

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT

THÀNH PHỐ HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN



HCMUTE

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 SẢN 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 Phố Hồ Chí Minh, tháng 5 năm 2022

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN



HCMUTE

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 ÁN PHẨM
THÂN 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
II. PHÂN CÔNG CÔNG VIỆC	3
III. THIẾT KẾ.....	3
1. Thuật toán.....	3
1.1 Công nghệ và thư viện:.....	3
1.2 Kiến trúc ứng dụng:.....	4
1.3 Môi trường thực thi:	4
1.4 Các bước triển khai:.....	4
2. Thiết kế lớp	5
2.1 Class Diagram	5
2.2 Danh mục các lớp.....	5
2.3 Mô tả các phương thức trong một lớp.....	8
2.3.1 Các phương thức trong lớp Artist	8
2.3.2 Các phương thức trong lớp Bill	9
2.3.3 Các phương thức trong lớp Bill	12
2.3.4 Các phương thức trong lớp Category.....	14
2.3.5 Các phương thức trong lớp Cart.....	15
2.3.6 Các phương thức trong lớp Genre.....	16
2.3.7 Các phương thức trong lớp MyServletContextListener.....	17
2.3.8 Các phương thức trong lớp Product.....	18
2.3.9 Các phương thức trong lớp Role.....	22
2.3.10 Các phương thức trong lớp User.....	24
2.3.11 Các phương thức trong lớp WebController	27
2.3.12 Các phương thức trong lớp ShopController.....	29
2.3.13 Các phương thức trong lớp GenreEM.....	32

2.3.14 Các phương thức trong lớp ProductEM	33
2.3.15 Các phương thức trong lớp DBUtil.....	35
2.3.16 Các phương thức trong lớp CartController	35
2.3.17 Các phương thức trong lớp BillController	38
2.3.18 Các phương thức trong lớp ArtistEM	39
2.3.19 Các phương thức trong lớp CategoryEM.....	40
2.3.20 Các phương thức trong lớp BillEM	41
3. Thiết kế dữ liệu	42
3.1 Mô hình liên kết thực thể.....	42
3.2 Mô hình quan hệ ERD	43
3.3 Bảng mô tả các table trong CSDL.....	44
4. Bảng mô tả các field trong table	44
4.1 bill.....	44
4.2 billdetail	45
4.3 artist	45
4.4 category	45
4.5 genre	46
4.6 product.....	46
4.7 role	47
4.8 user.....	47
5. Thiết kế giao diện.....	48
IV. CÀI ĐẶT VÀ KIỂM THỬ	59
V. KẾT LUẬN	63
TÀI LIỆU THAM KHẢO.....	64

DANH MỤC CÁC HÌNH

Hình 1 Giao diện dự kiến	2
Hình 2. Class Diagram	5
Hình 3. Mô hình liên kết thực thể	42
Hình 4. Mô hình quan hệ ERD	43
Hình 5. Trang home	48
Hình 6. Trang about.....	49
Hình 7. Trang Cart.....	50
Hình 8. Trang My Bill.....	51
Hình 9. Trang Shop	51
Hình 10. Trang Product detail	52
Hình 11. Trang Login.....	52
Hình 12. Trang Register	53
Hình 13. Trang Profile	53
Hình 14. Phần Header	53
Hình 15. Phần Footer	54
Hình 16. Trang Contact.....	55
Hình 17. Trang admin	55
Hình 18. Trang manage bill.....	55
Hình 19. Trang manage product.....	56
Hình 20. Trang add product	56
Hình 21. Trang Delete product.....	57
Hình 22. Trang Edit Product	58
Hình 23. Trang Manage Customer.....	58
Hình 24. Kiểm thử nhập sai định dạng email.....	59
Hình 25. Kiểm thử nhập sai email hoặc mật khẩu đã đăng kí vào cơ sở dữ liệu	59

Hình 26. Kiểm thử bỏ trống email hoặc password.....	60
Hình 27. Kiểm thử nhập số lượng sản phẩm là số âm	61
Hình 28. Kiểm thử bấm đặt hàng với giỏ hàng trống.....	61
Hình 29. Kiểm thử việc tính phí vận chuyển cho đơn hàng.....	61
Hình 30. Kiểm thử sao chép liên kết khi thanh toán đơn hàng qua một trình duyệt khác (1)	62
Hình 31. Kiểm thử sao chép liên kết khi thanh toán đơn hàng qua một trình duyệt khác (2)	62

DANH MỤC CÁC BẢNG

Bảng 1. Bảng phân công công việc	3
Bảng 2. Danh mục các lớp	6
Bảng 3. Danh mục các lớp _ Nguyễn Hoài Thương	7
Bảng 4. Danh mục các lớp _ Mai Tấn Sang.....	8
Bảng 5. Các phương thức trong lớp Artist	9
Bảng 6. Các phương thức trong lớp Bill	13
Bảng 7. Các phương thức trong lớp Category	15
Bảng 8. Các phương thức trong lớp Genre	16
Bảng 9. Các phương thức trong lớp Genre	17
Bảng 10. Các phương thức trong lớp MyServletContextListener	18
Bảng 11. Các phương thức trong lớp Product.....	22
Bảng 12. Các phương thức trong lớp Role.....	23
Bảng 13. Các phương thức trong lớp User.....	26
Bảng 14 Các phương thức trong lớp WebController	28
Bảng 15. Các phương thức trong lớp ShopController	32
Bảng 16. Các phương thức trong lớp GenreEM	33
Bảng 17. Các phương thức trong lớp ProductEM.....	35
Bảng 18. Các phương thức trong lớp DBUtil	35
Bảng 19. Các phương thức trong lớp CartController	38
Bảng 20. Các phương thức trong lớp BillController.....	39
Bảng 21. Các phương thức trong lớp ArtistEM	40
Bảng 22. Các phương thức trong lớp CategoryEM	40
Bảng 23. Các phương thức trong lớp BillEM	41
Bảng 24 Bảng mô tả các table trong CSDL	44
Bảng 25. Bảng mô tả các field trong bảng bill.....	44

Bảng 26. Bảng mô tả các field trong bảng billdetail.....	45
Bảng 27. Bảng mô tả các field trong bảng artist	45
Bảng 28. Bảng mô tả các field trong bảng category	45
Bảng 29. Bảng mô tả các field trong bảng genre	46
Bảng 30. Bảng mô tả các field trong bảng product.....	47
Bảng 31. Bảng mô tả các field trong bảng role.....	47
Bảng 32. Bảng mô tả các field trong bảng user	47
Bảng 33. Bảng thiết kế giao diện	58
Bảng 34. Cài đặt và kiểm thử.....	62

I. 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, xóa dữ liệu, quản lý 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 lý cửa hàng có thể quản lý việc thêm, sửa, xóa sản phẩm để cho khách hàng đặt hàng, bên cạnh đó người quản lý cũng có thể xem các đơn đặt hàng bởi khách hàng và quản lý 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 lý 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 ký tài khoản.

Trong trang web này, khách hàng có thể đặt bất kỳ 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 xóa 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 ký (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 lý 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 lý được nhập trực tiếp trong cơ sở dữ liệu). Người quản lý 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 *gửi email*, gắn các liên kết để người dùng có thể liên hệ để được hỗ trợ.

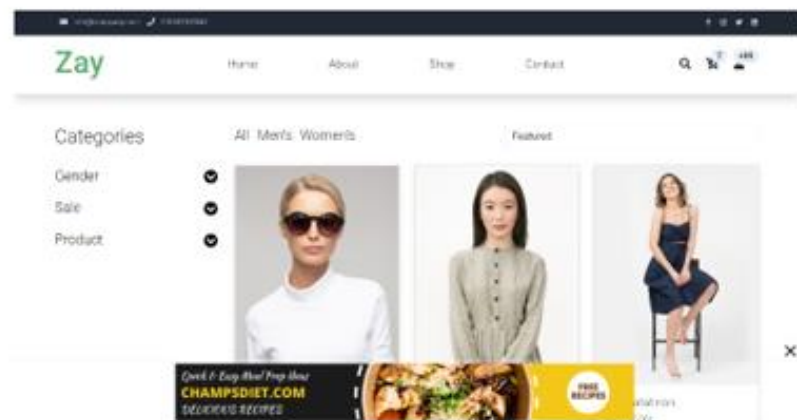
Giao diện dự kiến:

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

Đường

dẫn:

<https://templatemo.com/tm-559-zay-shop>



Hình 1 Giao diện dự kiến

II. PHÂN CÔNG CÔNG VIỆC

TT	Tên SV	Mô tả khái quát mảng công việc SV thực hiện trong đề 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 lý 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

III. THIẾT KẾ

1. 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, xóa dữ liệu, quản lý 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 lý 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 cơ 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 lý vòng đời và thực hiện truy vấn cơ sở dữ 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 lý 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 trữ và quản lý 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

