

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**



ĐINH QUANG LINH

**XÂY DỰNG WEBSITE QUẢN LÝ TOUR DU LỊCH
SỬ DỤNG SPRING MVC**

**KHÓA LUẬN TỐT NGHIỆP
NGÀNH KHOA HỌC MÁY TÍNH**

TP. HỒ CHÍ MINH, 2022

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH



ĐINH QUANG LINH

NGHIÊN CỨU SPRING MVC TRONG LẬP TRÌNH
JAVA ĐỂ XÂY DỰNG WEBSITE TOUR DU LỊCH

Mã số sinh viên: 1851010067

KHÓA LUẬN TỐT NGHIỆP
NGÀNH KHOA HỌC MÁY TÍNH

Giảng viên hướng dẫn: Ths. Nguyễn Thị Mai Trang

TP. HỒ CHÍ MINH, 2022

TRƯỜNG ĐẠI HỌC MỞ CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
THÀNH PHỐ HỒ CHÍ MINH **Độc lập – Tự do – Hạnh phúc**
KHOA CÔNG NGHỆ THÔNG TIN

GIẤY XÁC NHẬN

Tôi tên là: ĐINH QUANG LINH

Ngày sinh: 08/02/2000

Nơi sinh: Kon Tum

Chuyên ngành: Khoa học máy tính

Mã sinh viên: 1851010067

Tôi đồng ý cung cấp toàn văn thông tin khóa luận tốt nghiệp hợp lệ về bản quyền cho Thư viện Trường Đại học Mở Thành phố Hồ Chí Minh. Thư viện Trường Đại học Mở Thành phố Hồ Chí Minh sẽ kết nối toàn văn thông tin khóa luận tốt nghiệp vào hệ thống thông tin khoa học của Sở Khoa học và Công nghệ Thành phố Hồ Chí Minh.

Ký tên
(Ghi rõ họ và tên)

.....

**Ý KIẾN CHO PHÉP BẢO VỆ KHÓA LUẬN
CỦA GIẢNG VIÊN HƯỚNG DẪN**

Giảng viên hướng dẫn:

Sinh viên thực hiện: **Lớp:**

Ngày sinh: **Nơi sinh:**

Tên đề tài:

.....

.....

.....

.....

Ý kiến của giảng viên hướng dẫn về việc cho phép sinh viên được bảo vệ khóa luận trước Hội đồng:

.....

.....

.....

.....

.....

.....

.....

.....

.....

Thành phố Hồ Chí Minh, ngày ... tháng ... năm

Người nhận xét

Nguyễn Thị Mai Trang

LỜI CẢM ƠN

Sau một thời gian được vinh dự ngồi trên ghế nhà trường khóa luận tốt nghiệp này là một minh chứng cho những kiến thức và kinh nghiệm em đã học được tại môi trường này. Cùng với những lời động viên và giúp đỡ nhiệt tình từ các thầy cô và bạn bè em đã cố gắng hết sức để hoàn thành khóa luận tốt nghiệp.

Em xin chân thành cảm ơn Ban giám hiệu nhà trường Đại học Mở Thành phố Hồ Chí Minh và các thầy cô khoa công nghệ thông tin. Đặc biệt em xin chân thành cảm ơn Ths. Nguyễn Thị Mai Trang, một người cô tràn đầy nhiệt huyết, hết sức tận tâm trong công việc, luôn đồng hành giúp đỡ em và các bạn, không những giải đáp mọi thắc mắc cô còn góp ý cho khóa luận của em xử lý được tốt hơn, bên cạnh đó cô còn trao đổi, định hướng phát triển.

Trong quá trình thực hiện khóa luận tuy còn những khuyết điểm và hạn chế mà em chưa khắc phục được. Vì vậy em rất mong sự giúp đỡ và góp ý từ Quý thầy cô để có thêm kinh nghiệm, khắc phục những khuyết điểm để hoàn thiện bản thân mình hơn, góp một phần công sức nhỏ vào những việc có ích cho xã hội sau này.

Thành phố Hồ Chí Minh, ngày 10 tháng 05 năm 2022

Người thực hiện

Đinh Quang Linh

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

TÓM TẮT KHÓA LUẬN

1. Lý do chọn đề tài

Hiện nay nhu cầu đi du lịch và tìm kiếm các địa điểm du lịch thú vị cần phải nhanh chóng, dễ dàng, thuận tiện cho việc đặt vé. Để giải quyết những nhu cầu người dùng mong muốn bên cạnh đó còn quảng bá hình ảnh, văn hóa nước ta cho nhiều người trong và ngoài nước biết tới nhiều hơn, giúp cho ngành du lịch được phát triển mạnh mẽ hơn vì thế việc xây dựng các ứng dụng, website hiện tại là xu hướng.

2. Các vấn đề nghiên cứu

- Các công nghệ cần dùng để thực hiện ứng dụng gồm có: Spring MVC, Front Controller Design Pattern, Cloudinary, Hibernate, JSP, Bootstrap 4.0.
- Giới thiệu Spring framework, đơn giản và tiện dụng hơn khi sử dụng Cloudinary, các thành phần của Hibernate, sử dụng JSP thay vì sử dụng HTML.

