin đăng nhập.  Thông báo lỗi nếu có lỗi xảy ra trong quá trình đăng nhập.  Chuyển hướng đến trang đăng ký tài khoản mới. |
|  | Register  Hình 12. Trang Register | Đăng ký tài khoản và thông báo trong trường hợp có lỗi. | Nguyễn Hoài Thương  Form đăng ký để người dùng nhập thông tin đăng ký.  Thông báo lỗi nếu có lỗi xảy ra trong quá trình đăng ký. |
|  | Profile  Hình 13. Trang Profile | Hiển thị thông tin cá nhân của người dùng. | Nguyễn Hoài Thương  Hiển thị thông tin cá nhân của người dùng đã đăng nhập như tên ảnh, email, số điện thoại.  Nút theo dõi và nhắn tin. |
|  | Header  Hình 14. Phần Header | Cung cấp các liên kết chuyển hướng đến các trang chính khác của web, thông tin và thao tác của khách hàng. | Nguyễn Hoài Thương  Mai Tấn Sang  Navbar tạo một menu liên kết đến các trang chính như home, about, shop, contact.  Hiển thị số lượng sản phẩm trong giỏ hàng.  Hiển thị thông tin người dùng: kiểm tra đăng nhập hay chưa, nêu rồi hiển thị các tùy chọn thông tin cá nhân, đơn hàng, đăng xuất, nếu chưa thì hiển thị đăng nhập.  Cho phép người dùng nhập từ khóa để tìm kiếm sản phẩm. |
|  | Footer    Hình 15. Phần Footer | Cung cấp thông tin liên lạc, dẫn người dùng đến các phần quan trọng khác của trang web, tạo điểm kết thúc cho trang web. | Nguyễn Hoài Thương  Mai Tấn Sang  Thông tin liên hệ gồm số điện thoại, email và địa chỉ của hàng.  Danh sách các hạng mục sản phẩm.  1 ô nhập địa chỉ email để người dùng nhận thông báo từ cửa hàng.  Các liên kết đến mạng xã hội của trang web. |
|  | Contact  Hình 16. Trang Contact | Kết nối với khách hàng và nhận thông tin liên lạc từ họ, tạo cho khách hàng tương tác với trang web, gửi phản hồi qua mẫu liên hệ. | Sử dụng thư viện leaflet để tích hợp bản đồ vào trang giúp hiện vị trí cửa hàng.  Tạo mẫu liên hệ gồm thông tin như Tên, email, chủ đề, nội dung.  Gửi thông tin đến server khi người dùng ấn let’talk. |
|  | Admin  Hình 17. Trang admin | Thanh điều hướng đến các trang khác của trang admin | Navbar liên kết đến các trang khác của trang quản trị. |
|  | Manage bill  Hình 18. Trang manage bill | Giúp admin dễ dàng xem quản lý hóa đơn từ khách hàng. | Hiển thị danh sách hóa đơn.  Hiển thị thông tin của từng hóa đơn.  Hiển thị trạng thái hóa đơn.  Thực hiện các hành động xóa sửa thông tin hóa đơn. |
|  | Manage product  Hình 19. Trang manage product | Quản lý danh sách thông tin sản phẩm theo từng trang. | Bảng quản lý sản phẩm: hiển thị danh sách sản phẩm trong 1 bảng và các nút thêm mới, xóa, sửa để thao tác với sản phẩm.  Phân trang: hiển thị phân trang với nhiều sản phẩm. |
|  | Add product  Hình 20. Trag add product | Thêm mới sản phẩm vào hệ thống thông qua nhập thông tin sản phẩm từ form được cung cấp. | Form thêm mới sản phẩm gồm các field tên sản phẩm, ngày phát hành, mô tả, cân nặng, giá, đánh giá, số lượng bình luận, hình ảnh, danh mục , thể loại, tác giả.  Button addproduct sử dụng để gửi thông tin sau khi người dùng điền đầy dủ thông tin vào form. |
|  | Delete product  Hình 21. Trang Delete product | Xác nhận việc xóa 1 sản phẩm cụ thể từ hệ thống. | Form xác nhận xóa sản phẩm hiện thông tin chi tiết của sản phẩm.  Button delete để gửi yêu cầu xóa sản phẩm khi người dùng xác nhận. |
|  | Edit product    Hình 22. Trang Edit Product | Xem và chỉnh sửa thông tin của sản phẩm trong hệ thống. | Form xác nhận chỉnh sửa, cập nhật thông tin sản phẩm.  Button save changes để gửi yêu cầu cập nhật thông tin sản phẩm khi người dùng thực hiện chỉnh sửa. |
|  | Manage customer    Hình 23. Trang Manage Customer | Quản lý thông tin khách hàng. | Bảng danh sách khách hàng hiển thị thông tin từng khách hàng.  Button chỉnh sửa và xóa thông tin khách hàng. |