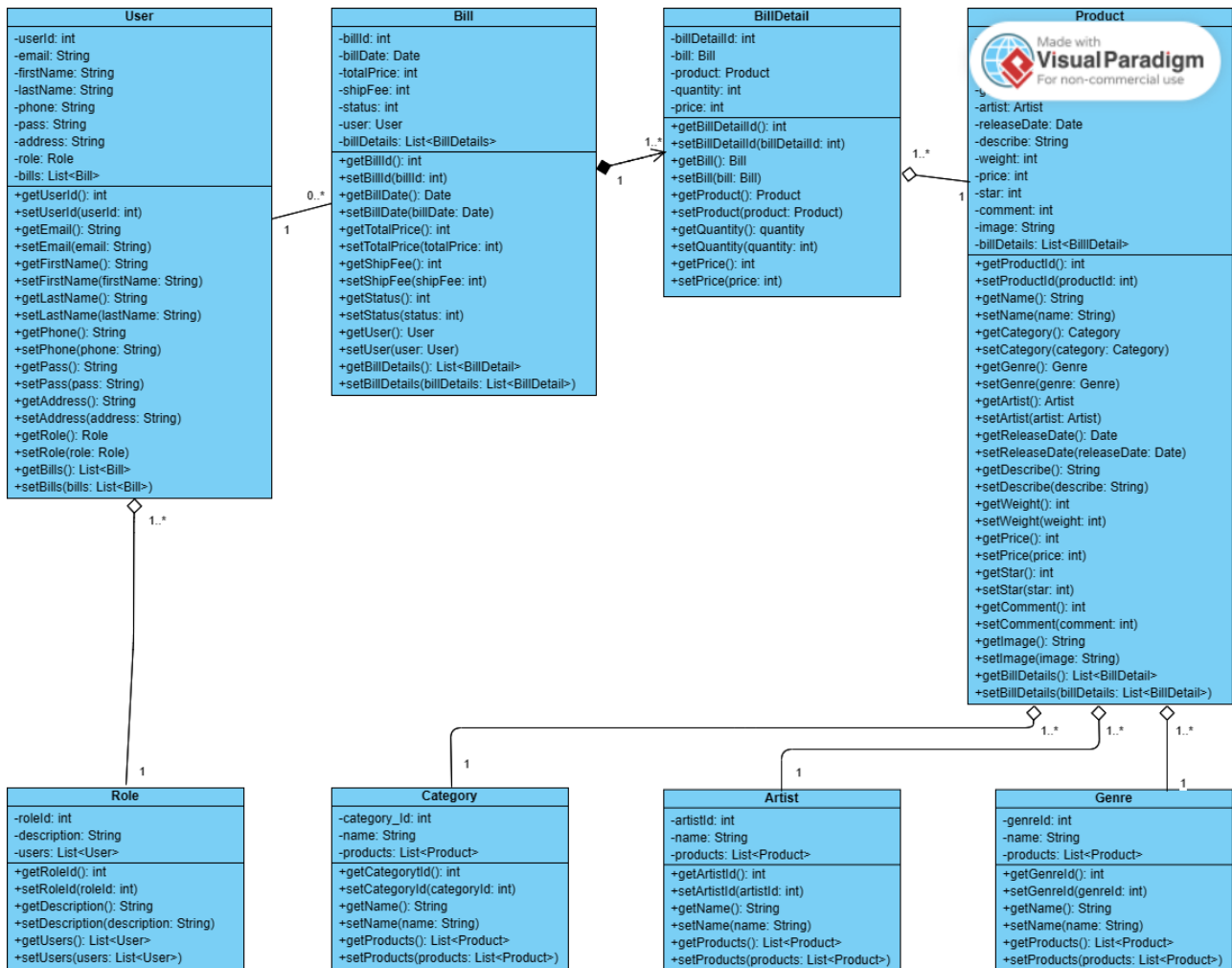
Tạo 1 DataBase Connections – MySQL và dán đường dẫn JDBC Connection ở trên và nhập các thông tin kết nối lại lần nữa

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)

2. Thiết kế lớp

2.1 Class Diagram



Hình 2. Class Diagram

2.2 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
1.	Artist	Thực hiện việc tạo bảng chứa thông tin artist.	Nguyễn Hoài Thương Mai Tấn Sang
2.	Bill	Thực hiện việc tạo bảng chứa thông tin bill.	Nguyễn Hoài Thương

			Mai Tấn Sang
3.	BillDetail	Thực hiện việc tạo bảng chứa thông tin của bill detail	Nguyễn Hoài Thương Mai Tấn Sang
4.	Category	Thực hiện việc tạo bảng chứa thông tin của category	Nguyễn Hoài Thương Mai Tấn Sang
5.	Genre	Thực hiện việc tạo bảng chứa thông tin genre.	Nguyễn Hoài Thương Mai Tấn Sang
6.	MyServletContextListener	Quản lí vòng đời của EntityManagerFactory	Nguyễn Hoài Thương Mai Tấn Sang
7.	Product	Thực hiện việc tạo bảng chứa thông tin của product	Nguyễn Hoài Thương Mai Tấn Sang
8.	Role	Thực hiện việc tạo bảng chứa thông tin của role	Nguyễn Hoài Thương Mai Tấn Sang
9.	User	Thực hiện việc tạo bảng chứa thông tin của user	Nguyễn Hoài Thương Mai Tấn Sang
10.	Cart	Thực hiện lưu trữ các thông tin liên quan đến cart.	Nguyễn Hoài Thương Mai Tấn Sang

Bảng 2. Danh mục các lớp

TT	Tên lớp Thừa kế hoặc dẫn xuất từ (ghi rõ tên lớp cơ sở) – nếu có	Mục đích
1.	WebController kế thừa từ HttpServlet	Xử lý các yêu cầu liên quan đến điều hướng website, gồm đăng nhập, đăng kí, đăng xuất và các trang thông tin khác
2.	ShopController kế thừa từ HttpServlet	Xử lý các yêu cầu liên quan đến trang Shop trong website
3.	GenreEM	Quản lý 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 lý riêng giúp duy trì tính tổ chức và tái sử dụng mã nguồn trong website
4.	ProductEM	Quản lý 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 lý riêng giúp duy trì tính tổ chức và tái sử dụng mã nguồn trong website
5.	DBUtil	Cung cấp cơ chế quản lý EntityManager và EntityManagerFactory trong môi trường JPA. Giúp quản lý 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
1.	CartController kế thừa từ HttpServlet	Hiển thị trang Cart, xóa, sửa sản phẩm. Hiển thị giá tiền, tổng giá, phí ship. Tạo Bill.
2.	BillController kế thừa từ HttpServlet	User: hiển thị mybill gồm danh sách hóa đơn Admin: hiển thị managebill gồm danh sách hóa đơn.
3.	ArtistEM.java	Thực hiện các truy vấn liên quan đến Artist trong database
4.	CategoryEM.java	Thực hiện các truy vấn liên quan đến Category trong database
5.	BilleM.java	Thực hiện các truy vấn liên quan đến Bill trong database

Bảng 4. Danh mục các lớp _ Mai Tấn Sang

2.3 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
1.	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
2.	setArtistId(int artistId) Input: artistId Output: không có mà sẽ thay đổi trạng thái của đối	Thiết lập giá trị cho artistId của đối tượng	Artist.java(33)	Nguyễn Hoài Thương Mai Tấn Sang

	tượng, đặt giá trị mới cho artistId			
3.	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
4.	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
5.	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
6.	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
1.	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

2.	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
3.	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
4.	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
5.	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
6.	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
7.	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

8.	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
9.	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
10.	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
11.	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
12.	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
13.	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
14.	setBillDetails(List<BillDetail> billDetails) Input: danh sách billDetails	Thiết lập danh sách billDetails của đối tượng	Bill.java(97)	Nguyễn Hoài Thương

	Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho danh sách billDetails			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
1.	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
2.	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
3.	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
4.	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

5.	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
6.	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
7.	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
8.	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
9.	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
10.	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
1.	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
2.	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
3.	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
4.	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
5.	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
6.	setProducts(List<Product> products)	Thiết lập giá trị cho danh sách	Category.java(49)	Nguyễn Hoài Thương

	Input: danh sách products Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho danh sách products	products của đối tượng		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
1.	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
2.	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
3.	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
4.	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

5.	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
6.	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
1.	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
2.	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
3.	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
4.	setName(String name) Input: name	Thiết lập giá trị cho name của đối tượng	Genre.java(39)	Nguyễn Hoài Thương Mai Tấn Sang

	Output: không có mà sẽ thay đổi trạng thái của đối tượng, đặt giá trị mới cho name			
5.	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
6.	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
1.	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
2.	contextDestroyed(ServletContextEvent sce) Input: sce Output: không có	Đóng EntityManagerFactory khi ứng dụng bị hủy	MyServletContextListener.java (21)	Nguyễn Hoài Thương Mai Tấn Sang

3.	EntityManagerFactory getEmFactory() Input: không có Output: emf	Cho phép các phần khác của web nhận EntityManagerFactory 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
1.	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
2.	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
3.	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
4.	setName(String name) Input: name Output: không có mà sẽ thay đổi trạng thái của đối tượng	Thiết lập giá trị cho name của đối tượng	Product.java(77))	Nguyễn Hoài Thương Mai Tấn Sang

	tượng, đặt giá trị mới cho name			
5.	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
6.	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
7.	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
8.	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
9.	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
10.	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

11.	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
12.	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
13.	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
14.	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
15.	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
16.	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

17.	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
18.	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
19.	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
20.	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
21.	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
22.	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