3. Kết quả

Trải qua một khoảng thời gian tìm hiểu và thực hiện đề tài, ứng dụng đã giải quyết các chức năng cơ bản ban đầu đã đặt ra, đồng thời cũng đặt ra những vấn đề mới nhằm hoàn thiện ứng dụng. Dựa vào sự nghiên cứu học hỏi, cùng với hướng dẫn của thầy cô đề tài đã hoàn thành được bao gồm các mục tiêu như sau:

- Xác định và phân quyền chia quyền cơ bản cho 3 đối tượng người dùng gồm admin, nhân viên, khách hàng.
- Hiểu được cách thức hoạt động của ứng dụng.
- Xác định và áp dụng được các công nghệ trong việc xây dựng ứng dụng.
- Hoàn thành các mục tiêu đã đề ra.

4. Hướng phát triển đề tài

Việc xây dựng một ứng dụng đặt tour du lịch ngày càng phát triển hơn, số lượng người đi du lịch cũng vì vậy tăng lên, bên cạnh đáp ứng nhu cầu và thúc đẩy cho việc đi du lịch. Việc tích hợp tính năng thanh toán trực tuyến và kết hợp với những địa điểm nổi tiếng để xây dựng hình ảnh đẹp cho nước nhà.

5. Bố cục khóa luận

Nội dung báo cáo đề nghiên cứu Spring MVC trong lập trình java để xây dựng website đặt tour du lịch bao gồm 4 chương:

Chương 1: Giới thiệu đề tài: Chương 1 là phần tìm hiểu chung về ảnh hưởng của du lịch khi kết hợp với công nghệ thông tin, đưa ra lý do để chọn đề tài để xây dựng website, các mục tiêu khi thực hiện đề tài và các đối tượng cần nghiên cứu trong đề tài.

Chương 2: Cơ sở lý thuyết: Chương 2 là phần trình bày giới thiệu về các công cụ được sử dụng trong hệ thống gồm giới thiệu Spring framework, các thành phần trong Spring framework, đặc biệt giới thiệu và hướng dẫn xây dựng website với Spring MVC và các công nghệ hỗ trợ khác cần thiết khi xây dựng hệ thống này như Front Controller Design Pattern, Cloudinary, Hibernate, JSP.

Chương 3: Hệ thống đặt tour du lịch: Chương 3 tập trung vào việc phân tích các thành phần chính của hệ thống, giao diện hệ thống và thiết kế cơ sở dữ liệu trong đó việc mô tả tổng quát về các chức năng quan trọng của hệ thống cùng với các sơ đồ hoạt động của chúng và các thành phần thiết kế cơ sở dữ liệu trong hệ thống là các thành phần chính.

Chương 4: Kết luận và hướng phát triển: Chương 4 là phần tổng kết lại những kết quả đã đạt được trong quá trình thực hiện xây dựng hệ thống, bên cạnh đó là các mặt hạn chế của hệ thống cần được khắc phục, nâng cấp đồng thời đưa ra hướng phát triển trong thời gian tới.

TÓM TẮT KHÓA LUẬN

(PHIÊN BẢN TIẾNG ANH)

Currently, the need to travel and search for interesting tourist destinations needs to be fast, easy, and convenient for booking tickets. In order to address the needs of users, besides promoting the image and culture of our country to more people at home and abroad, and helping the tourism industry to develop more strongly, the construction of current applications and websites is a trend. The construction of a tour booking app is growing more and more, and the number of people traveling also increases, besides meeting the demand and promoting travel. The integration of online payment features and combined with famous locations to build a beautiful image for the country.

Spring MVC, Front Controller Design Pattern, Cloudinary, Hibernate, JSP, and Bootstrap 4.0 are the technologies applied to build this website.

Over some time learning and implementing the topic, the application has solved the basic functions that were originally set, while also posing new problems to perfect the application. Based on the study, along with the instructor's instructions, the completed topic includes the following objectives:

- Identify and decentralize basic permissions among three user audiences including admins, employees, and customers.
- Understand how the application works.
- Identify and apply technologies in the construction of applications.
- Accomplish the set goals.

MỤC LỤC

LỜI CẢM ƠN	3
NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN.....	4
TÓM TẮT KHÓA LUẬN.....	5
TÓM TẮT KHÓA LUẬN (PHIÊN BẢN TIẾNG ANH)	7
DANH MỤC TỪ VIẾT TẮT	10
DANH MỤC HÌNH VẼ	11
DANH MỤC BẢNG	12
MỞ ĐẦU.....	13
Chương 1. GIỚI THIỆU ĐỀ TÀI.....	14
1.1. Giới thiệu đề tài và lý do chọn đề tài	14
1.2. Mục tiêu của đề tài	14
1.3. Phạm vi và đối tượng nghiên cứu	15
Chương 2. CƠ SỞ LÝ THUYẾT	16
2.1. Spring framework.....	16
2.1.1. Giới thiệu Spring framework.....	16
2.1.2. Giới thiệu Spring MVC	18
2.1.3. Xây dựng Website với Spring MVC	18
2.2. Front Controller Design Pattern	24
2.3. Cloudinary	25
2.3.1. Cài đặt Cloudinary.....	25
2.3.1. Sử dụng Cloudinary.....	26
2.4. Kiến trúc Hibernate	29
2.4.1. Các đối tượng trong kiến trúc Hibernate	29
2.5. JSP	33
2.6. Bootstrap	35