Bảng 33. Bảng thiết kế giao diện

1. CÀI ĐẶT VÀ KIỂM THỬ

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tình huống | Mục đích | Giải thích ngắn gọn quyết định lựa chọn tình huống và dữ liệu đầu vào |
|  | Dữ liệu vào: 123  Kết quả dự kiến: thông báo sai định dạng email    Hình 24. Kiểm thử nhập sai định dạng email | Kiểm tra người dùng có nhập đúng một email hợp lệ hay không | Đảm bảo người dùng nhập đúng định dạng email |
|  | Dữ liệu vào: nhập sai email hoặc password  Kết quả dự kiến: thông báo email hoặc mật khẩu không đúng  Hình 25. Kiểm thử nhập sai email hoặc mật khẩu đã đăng kí vào cơ sở dữ liệu | Kiểm tra người dùng có nhập đúng thông tin đăng nhập không | Đảm bảo người dùng có tài khoản đã đăng kí được lưu trong cơ sở dữ liệu, tránh việc mua hàng không giao đúng hoặc thiếu sót thông tin, tránh việc khách hàng hoặc quản lí đăng nhập bị sai vai trò, dẫn đến việc nhầm lẫn không mong muốn |
|  | Dữ liệu vào: để email rỗng hoặc password rỗng và bấm đăng nhập  Kết quả dự kiến: thông báo nhập vào field đó    Hình 26. Kiểm thử bỏ trống email hoặc password | Kiểm tra người dùng có nhập email hoặc password để kiểm tra trong cơ sở dữ liệu | Đảm bảo người dùng có nhập đủ field để kiểm tra dữ liệu người dùng trong cơ sở dữ liệu |
|  | Dữ liệu vào: để các thông tin đăng kí rỗng và bấm đăng ký  Kết quả dự kiến: thông báo nhập vào field đó | Kiểm tra người dùng có nhập đầy đủ các thông tin đăng ký hay không | Đảm bảo người dùng nhập đủ thông tin đăng kí để dễ dàng kiểm soát thông tin khi cần |
|  | Dữ liệu vào: số âm  Kết quả dự kiến: thông báo phải nhập số lượng lớn hơn 0    Hình 27. Kiểm thử nhập số lượng sản phẩm là số âm | Kiểm tra xem người dùng có nhập đúng số lượng trong khoảng từ 0 trở lên hay không | Đảm bảo không xảy ra lỗi trong quá trình xử lí và đảm bảo tính nhất quán việc mua hàng trên trang web |
|  | Dữ liệu vào: bấm đặt hàng khi giỏ hàng trống  Kết quả dự kiến: thông báo giỏ hàng trống    Hình 28. Kiểm thử bấm đặt hàng với giỏ hàng trống | Kiểm tra xem người dùng việc người dùng đặt các sản phẩm | Đảm bảo không xảy ra lỗi trong quá trình lưu bill |
|  | Dữ liệu vào: một đơn hàng với tổng giá trị trên 500.000 và bấm mua hàng  Kết quả dự kiến: phí vận chuyển là 0    Hình 29. Kiểm thử việc tính phí vận chuyển cho đơn hàng | Kiểm tra xem trang web đã được thiết lặp đúng điều kiện để miễn ship hay chưa | Đảm bảo việc tính phí ship đúng với yêu cầu đặt ra |
|  | Dữ liệu vào: dán liên kết lúc thanh toán đơn hàng vào trình duyệt khác    Hình 30. Kiểm thử sao chép liên kết khi thanh toán đơn hàng qua một trình duyệt khác (1)  Kết quả dự kiến: thông báo lỗi không có người dùng đăng nhập hoặc giỏ hàng trống (kể cả bên trình duyệt kia có đăng nhập hay không)    Hình 31. Kiểm thử sao chép liên kết khi thanh toán đơn hàng qua một trình duyệt khác (2) | Kiểm tra xem khi sao chép liên kết khi thanh toán qua một trình duyệt khác xem có đặt tương tự thông tin đơn hàng bên nguồn hay không | Đảm bảo việc thanh toán đơn hàng chỉ diễn ra trên session hiện tại đang đăng nhập, khi sao chép liên kết qua session khác không lưu cùng 1 đơn hàng. |