23.	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
24.	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
25.	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
26.	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
1.	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

2.	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
3.	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
4.	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
5.	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
6.	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
1.	getId() Input: không có Output: userId	Trả về giá trị userId cho đối tượng hiện tại	User.java(50)	Nguyễn Hoài Thương Mai Tấn Sang
2.	setId(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
3.	<b b="" 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
4.	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
5.	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
6.	setFirstName(String firstName) Input: firstName Output: không có mà sẽ thay đổi trạng thái của đối tượng	Thiết lập giá trị cho firstName của đối tượng	User.java(70)	Nguyễn Hoài Thương Mai Tấn Sang

	tượng, đặt giá trị mới cho firstName			
7.	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
8.	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
9.	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
10.	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
11.	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
12.	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

13.	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
14.	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
15.	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
16.	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
17.	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
18.	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
1.	doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException Input: <ul style="list-style-type: none">- 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)
2.	doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException Input: <ul style="list-style-type: none">- 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 lý yêu cầu HTTP POST từ client, xác định hành động dựa trên tham số action trong request và gọi các phương thức xử lý tương ứng	WebController.java(77)

3.	<p>handleLogin(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException</p> <p>Input:</p> <ul style="list-style-type: none"> - request: đối tượng chứa thông tin yêu cầu HTTP từ client - response: đối tượng cho phép Servlet tạo phản hồi đáp lại client <p>Output: không có, mà sẽ chuyển hướng trang hoặc nếu không đăng nhập thành công sẽ báo lỗi</p>	<p>Xử lý yêu cầu đăng nhập từ client. Kiểm tra thông tin đăng nhập của người dùng, xác thực trong cơ sở dữ liệu và chuyển hướng dựa trên kết quả của quá trình đăng nhập</p>	WebController.java(90)
4.	<p>handleRegister(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException</p> <p>Input:</p> <ul style="list-style-type: none"> - request: đối tượng chứa thông tin yêu cầu HTTP từ client - response: đối tượng cho phép Servlet tạo phản hồi đáp lại client <p>Output: không có, mà sẽ chuyển hướng trang hoặc nếu không đăng kí thành công sẽ báo lỗi</p>	<p>Xử lý 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 lý trường hợp email đã tồn tại trong cơ sở dữ liệu</p>	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
1.	doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException Input: <ul style="list-style-type: none">- 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 lý 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)
2.	getProductBySearchName(HttpServletRequest request, HttpServletResponse response) Input: <ul style="list-style-type: none">- request: đối tượng chứa thông tin yêu cầu HTTP từ client- response: đối tượng cho phép Servlet tạo phản hồi đáp lại client	Xử lý 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	ShopController.java(118)

	Output: không có, mà sẽ chuyển hướng trang để hiển thị danh sách sản phẩm giống với kết quả tìm kiếm	phương thức searchProductsByName từ ProductEM) và chuyển trang hiển thị kết quả tìm kiếm	
3.	getProductByCategory(HttpServletRequest request, HttpServletResponse response) Input: <ul style="list-style-type: none"> - 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 lý 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)
4.	getProductByArtist(HttpServletRequest request, HttpServletResponse response) Input: <ul style="list-style-type: none"> - request: đối tượng chứa thông tin yêu cầu HTTP từ client 	Xử lý 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	ShopController.java(199)

	<ul style="list-style-type: none"> - response: đối tượng cho phép Servlet tạo phản hồi đáp lại client <p>Output: không có, mà sẽ chuyển hướng trang để hiển thị danh sách sản phẩm theo nghệ sĩ cụ thể</p>	<p>getProductsByArtist Id từ ProductEM)</p> <p>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.</p>	
5.	<p>getDetailProduct(HttpServletRequest request, HttpServletResponse response)</p> <p>Input:</p> <ul style="list-style-type: none"> - request: đối tượng chứa thông tin yêu cầu HTTP từ client - response: đối tượng cho phép Servlet tạo phản hồi đáp lại client <p>Output: không có, mà sẽ chuyển hướng trang để hiển thị một sản phẩm cụ thể</p>	<p>Xử lý 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</p>	ShopController.java(242)
6.	<p>sortProduct(HttpServletRequest request, HttpServletResponse response)</p> <p>Input:</p> <ul style="list-style-type: none"> - request: đối tượng chứa thông tin yêu cầu HTTP từ client - response: đối tượng cho phép Servlet tạo phản hồi đáp lại client 	<p>Xử lý 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</p>	ShopController.java(272)

	Output: không có, mà sẽ chuyển hướng trang để hiển thị danh sách sản phẩm đã được sắp xếp	trang hiển thị danh sách sản phẩm đã được sắp xếp.	
7.	doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException Input: <ul style="list-style-type: none"> - 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
1.	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)
2.	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)

3.	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 find 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
1.	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)
2.	getAllProducts() Input: không có Output: danh sách các đối tượng Product lấy từ cơ sở dữ liệu	Lấy danh sách tất cả sản phẩm từ cơ sở dữ liệu	ProductEM.java(18)
3.	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)
4.	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)
5.	getProductsSortedByPriceAsc() Input: không có	Lấy danh sách sản phẩm được sắp xếp	ProductEM.java(42)

	Output: danh sách các đối tượng Product được sắp xếp theo giá từ thấp đến cao	theo giá từ thấp đến cao	
6.	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)
7.	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)
8.	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)
9.	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)
10.	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)

11.	updateProduct(Product product) Input: product Output: không có	Cập nhật thông tin của sản phẩm trong cơ sở dữ liệu	ProductEM.java(96)
12.	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
1.	getEntityManager() Input: không có Output: EntityManager	Thực hiện các thao tác truy vấn. lưu trữ và cập nhật dữ liệu	DBUtil.java(11)
2.	closeEntityManagerFactory() Input: không có Output: đóng EntityManagerFactory	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
1.	doGet(HttpServletRequest request, HttpServletResponse response) input: request thực hiện bởi máy khách và tham số action từ yêu cầu. output: phản hồi và chuyển 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 order.	CartController.java(51)

2.	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)
3.	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)
4.	handleAddCart(HttpServletRequest request, HttpServletResponse response) input: request, response. 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 vào giỏ hàng theo productid từ request	CartController.java(171)
5.	deleteCart(int productId, HashMap<Integer, Cart> cart) Input: productId, Cart. Output: Đối tượng giỏ hàng sau khi xóa sản phẩm có mã tương ứng (nếu có).	Xóa sản phẩm có productid khỏi giỏ hàng	CartController.java(203)

6.	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 tăng số lượng và cập nhật tổng giá trị của sản phẩm.	CartController.java(213)
7.	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)
8.	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 phẩm trong giỏ hàng	CartController.java(243)
9.	doPost(HttpServletRequest request, HttpServletResponse response) Input: request, response. Output: thực hiện hành động thay đổi, cập nhật, xóa sản phẩm trong giỏ hàng.	Xử lý các chỉnh sửa trong giỏ hàng của người dùng.	CartController.java(254)
10.	handleEditCart(HttpServletRequest request, HttpServletResponse response)	Xử lý yêu cầu chỉnh sửa số lượng sản phẩm giỏ hàng.	CartController.java(261)

	<p>Input: request, response.</p> <p>Output: xử lý yêu cầu sau đó chuyển hướng người dùng đến cart.jsp để xem giỏ hàng đã cập nhật.</p>		
11.	<p>editCart(int productId, int quantity, HashMap<Integer, Cart> cart)</p> <p>Input: productId, quantity, cart.</p> <p>Output: đối tượng giỏ hàng chứa thông tin về các sản phẩm.</p>	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
1.	<p>doGet(HttpServletRequest request, HttpServletResponse response)</p> <p>Input: request, response, entityManage.</p> <p>Output: nếu action là mybill, danh sách hóa đơn của người dùng, nếu action không xác định hoặc null, danh sách các hóa đơn trên manage-bill.</p>	<p>Xử lý action, nếu là mybill sẽ chuyển đến</p> <p>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.</p>	BillController.java(43)

2.	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
1.	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)
2.	getAllArtists() Input: không có. Output: Trả về danh sách các Artist, mỗi phần tử của Artist chứa thông tin về 1 tác giả.	Cung cấp danh sách tất cả các Artist trong database.	ArtistEM.java(17)
3.	getArtistById(int artistId) Input: artistId.	Cung cấp một cách thuận tiện để lấy thông tin Artist từ database dựa trên artistId.	ArtistEM.java(25)

	Output: đối tượng Artist chưa thông tin về artistId tương ứng, nếu không tìm thấy artist trả về null.		
--	---	--	--

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
1.	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)
2.	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)
3.	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*