2.6.1.	Giới thiệu Bootstrap	35
2.6.2.	Cài đặt và sử dụng Bootstrap	35
Chương 3.	HỆ THỐNG ĐẶT TOUR DU LỊCH	38
3.1.	Phân tích hệ thống.....	38
3.1.1.	Mô tả chức năng.....	38
3.1.2.	Sơ đồ hoạt động.....	40
3.2.	Giao diện hệ thống	44
3.2.1.	Giao diện chung.....	44
3.2.2.	Giao diện chung của admin và nhân viên	48
3.2.3.	Giao diện admin	51
3.3.	Thiết kế cơ sở dữ liệu.....	52
3.3.1.	Các bảng	52
3.3.2.	Chi tiết các bảng dữ liệu.....	52
3.3.3.	Mô hình thực thể kết hợp	55
Chương 4.	KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	57
4.1.	Kết quả đã đạt được.....	57
4.2.	Hạn chế.....	57
4.2.1.	Về mặt khách hàng	57
4.2.2.	Về mặt nhân viên và admin.....	58
4.3.	Hướng phát triển	58
TÀI LIỆU THAM KHẢO		59

DANH MỤC TỪ VIẾT TẮT

STT	Ký hiệu chữ viết tắt	Chữ đầy đủ
1	MVC	Model-View-Controller
2	CSS	Cascading Style Sheets
3	JSP	JavaServer Pages
4	AOS	Animate on Scroll
5	SQL	Structured Query Language
6	AOP	Aspect Oriented Programming
7	IoC	Inversion of Control
8	SpEL	Spring Expression Language
9	Admin	Administrator
10	ORM	Object-Relational Mapping
11	API	Application Programming Interface
12	HQL	Hibernate Query Language
13	JSP	JavaServer Pages
14	JS	JavaScript
15	CDN	Content Delivery Network
16	HTML	Hypertext Markup Language

DANH MỤC HÌNH VẼ

Hình 2.1. Kiến trúc của Spring Framework	16
Hình 2.2. Hình tạo project	18
Hình 2.3. Hình cài đặt Apache Tomcat	19
Hình 2.4. Hình cài đặt server cho project.....	19
Hình 2.5. Hình cấu trúc thư mục theo projects.....	24
Hình 2.6. Hình cấu trúc thư mục theo files	24
Hình 2.7. Các luồng xử lý và mô hình Front Controller Design Pattern.....	25
Hình 2.8. Hình kiến trúc Hibernate	29
Hình 2.9. Sơ đồ chuyển động trạng thái của các persistent object.....	32
Hình 2.10. Vị trí của JSP trong một ứng dụng	34
Hình 3.1. Sơ đồ hoạt động chức năng đăng nhập.....	40
Hình 3.2. Sơ đồ hoạt động chức năng thêm tour	41
Hình 3.3. Sơ đồ hoạt động chức năng thanh toán.....	42
Hình 3.4. Sơ đồ hoạt động chức năng thống kê doanh thu.....	43
Hình 3.5. Sơ đồ hoạt động chức năng thêm người dùng	44
Hình 3.6. Hình giao diện đăng nhập.....	45
Hình 3.7. Hình giao diện đăng ký.....	46
Hình 3.8. Hình giao diện danh sách tour 1 (người dùng).....	46
Hình 3.9. Hình giao diện danh sách tour 2 (nhân viên và admin).....	47
Hình 3.10. Hình giao diện chi tiết tour	47
Hình 3.11. Hình giao diện thanh toán.....	48
Hình 3.12. Hình giao diện thêm tour	49
Hình 3.13. Hình giao diện sửa tour.....	49
Hình 3.14. Hình giao diện quản lý người dùng	50
Hình 3.15. Hình giao diện quản lý người dùng đặt tour.....	50
Hình 3.16. Hình giao diện thống kê	51
Hình 3.17. Hình giao diện thêm người dùng	52
Hình 3.18. Mô hình thực thể kết hợp	56

DANH MỤC BẢNG

Bảng 2.1. Bảng các thẻ trong JSP.....	34
Bảng 2.2. Các class cơ bản của Bootstrap.....	37
Bảng 3.1. Các bảng cơ sở dữ liệu.....	52
Bảng 3.2. Bảng cơ sở dữ liệu tour.....	53
Bảng 3.3. Bảng cơ sở dữ liệu người dùng.....	53
Bảng 3.4. Bảng cơ sở dữ liệu hóa đơn.....	54
Bảng 3.5. Bảng cơ sở dữ liệu chi tiết hóa đơn.....	54
Bảng 3.6. Bảng cơ sở dữ liệu bình luận.....	55

MỞ ĐẦU

Trong thời buổi hiện nay, nhu cầu đi lại đặc biệt là việc đi du lịch đang phát triển và phát triển rất mạnh, nó góp phần thúc đẩy việc sử dụng các ứng dụng, website để tìm kiếm và đặt tour theo nhu cầu của mình nhưng không cần đến các địa điểm, cửa hàng đại lý để đăng ký đặt tour. Việc này thúc đẩy nhu cầu thiết kế website, ứng dụng tăng cao. Việc tìm kiếm và đặt tour trên các ứng dụng, website sẽ dễ dàng hơn bao giờ hết nó giúp cho người dùng không phải lãng phí thời gian, tiết kiệm chi phí. Những ứng dụng, website này không chỉ hỗ trợ việc đặt và tìm kiếm tour của người dùng nó còn giúp đỡ cho các công ty, tổ chức quản lý dễ dàng, giảm thiểu các chi phí về nhân lực, ngoài ra nó còn tiếp cận tới nhiều loại khách hàng, ở mọi lứa tuổi có nhu cầu sử dụng. [1]