Bảng 34. Cài đặt và kiểm thử

1. KẾT LUẬN

Trải qua quá trình thực hiện đồ án CNTT, chúng em tự đánh giá mức độ hoàn thành đồ án này là trên 97% vì đã dành nhiều công sức và thời gian để hoàn thiện đồ án. Tuy vậy, đồ án vẫn còn tồn tại một vài thiếu sót so với một website thương mại hoàn chỉnh nổi tiếng khác như shopee.vn, lazada.vn…

Một trong những khó khăn lớn nhất mà bọn em gặp phải là tìm kiếm thư viện và thiết kế giao diện phù hợp với đề tài đã đăng kí vì bọn em muốn làm cho “***website bán sách, băng đĩa và ấn phẩm thần tượng trực tuyến”*** này trở nên thân thiện hơn so với người sử dụng, muốn tạo ra một trang web mang tính thẩm mĩ cao. Việc này đòi hỏi chúng em phải tham khảo qua rất nhiều nguồn template khác nhau rồi đưa ra lựa chọn cuối cùng. Về template bọn em đã chọn template mẫu của templatemo.com và tham khảo cách thức hoạt động của các trang web liên quan đến đề tài như hangdiathoidai.vn, withmuu.com, ktown4u.com, tiki.vn…

Dù đánh giá là hoàn thành trên 97% nhưng vẫn còn tồn tại một số chức năng, một số button chưa thật sự được sử dụng trong web này. Điều này một phần ảnh hưởng đến hiệu suất và ảnh hưởng của người dùng. Bọn em đã rất cố gắng để hoàn thiện trang web hơn nhưng vẫn chưa đủ kiến thức để có thể vận dụng, cũng như là không đủ thời gian để hoàn thiện đồ án này hơn.

Bên cạnh những khuyết điểm trên thì trang web cũng sở hữu những ưu điểm lớn. Ưu điểm lớn nhất của đồ án này là đánh vào tính thực tiễn hiện nay, nhu cầu mua sách, tạp chí, đĩa cd, album KPOP, JPOP, VPOP, USUK trong giới trẻ hiện nay rất cao. Có thể thấy ngày càng nhiều fanpage trên các nền tảng facebook, instagram được lập ra để bán những sản phẩm này. Ngoài ra, nếu có theo dõi sẽ thấy được lượng album mà các hãng đĩa phát hành ngày càng đạt những con số khổng lồ như vài triệu bản cho một album nào đó. Nắm bắt được nhu cầu đó, thì chúng em đã xây dựng trang web đánh vào các đối tượng cửa hàng và khách hàng tiềm năng này, trang web bọn em đã xây dựng thực sự rất trực quan và giao diện dễ nhìn. Tuy vậy, vẫn còn thiếu sót về bán hàng, chưa chuyên nghiệp được như các web thương mại điện tử như hiện tại, do đó bọn em sẽ tiếp tục nghiên cứu và cải thiện trang web này hơn trong tương lai.

TÀI LIỆU THAM KHẢO

Murach, J., & Urban, M. (2014*). Murach’s Java Servlets and JSP (3rd ed.).* ISBN 978-1-890774-78-3.

TemplateMo. (2021, 10 January). Zay Shop Template. https://templatemo.com/tm-559-zay-shop