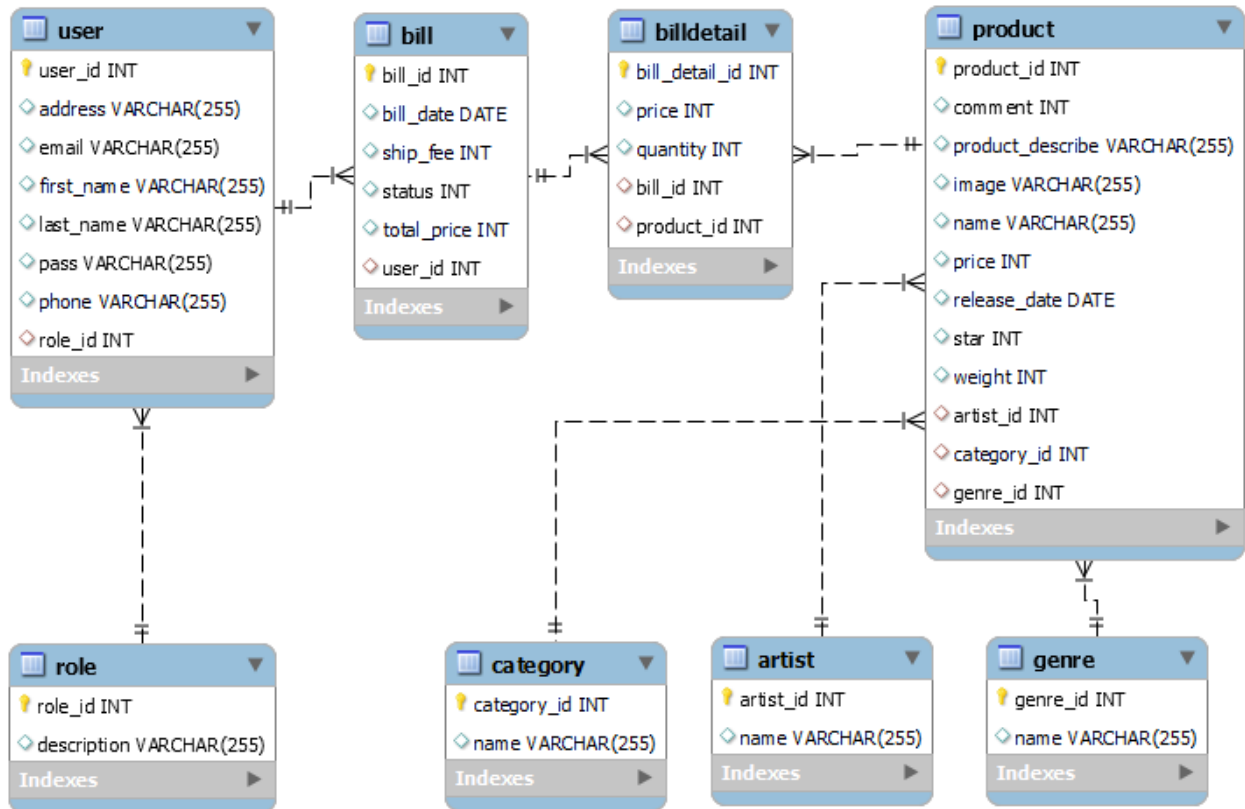
Sinh viên phụ trách: Mai Tấn Sang

TT	Phương thức	Mục đích	Tên file, số thứ tự dòng chứa khai báo
1.	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)
2.	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 sẵn trong database	BillEM.java(43)
3.	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

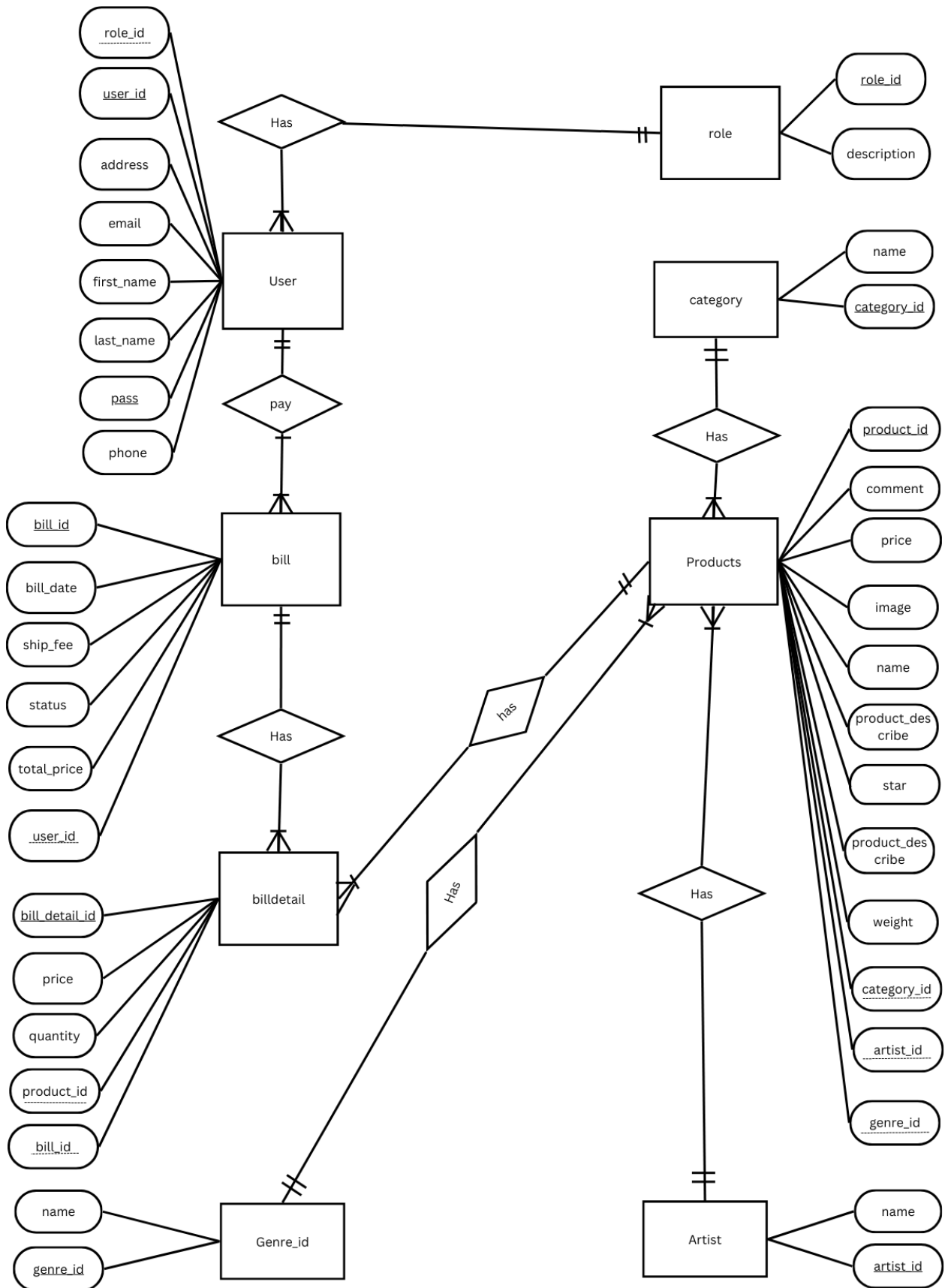
3. Thiết kế dữ liệu

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
1.	bill	Lưu trữ thông tin về ngày đặt, phí ship, trạng thái đơn hàng, tổng giá.
2.	billdetail	Lưu trữ thông tin về giá, số lượng.
3.	artist	Lưu thông tin tên tác giả.
4.	category	Lưu thông tin hạng mục.
5.	genre	Lưu thông tin thể loại.
6.	role	Lưu thông tin về vai trò
7.	product	Lưu thông tin về mô tả sản phẩm, bình luận, hình ảnh , tên , giá, ngày phát hành, đánh giá, cân nặng, tác giả, hạng mục, thể loại.
8.	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

4. Bảng mô tả các field trong table

4.1 bill

TT	Tên field	Kiểu dữ liệu	Mục đích
1.	bill_id	int NOT NULL AUTO_INCREMENT, PRIMARY KEY	Id duy nhất của bill dùng để cập nhật thông tin.
2.	bill_date	date	Ngày mà khách hàng đặt hàng.
3.	ship_fee	int	Giá tiền ship
4.	status	int	Trạng thái đơn hàng
5.	total_price	int	Tổng giá tiền
6.	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
1.	bill_detail_id	int NOT NULL AUTO_INCREMENT, PRIMARY KEY	Id duy nhất của bill detail để cập nhật thông tin
2.	price	int	Lưu giá sản phẩm
3.	quantity	int	Lưu số lượng sản phẩm
4.	bill_id	int FOREIGN KEY	Id duy nhất của bill
5.	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
1.	artist_id	int NOT NULL AUTO_INCREMENT, PRIMARY KEY	Lưu id duy nhất của artist để cập nhật thông tin.
2.	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
1.	category_id	int NOT NULL AUTO_INCREMENT, PRIMARY KEY	Lưu id duy nhất của category để cập nhật thông tin
2.	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
1.	genre_id	int NOT NULL AUTO_INCREMENT, PRIMARY KEY	Lưu id duy nhất của genre để cập nhật thông tin
2.	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
1.	product_id	int NOT NULL AUTO_INCREMENT, PRIMARY KEY	Lưu id duy nhất của product để cập nhật thông tin.
2.	comment	int	Lưu số lượng bình luận về sản phẩm.
3.	product_describe	varchar	Lưu mô tả về sản phẩm
4.	image	varchar	Lưu thông tin hình ảnh
5.	name	varchar	Lưu tên sản phẩm
6.	price	int	Lưu giá sản phẩm
7.	release_date	date	Lưu ngày phát hành sản phẩm.
8.	star	int	Lưu đánh giá sản phẩm.
9.	weight	int	Lưu cân nặng sản phẩm
10.	artist_id	int FOREIGN KEY	Lưu id duy nhất của artist để cập nhật thông tin
11.	category_id	int FOREIGN KEY	Lưu id duy nhất của category để cập nhật thông tin