Nhờ vào những lợi ích mà các ứng dụng, website này mang lại nên nhiều công ty, tổ chức đã sở hữu riêng cho mình một vài ứng dụng, website. Để hiểu lý do tại sao những ứng dụng, website du lịch lại được nhiều người sử dụng như hiện nay. Chúng được thiết kế với giao diện đẹp mắt, thân thiện, gây được ấn tượng với người dùng sử dụng. Ngoài ra nó còn có các chức năng như đặt, tìm kiếm tour. Quản lý các tour khách hàng đặt, quản lý tài khoản người dùng, thống kê. Dựa vào những chức năng nổi bật và lợi ích nó mang lại, việc sở hữu một website du lịch chắc hẳn rất cần thiết. Để đáp ứng nhu cầu của người dùng các ứng dụng, các framework được ra đời và phát triển, nó giúp lập trình viên dễ dàng tiếp xúc và phát triển hơn. Các công cụ mạnh mẽ như Spring MVC, bootstrap và nhiều framework khác. Đang được rất nhiều lập trình viên sử dụng để xây dựng ứng dụng chức tỏ rằng các công cụ này là một trong những nền tảng đang rất phát triển hiện nay.

Chương 1. GIỚI THIỆU ĐỀ TÀI

Chương này nhằm giới thiệu về sức ảnh hưởng của du lịch khi được áp dụng công nghệ thông tin, lý do chọn đề tài này, mục tiêu và phạm vi đối tượng nghiên cứu của đề tài này.

1.1. Giới thiệu đề tài và lý do chọn đề tài

Du lịch là một ngành mang lại kinh tế hàng đầu, cũng vì thế du lịch phát triển khá nhanh cùng tốc độ cao. Bên cạnh là ngành kinh tế được xem là mũi nhọn của nhiều quốc gia và cả các tỉnh của nước ta, du lịch còn được xem là một cách để gìn giữ hòa bình, hợp tác lâu dài giữa các quốc gia, các nước với nhau. Những hạn chế của du lịch thông thường gặp phải như việc hạn chế đi lại của giữa các nước do ảnh hưởng của dịch, chi phí cho người lao động, truyền thông và quảng bá khiến giá thành cao hơn. Nhằm đưa ngành du lịch của nước ta phát triển và cho bạn bè các nước biết đến nhiều hơn về văn hóa, con người, cảnh đẹp và các kì quan của nước ta em đã nghiên cứu và thực hiện xây dựng một website đặt tour du lịch. [2]

Website đặt tour du lịch là một nơi để đăng ký, đặt tour và trải nghiệm những chuyến đi một cách dễ dàng và vô cùng thân thiện với người dùng, nó còn là một nơi để người dùng tìm kiếm chính xác, nhanh gọn về một địa điểm đẹp, lạ có thể nhiều người chưa từng biết đến trong nước. Website đặt tour du lịch được em lên ý tưởng và thực hiện bằng Java – là một ngôn ngữ hướng đối tượng phổ biến hiện nay. Để phục vụ cho việc hoàn thiện website em sử dụng Spring MVC, cùng với các ngôn ngữ JavaScript, CSS cùng các thư viện, công nghệ như JSP, AOS JS, Bootstrap giúp sinh động và đẹp mắt hơn trong việc sử dụng website, trong việc thiết kế chúng giúp cho việc thiết kế và lập trình dễ dàng hơn, tiết kiệm công sức, thời gian và vô cùng quen thuộc.

1.2. Mục tiêu của đề tài

Mục tiêu của đề tài này em muốn giúp cải thiện kỹ năng lập trình của bản thân, học hỏi thêm những kinh nghiệm về lập trình và thiết kế một website với một framework vô cùng phổ biến của Java đó chính là Spring MVC. Nhằm áp dụng những kiến thức, kinh nghiệm được học trên trường và tìm hiểu trên mạng vào đề tài này để có một website đầy đủ những tính năng cơ bản của một website du lịch cần có ví dụ

như tìm kiếm, thêm, sửa, xóa. Bên cạnh những chức năng em còn tăng sức sáng tạo của bản thân về mặt thiết kế giao diện của website một cách phù hợp và đẹp mắt với người sử dụng.

1.3. Phạm vi và đối tượng nghiên cứu

Website được xây dựng dựa trên các công nghệ phổ biến, tiện dụng hiện nay như Spring MVC, Bootstrap, Mysql. Phạm vi của đề tài này bao gồm người sử dụng hệ website: là những người đăng ký tour du lịch, người tìm kiếm thông tin về du lịch và những tổ chức, công ty quản lý website.

Đề tài này được xây dựng cho 3 đối tượng người dùng:

- Admin: Vai trò này có quyền xem, tìm kiếm, thêm, sửa, xóa nhân viên (quản lý nhân viên), tour du lịch (quản lý tour), quản lý các tour được khách hàng đặt và xem thống kê doanh thu theo tháng và năm.
- Nhân viên: Với vai trò này nhân viên có quyền thêm, sửa, tìm kiếm tour (quản lý tour), quản lý các tour được khách hàng đặt và quản lý tour.
- Khách hàng: Người dùng ở vai trò này có quyền xem, tìm kiếm tour, đặt tour trước và thanh toán những tour đã đặt.