12.	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
1.	role_id	int NOT NULL AUTO_INCREMENT, PRIMARY KEY	Lưu id duy nhất của role để cập nhật thông tin
2.	description	varchar	Lưu mô tả của role

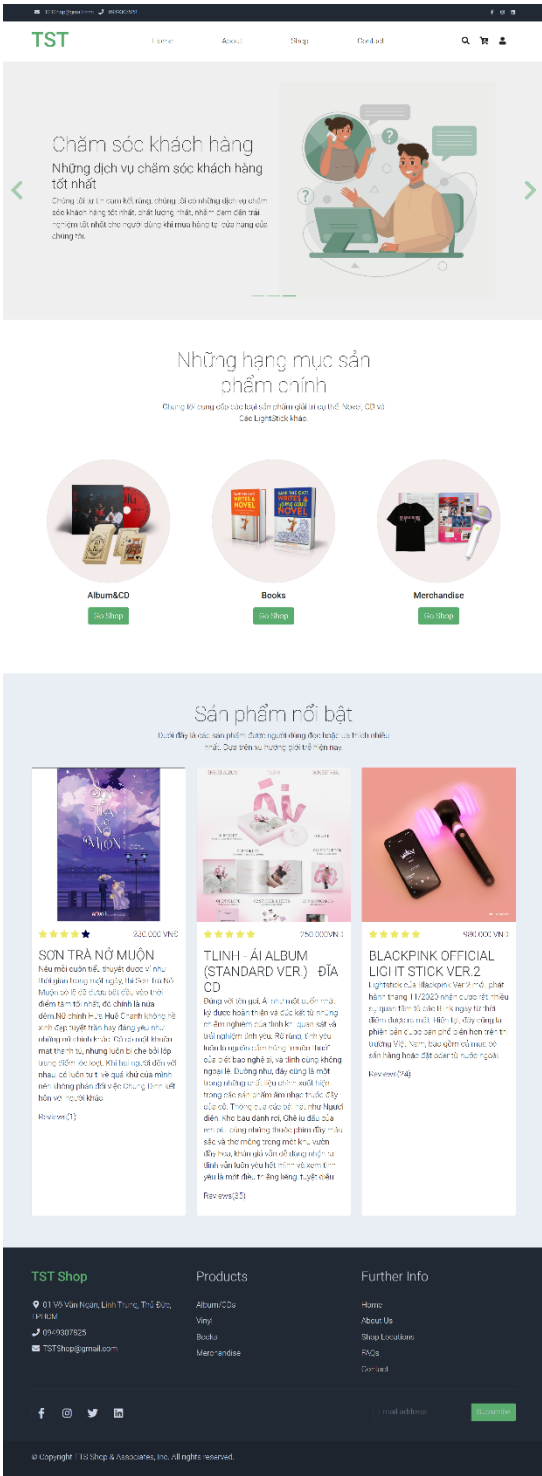
Bảng 31. Bảng mô tả các field trong bảng role

4.8 user

TT	Tên field	Kiểu dữ liệu	Mục đích
1.	user_id	int NOT NULL AUTO_INCREMENT	Lưu id duy nhất của use để cập nhật thông tin.
2.	address	varchar	Lưu thông tin về địa chỉ.
3.	email	varchar	Lưu thông tin về email.
4.	first_name	varchar	Lưu first name của người dùng
5.	last_name	varchar	Lưu last name của người dùng
6.	pass	varchar	Lưu mật khẩu.
7.	phone	varchar	Lưu số điện thoại người dùng.
8.	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

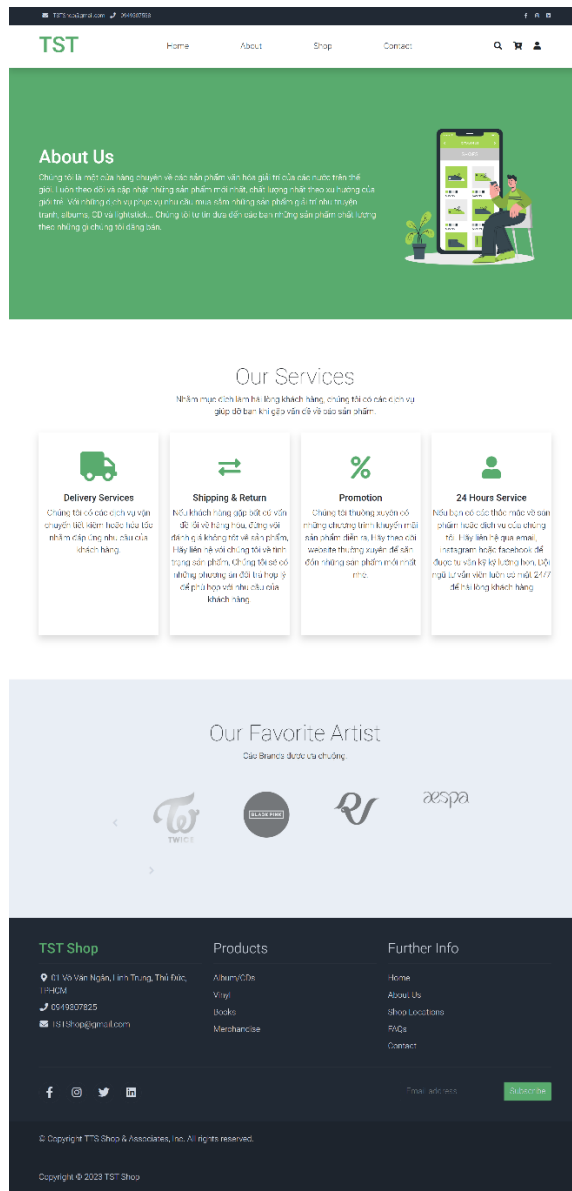
5. Thiết kế giao diện

TT	Màn hình/ Cửa sổ/ Dialog	Mục đích	Giải thích ngắn gọn các quyết định thiết kế khi thiết kế màn hình.
1.	<p>Home</p> 	<p>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.</p>	<p>Nguyễn Hoài Thương</p> <p>Mai Tấn Sang</p> <p>ModeSearch: hiển thị ô tìm kiếm.</p> <p>Banner Hero: sử dụng carousel trình bày quảng cáo với ảnh và mô tả.</p> <p>Categories of the month: giới thiệu các hạng mục sản phẩm chính với hình ảnh và liên kết đến sản phẩm tương ứng.</p> <p>Feature product: trình bày sản phẩm nổi bật, click vào sẽ dẫn họ đến trang product-details của sản phẩm đó</p>

Hình 5. Trang home

2.

About



Hình 6. Trang about

Giới thiệu về cửa hàng với các dịch vụ, nghệ sĩ mà cửa hàng cung cấp

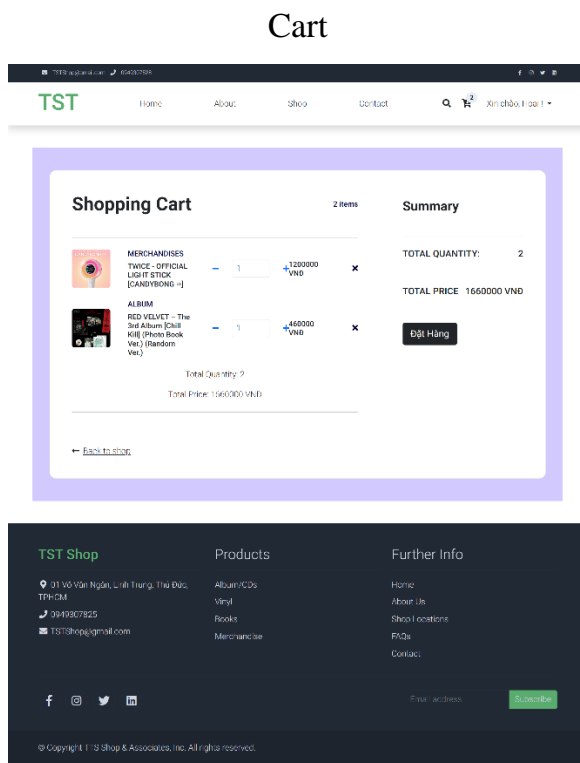
Nguyễn Hoài Thương
Mai Tấn Sang

Phần giới thiệu-About us: giới thiệu về cửa 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

3.



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.

Summary: 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

4.

My bill

STT	Mã hóa đơn	Ngày đặt hàng	Đơn giá	Phí vận chuyển	Tổng chi phí	Trạng thái
1	1	2023-12-02	2030000 VNĐ	9 VNĐ	2030009 VNĐ	Đang xử lý
2	2	2023-12-02	1880000 VNĐ	30000 VNĐ	1910000 VNĐ	Đang xử lý

TST Shop
 01 Võ Văn Ngân, Linh Trung, Thủ Đức, TP.HCM
 0949007825
 TSTShop@gmail.com

Products
 Album/CDs
 Vinyl
 Books
 Merchandise

Further Info
 Home
 About Us
 Shop Locations
 FAQs
 Contact

© Copyright TTS Shop & Associates, Inc. All rights reserved.

Hình 8. Trang My Bill

Hiện thị đơn hàng của người dùng

Mai Tấn Sang

Hiện thị danh sách hóa đơn của người dùng, dùng vòng lặp để liệt kê thông tin từng hóa đơn.

Hiện thị trạng thái của từng hóa đơn (đang xử lý, đang giao hàng, đã hủy, thành công).

5.

Shop

Categories
 All
 ALBUM
 VINYL
 BOOKS
 MERCHANDISE

Artists
 TWICE
 TILINE
 RED VELVET
 GOSPE
 TAYO SAWFT
 HOANG THUY LINH
 BLACKPINK
 NEWJEANS
 NIKKOK
 Gai Oan Chien
 Thien Trang
 Hieu Hieu Thanh
 Hoi Diem Thu Long
 Diem Luc Vo Lam
 Lieu Vu Tinh

Products
 Album/CDs
 Vinyl
 Books
 Merchandise

Further Info
 Home
 About Us
 Shop Locations
 FAQs
 Contact

Our Favorite Artist
 Các Artist được ưa chuộng
 Taylor Swift
 TWICE
 BLACKPINK
 NIKKOK

TST Shop
 01 Võ Văn Ngân, Linh Trung, Thủ Đức, TP.HCM
 0949007825
 TSTShop@gmail.com

© Copyright TTS Shop & Associates, Inc. All rights reserved.

Hình 9. Trang Shop

Hiện 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 hiệu khác nhau.

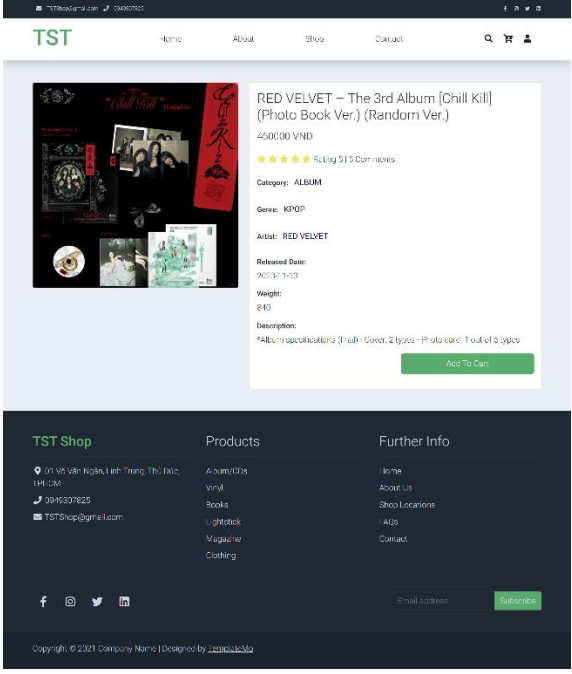
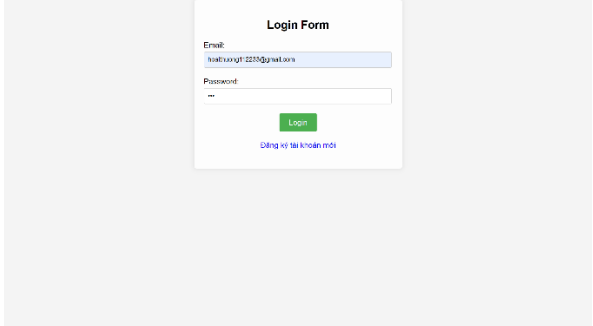
Nguyễn Hoài Thương

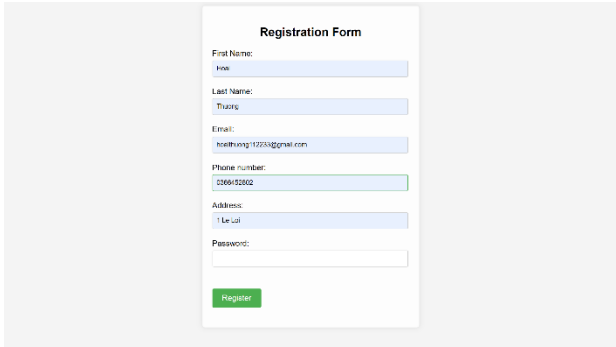
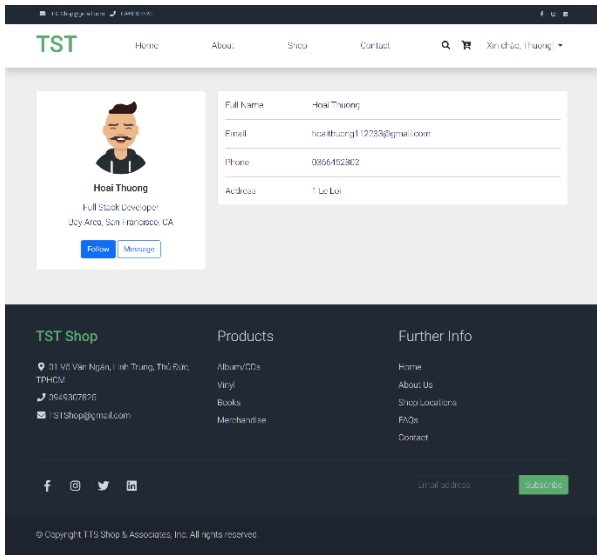
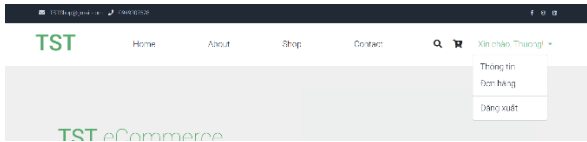
Hiện thị danh sách các hạng mục sản phẩm, nghệ sĩ.

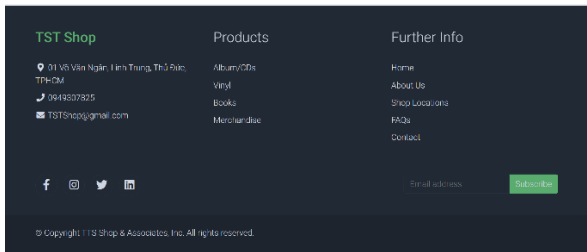
Hiện thị sản phẩm có hình ảnh, nút xem chi tiết và thêm vào giỏ hàng.

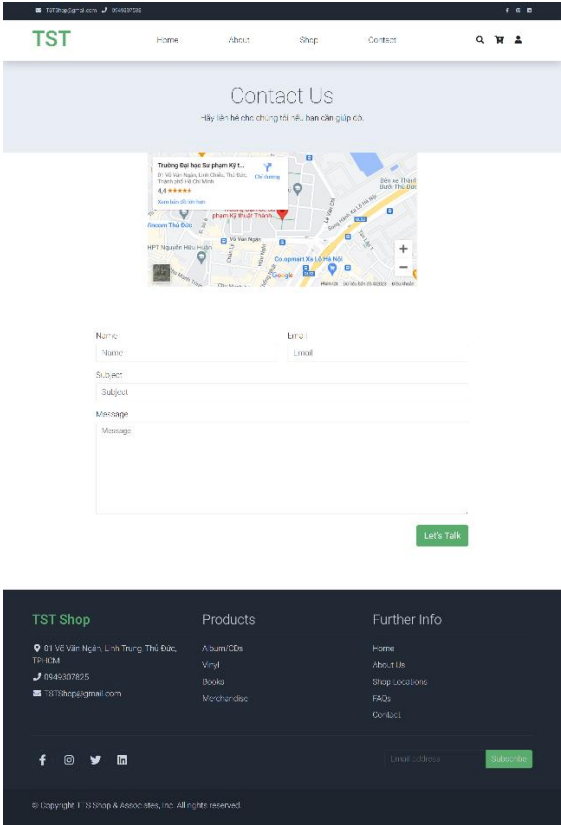
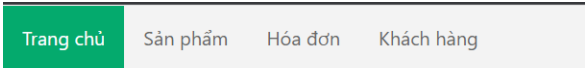

Bộ lọc và sắp xếp theo giá và tên.