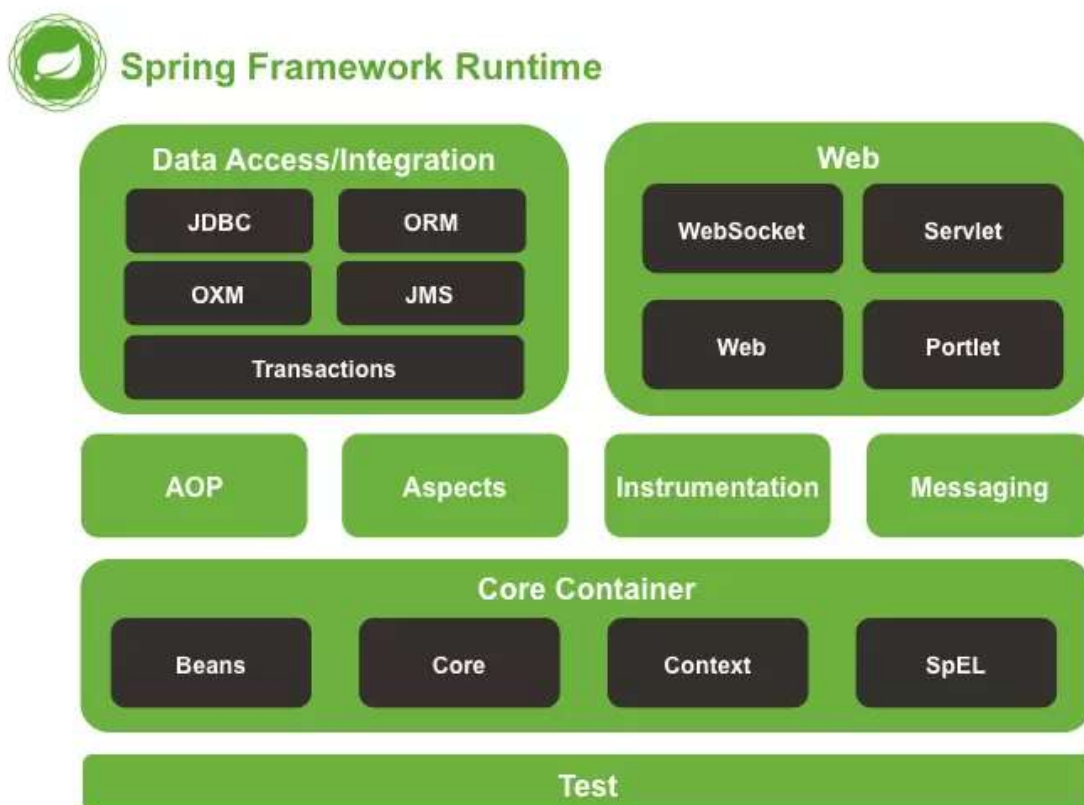
Chương 2. CƠ SỞ LÝ THUYẾT

Chương này nhằm giới thiệu về Spring framework, Spring MVC, Cloudinary, Hibernate, JSP, những điểm mạnh của bootstrap và tìm hiểu cách hoạt động của Front Controller Design Pattern.

2.1. Spring framework

2.1.1. Giới thiệu Spring framework

Spring là framework rất phổ biến hiện nay, nó được nhiều lập trình viên sử dụng để xây dựng và phát triển các ứng dụng Java, vì nó là mã nguồn mở, có nhiều ưu điểm mạnh mẽ như nhẹ, dễ kiểm thử, dễ bảo trì và có thể sử dụng lại code chính vì vậy cộng đồng sử dụng Spring framework rất đông và rất phát triển. [3]



Hình 2.1. Kiến trúc của Spring Framework

(Nguồn ảnh: <https://viettuts.vn/spring/kien-truc-spring>)

Bên trên là hình minh họa về kiến trúc của Spring Framework. Spring được chia thành nhiều module khác nhau bao gồm Data Access/Integration, Web, AOP, Aspects, Instrumentation, Core Container, Test. Mỗi module trong Spring sử dụng cho các mục

đích khác nhau. Spring tổ chức theo module vì vậy số lượng tệp và các lớp nhiều, nhưng khi ta đã quen thì cách tổ chức này rất tiện lợi và dễ sử dụng. [3]

2.1.1.1. Data Access/Integration

Những module của nhóm này cung cấp các xử lý liên quan tới cơ sở dữ liệu, tập trung vào những thứ cần thiết ta cần phải làm và giúp giảm bớt các mã nguồn bị trùng lặp trong quá trình tương tác với cơ sở dữ liệu. Nó còn hỗ trợ các công đoạn kể cả việc xử lý các ngoại lệ cho đến khi ta hết sử dụng đóng kết nối. [3]

2.1.1.2. Web

Đây là Spring MVC. Những module này sử dụng IoC để hỗ trợ tạo ứng dụng Web dễ dàng mở rộng và bảo trì. [3]

2.1.1.3. AOP, Aspects, Instrumentation và Messaging

Nhóm này gồm những module giúp tránh các lỗ hổng về bảo mật, chúng phục vụ cho việc xác định và phân quyền cho các ứng dụng của Spring. [3]