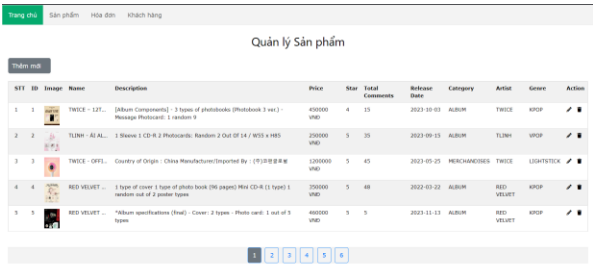
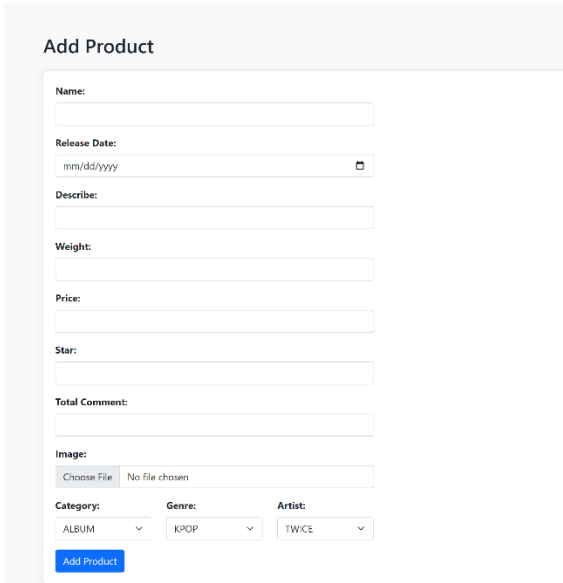
Hiện thị các thương hiệu phổ biến

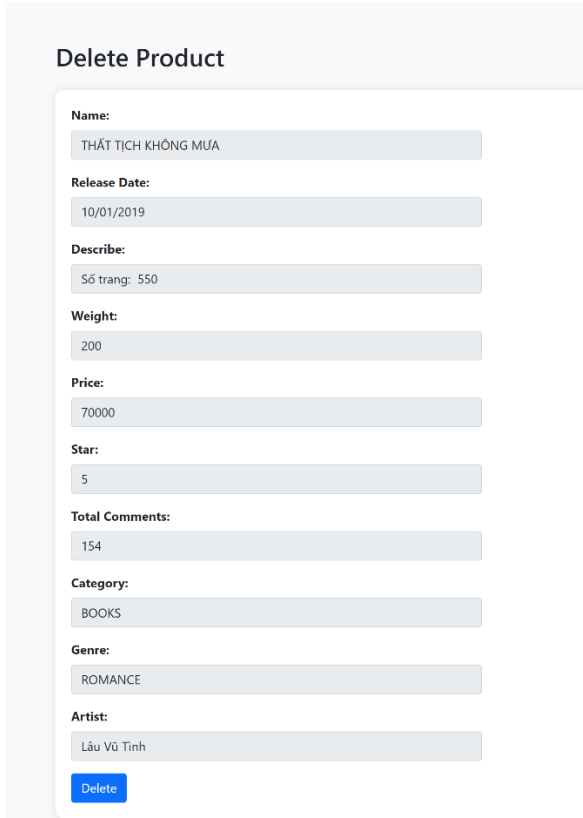
<p>6.</p>	<p style="text-align: center;">Product detail</p>  <p style="text-align: center;"><i>Hình 10. Trang Product detail</i></p>	<p>Cung cấp thông tin chi tiết sản phẩm đến người dùng và thực hiện thêm vào giỏ hàng.</p>	<p>Nguyễn Hoài Thương</p> <p>Hiển thị thông tin chi tiết về sản phẩm: hình ảnh, giá, đánh giá, mô tả, thông tin,...</p> <p>Button add to cart cho phép người dùng thêm sản phẩm vào giỏ hàng.</p>
<p>7.</p>	<p style="text-align: center;">Login</p>  <p style="text-align: center;"><i>Hình 11. Trang Login</i></p>	<p>Đăng nhập và thông báo trong trường hợp có lỗi.</p>	<p>Nguyễn Hoài Thương</p> <p>Form đăng nhập để người dùng nhập thông tin đăng nhập.</p> <p>Thông báo lỗi nếu có lỗi xảy ra trong quá trình đăng nhập.</p> <p>Chuyển hướng đến trang đăng ký tài khoản mới.</p>

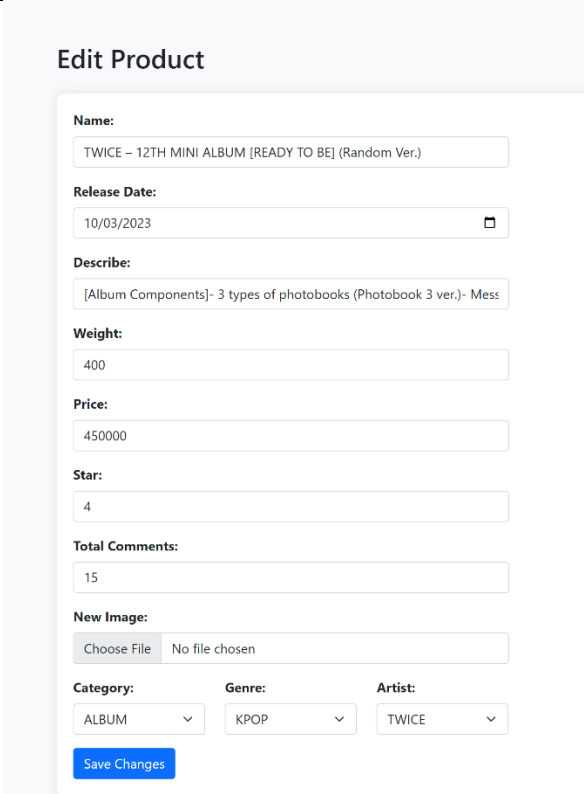

8.	<h3>Register</h3>  <p>The registration form is titled 'Registration Form'. It contains the following fields: First Name (with 'hoa' entered), Last Name (with 'thuong' entered), Email (with 'hoathuong123@gmail.com' entered), Phone number (with '0966452802' entered), Address (with '1 Le Loi' entered), and Password. A green 'Register' button is at the bottom.</p> <p><i>Hình 12. Trang Register</i></p>	<p>Đăng ký tài khoản và thông báo trong trường hợp có lỗi.</p>	<p>Nguyễn Hoài Thương</p> <p>Form đăng ký để người dùng nhập thông tin đăng ký.</p> <p>Thông báo lỗi nếu có lỗi xảy ra trong quá trình đăng ký.</p>
9.	<h3>Profile</h3>  <p>The profile page shows user information for 'Hoai Thuong'. It includes a profile picture, a bio 'Full Stack Developer, Bay Area, San Francisco, CA', and buttons for 'Follow' and 'Message'. A table displays user details: Full Name (Hoai Thuong), Email (hoathuong123@gmail.com), Phone (0966452802), and Address (1 Le Loi). Below this is a 'TST Shop' section with contact info and a 'Products' list (Album/CDs, Vinyl, Books, Merchandise). A 'Further Info' section lists links like Home, About Us, Shop Locations, FAQs, and Contact. A 'Subscribe' button is at the bottom right.</p> <p><i>Hình 13. Trang Profile</i></p>	<p>Hiển thị thông tin cá nhân của người dùng.</p>	<p>Nguyễn Hoài Thương</p> <p>Hiển thị thông tin cá nhân của người dùng đã đăng nhập như tên ảnh, email, số điện thoại.</p> <p>Nút theo dõi và nhắn tin.</p>
10.	<h3>Header</h3>  <p>The header features the 'TST' logo, navigation links (Home, About, Shop, Contact), a search icon, and a user profile icon. A dropdown menu is open for the user profile, showing 'Thông tin', 'Giỏ hàng', and 'Đăng xuất'.</p> <p><i>Hình 14. Phần Header</i></p>	<p>Cung cấp các liên kết hướng đến các trang chính khác của web, thông tin và thao tác của khách hàng.</p>	<p>Nguyễn Hoài Thương</p> <p>Mai Tấn Sang</p> <p>Navbar tạo một menu liên kết đến các trang chính như home, about, shop, contact.</p> <p>Hiển thị số lượng sản phẩm trong giỏ hàng.</p> <p>Hiển thị thông tin người dùng: kiểm tra</p>

			<p>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.</p> <p>Cho phép người dùng nhập từ khóa để tìm kiếm sản phẩm.</p>
11.	<p>Footer</p>  <p><i>Hình 15. Phần Footer</i></p>	<p>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.</p>	<p>Nguyễn Hoài Thương</p> <p>Mai Tấn Sang</p> <p>Thông tin liên hệ gồm số điện thoại, email và địa chỉ của hàng.</p> <p>Danh sách các hạng mục sản phẩm.</p> <p>1 ô nhập địa chỉ email để người dùng nhận thông báo từ cửa hàng.</p> <p>Các liên kết đến mạng xã hội của trang web.</p>

12.	 <p><i>Hình 16. Trang Contact</i></p>	<p>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.</p>	<p>Sử dụng thư viện leaflet để tích hợp bản đồ vào trang giúp hiện vị trí cửa hàng.</p> <p>Tạo mẫu liên hệ gồm thông tin như Tên, email, chủ đề, nội dung.</p> <p>Gửi thông tin đến server khi người dùng ấn let'talk.</p>
13.	 <p><i>Hình 17. Trang admin</i></p>	<p>Thanh điều hướng đến các trang khác của trang admin</p>	<p>Navbar liên kết đến các trang khác của trang quản trị.</p>
14.	 <p><i>Hình 18. Trang manage bill</i></p>	<p>Giúp admin dễ dàng xem quản lý hóa đơn từ khách hàng.</p>	<p>Hiển thị danh sách hóa đơn.</p> <p>Hiển thị thông tin của từng hóa đơn.</p> <p>Hiển thị trạng thái hóa đơn.</p>