2.1.1.4. Core Container

Đây là nhóm gồm các module quan trọng nhất của Spring đó là:

- Beans: chúng được định nghĩa trong Spring và được quản lý bởi Spring Ioc. Đặc biệt có thuộc tính scope của đối tượng bean là phạm vi hoạt động của đối tượng bean mặc định là singleton chỉ tạo bean một lần duy nhất. [3]
- Core: IoC, DI là những đặc trưng quan trọng nhất của Spring. [3]
- Context: ApplicationContext container sẽ được nạp các định nghĩa và móc nối chúng với nhau. [3]
- SpEL: Hỗ trợ việc truy vấn và thao tác với các toán tử logic. [3]

2.1.1.5. Test

Nhóm này giúp cho việc tạo các unit test phục vụ việc kiểm tra hoạt động của các chức năng chúng ta viết. [3]

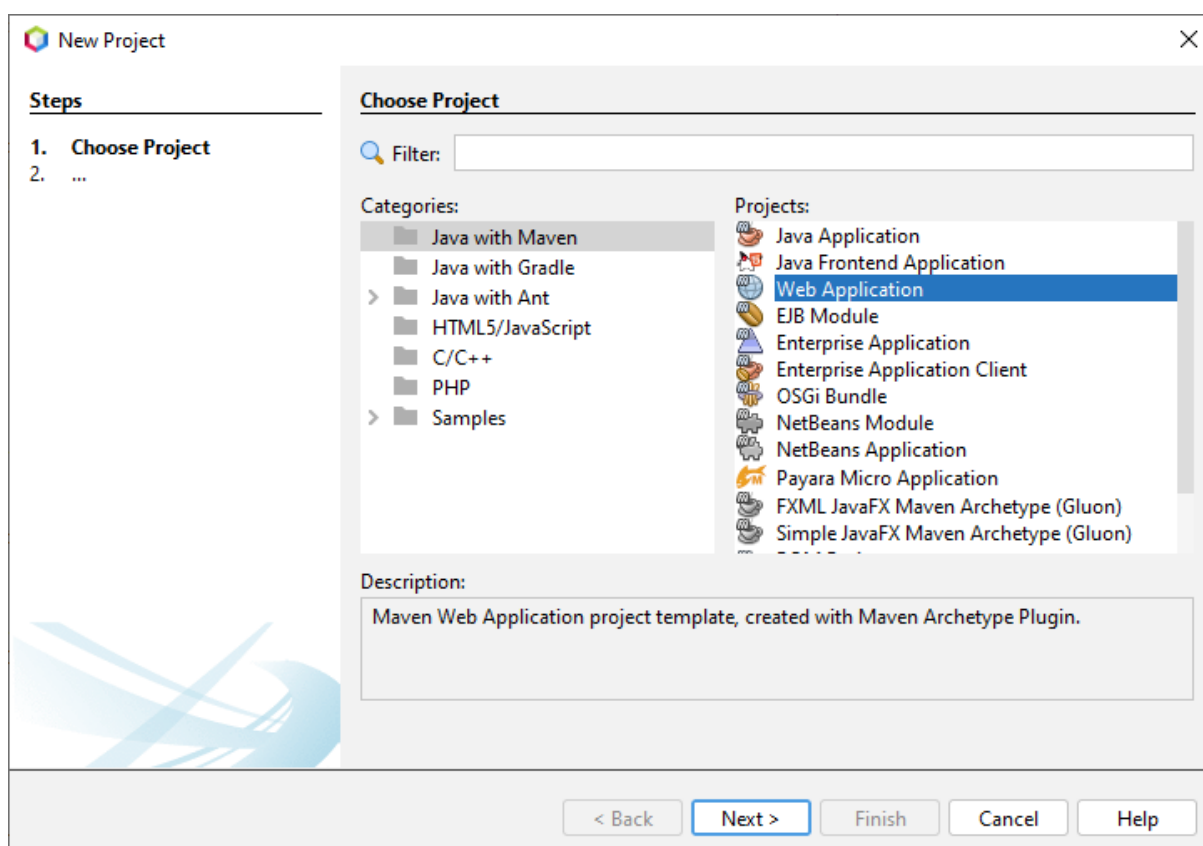
2.1.2. Giới thiệu Spring MVC

Spring MVC là một framework khá phổ biến và được rất nhiều lập trình viên sử dụng, vì là một framework được thực hiện theo mô hình MVC (Model-View-Controller) dùng để xây dựng các ứng dụng Web. [3]

2.1.3. Xây dựng Website với Spring MVC

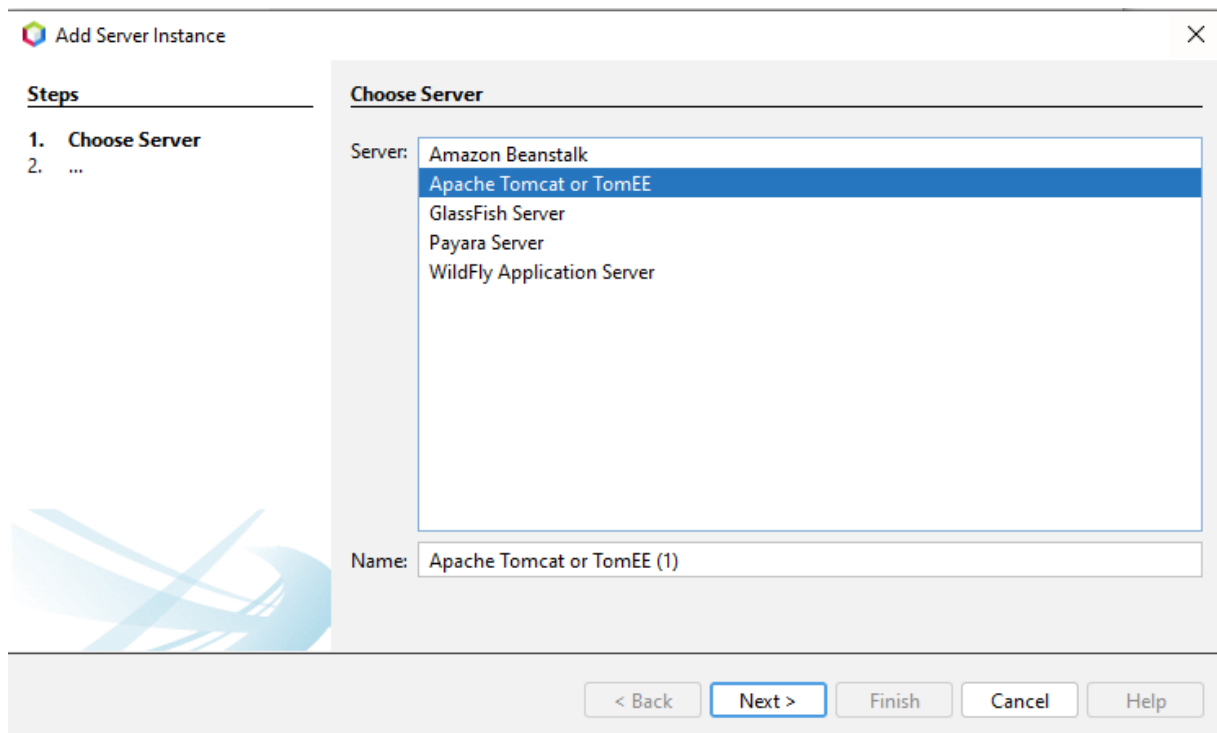
2.1.3.1. Tạo project

Đầu tiên ta chọn Java with Maven sau đó tìm đến Web Application để tạo project như hình sau.