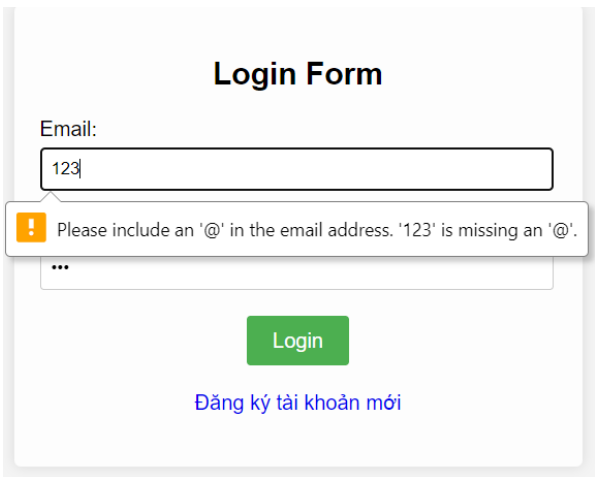
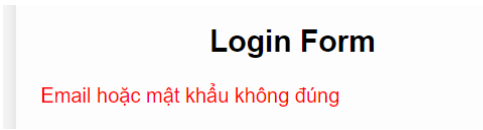
			Thực hiện các hành động xóa sửa thông tin hóa đơn.
15.	<p style="text-align: center;">Manage product</p>  <p style="text-align: center;"><i>Hình 19. Trang manage product</i></p>	Quản lý danh sách thông tin sản phẩm theo từng trang.	<p>Bảng quản lý sản phẩm: hiển thị danh sách sản phẩm trong 1 bảng và các nút thêm mới, xóa, sửa để thao tác với sản phẩm.</p> <p>Phân trang: hiển thị phân trang với nhiều sản phẩm.</p>
16.	<p style="text-align: center;">Add product</p>  <p style="text-align: center;"><i>Hình 20. Trang add product</i></p>	Thêm mới sản phẩm vào hệ thống thông qua nhập thông tin sản phẩm từ form được cung cấp.	<p>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.</p> <p>Button addproduct sử dụng để gửi thông tin sau khi người dùng điền đầy đủ thông tin vào form.</p>

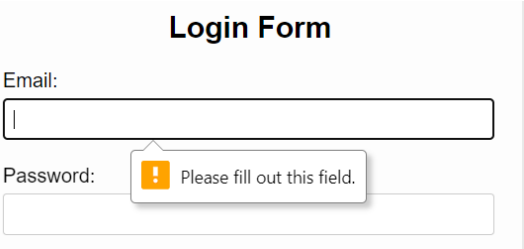
17.	<p style="text-align: center;">Delete product</p>  <p style="text-align: center;"><i>Hình 21. Trang Delete product</i></p>	Xác nhận việc xóa 1 sản phẩm cụ thể từ hệ thống.	Form xác nhận xóa sản phẩm hiện thông tin chi tiết của sản phẩm. Button delete để gửi yêu cầu xóa sản phẩm khi người dùng xác nhận.
18.	<p style="text-align: center;">Edit product</p>	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.

	 <p><i>Hình 22. Trang Edit Product</i></p>		
19.	<p>Manage customer</p>  <p><i>Hình 23. Trang Manage Customer</i></p>	<p>Quản lý thông tin khách hàng.</p>	<p>Bảng danh sách khách hàng hiển thị thông tin từng khách hàng.</p> <p>Button chỉnh sửa và xóa thông tin khách hàng.</p>

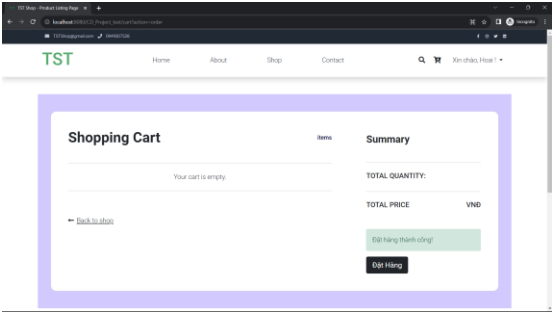
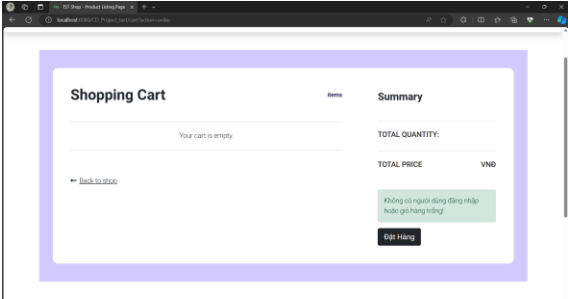
Bảng 33. Bảng thiết kế giao diện

IV. 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
1.	<p>Dữ liệu vào: 123</p> <p>Kết quả dự kiến: thông báo sai định dạng email</p>  <p><i>Hình 24. Kiểm thử nhập sai định dạng email</i></p>	Kiểm tra người dùng có nhập đúng một email hợp lệ hay không	Đảm bảo người dùng nhập đúng định dạng email
2.	<p>Dữ liệu vào: nhập sai email hoặc password</p> <p>Kết quả dự kiến: thông báo email hoặc mật khẩu không đúng</p>  <p><i>Hình 25. Kiểm thử nhập sai email hoặc mật khẩu đã đăng kí vào cơ sở dữ liệu</i></p>	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
3.	<p>Dữ liệu vào: để email rỗng hoặc password rỗng và bấm đăng nhập</p> <p>Kết quả dự kiến: thông báo nhập vào field đó</p>  <p><i>Hình 26. Kiểm thử bỏ trống email hoặc password</i></p>	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
4.	<p>Dữ liệu vào: để các thông tin đăng kí rỗng và bấm đăng ký</p> <p>Kết quả dự kiến: thông báo nhập vào field đó</p>	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
5.	<p>Dữ liệu vào: số âm</p> <p>Kết quả dự kiến: thông báo phải nhập số lượng lớn hơn 0</p>	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

	<div><div>Shopping Cart1 items</div><div><div><div><div><div><div>ALBUM</div><div><div><div>READY TO BE</div><div></div></div></div><div><div>TWICE - 12TH MINI ALBUM</div><div>[READY TO BE]</div><div>(Random)</div></div></div><div><div>-</div><div>-1</div><div>+</div></div><div><div>450000 VND</div><div>x</div></div></div><div><div>Value must be greater than or equal to 0.</div></div></div><div>Hình 27. Kiểm thử nhập số lượng sản phẩm là số âm</div></div></div></div>														
6.	<div><div>Dữ liệu vào: bấm đặt hàng khi giỏ hàng trống</div><div>Kết quả dự kiến: thông báo giỏ hàng trống</div><div><div><div>localhost:3000 says</div><div>Your cart is empty.</div><div>OK</div></div><div><div>Shopping Cart0 items</div><div><div>Your cart is empty.</div><div>Back to shop</div></div><div><div>Summary</div><div><div>TOTAL QUANTITY: 0</div><div>TOTAL PRICE 0 VND</div><div>Đặt Hàng</div></div></div></div><div>Hình 28. Kiểm thử bấm đặt hàng với giỏ hàng trống</div></div></div>	Kiểm tra xem người dùng việc người dùng đặt các sản phẩm	Đảm bảo không xảy ra lỗi trong quá trình lưu bill												
7.	<div><div>Dữ liệu vào: một đơn hàng với tổng giá trị trên 500.000 và bấm mua hàng</div><div>Kết quả dự kiến: phí vận chuyển là 0</div><div>Hóa đơn của tôi</div><div><table><tr><th>Đơn giá</th><th>Phí vận chuyển</th><th>Tổng chi phí</th></tr><tr><td>2050000 VND</td><td>0 VND</td><td>2050000 VND</td></tr><tr><td>180000 VND</td><td>30000 VND</td><td>210000 VND</td></tr><tr><td>2050000 VND</td><td>0 VND</td><td>2050000 VND</td></tr></table></div><div>Hình 29. Kiểm thử việc tính phí vận chuyển cho đơn hàng</div></div>	Đơn giá	Phí vận chuyển	Tổng chi phí	2050000 VND	0 VND	2050000 VND	180000 VND	30000 VND	210000 VND	2050000 VND	0 VND	2050000 VND	Kiểm tra xem trang web đã được thiết lập đúng điều kiện để miễn ship hay chưa	Đảm bảo việc tính phí ship đúng với yêu cầu đặt ra
Đơn giá	Phí vận chuyển	Tổng chi phí													
2050000 VND	0 VND	2050000 VND													
180000 VND	30000 VND	210000 VND													
2050000 VND	0 VND	2050000 VND													

8.	<p>Dữ liệu vào: dán liên kết lúc thanh toán đơn hàng vào trình duyệt khác</p>  <p><i>Hình 30. Kiểm thử sao chép liên kết khi thanh toán đơn hàng qua một trình duyệt khác (1)</i></p> <p>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)</p>  <p><i>Hình 31. Kiểm thử sao chép liên kết khi thanh toán đơn hàng qua một trình duyệt khác (2)</i></p>	<p>Kiểm tra xem khi sao chép liên kết khi thanh toán qua một trình duyệt khác xem có đặt đơn hàng bên nguồn hay không</p>	<p>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.</p>
----	--	---	---

Bảng 34. Cài đặt và kiểm thử

V. 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 mỹ 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>