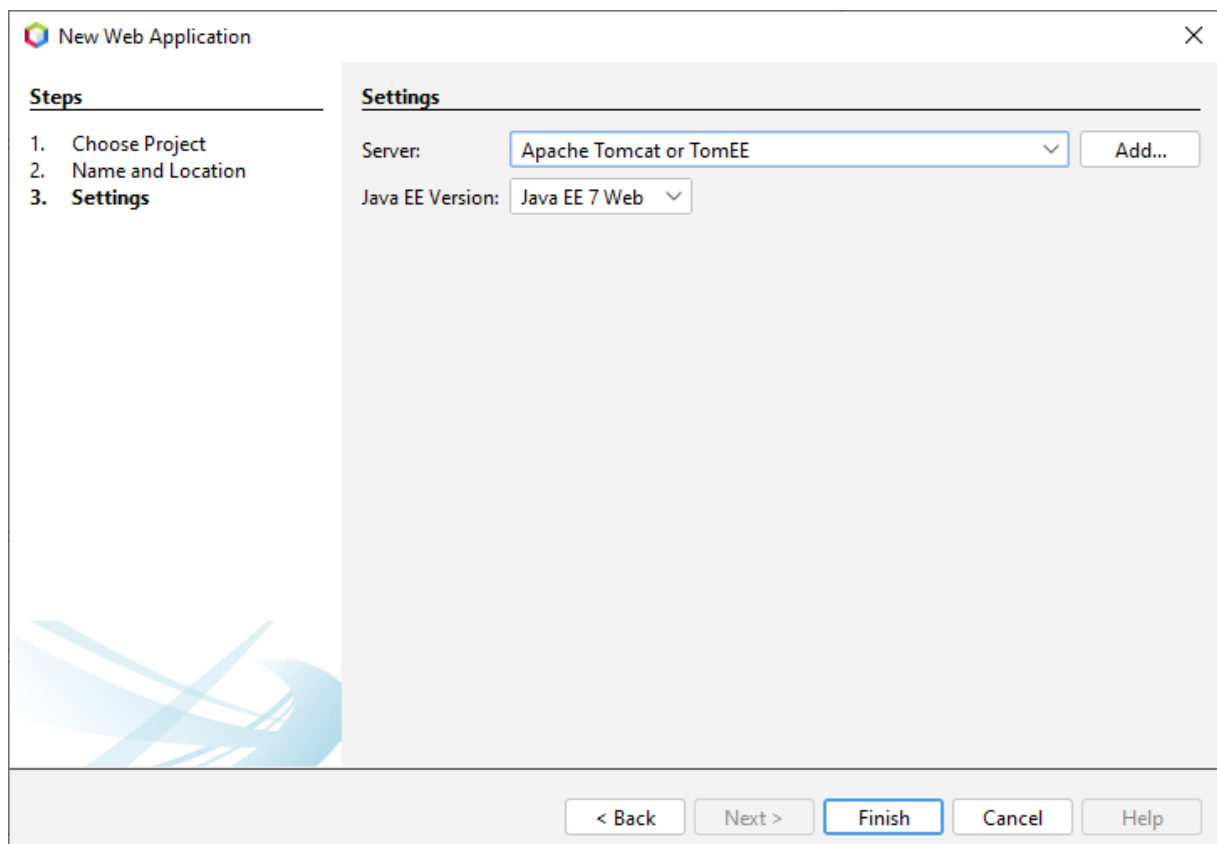
Hình 2.2. Hình tạo project

Ứng dụng này sử dụng Apache Tomcat để làm server, để sử dụng Tomcat ta tải tại <http://tomcat.apache.org/> sau đó thêm và chọn server với tên là Apache Tomcat or TomEE như hình sau sau đó nhập thông tin cần thiết để tạo.



Hình 2.3. Hình cài đặt Apache Tomcat

Tiếp theo ta chọn server để tiến hành hoàn thành việc tạo project.



Hình 2.4. Hình cài đặt server cho project

Thêm các dependency hỗ trợ việc cấu hình đầu tiên vào trong pom.xml.

```

<dependency>
    <groupId>javax</groupId>
    <artifactId>javaee-web-api</artifactId>
    <version>7.0</version>
    <scope>provided</scope>
</dependency>
<dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-webmvc</artifactId>
    <version>5.3.9</version>
</dependency>
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>jstl</artifactId>
    <version>1.2</version>
</dependency>

```

Tiếp theo ta tạo thư mục /WEB-INF/applicationContext.xml trong webapp

```

<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:context="http://www.springframework.org/schema/context"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
         http://www.springframework.org/schema/beans/spring-beans-3.2.18.xsd
         http://www.springframework.org/schema/context
         http://www.springframework.org/schema/context/spring-context-3.2.18.xsd">
    <context:annotation-config />
</beans>

```

Sau đó ta cấu hình cơ bản các gói trong configs

Trong lớp WebConfig.java

```
@Configuration
@EnableWebMvc
@ComponentScan(basePackages = {
    "com.dql.controllers"})
public class WebConfig implements WebMvcConfigurer{

    @Override
    public void configureDefaultServletHandling(
        DefaultServletHandlerConfigurer configurer) {
        configurer.enable();
    }

    @Bean
    public InternalResourceViewResolver
        getInternalResourceViewResolver() {
        InternalResourceViewResolver resource =
            new InternalResourceViewResolver();
        resource.setViewClass(JstlView.class);
        resource.setPrefix("/WEB-INF/jsp/");
        resource.setSuffix(".jsp");

        return resource;
    }
}
```

Trong lớp DispatcherServletInitializer

```
public class DispatcherServletInitializer extends
AbstractAnnotationConfigDispatcherServletInitializer{

    @Override

    protected Class<?>[] getRootConfigClasses() {

        return null;

    }

    @Override

    protected Class<?>[] getServletConfigClasses() {

        return new Class[] {

            WebConfig.class };

    }

    @Override

    protected String[] getServletMappings() {

        return new String [] {"/"};

    }

}
```

Trong thư mục trong webapp với tên WEB-INF/jsp, tại đây ta tạo tập tin jsp mới với tên hello.jsp cùng nội dung sau.

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <title>Hello World </title>
    </head>
    <body>
        <h1>${welcome}</h1>
    </body>
</html>
```

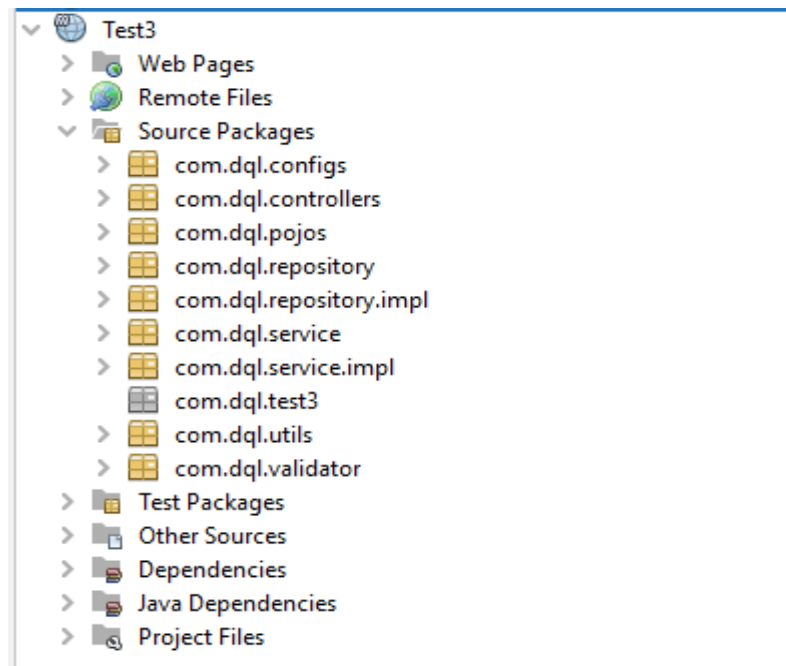

Tạo một lớp controller như sau là đã hoàn thành chương trình đầu tiên với Spring MVC.

```
@Controller
public class HomeController {@RequestMapping(value = "/")
    public String index(Model model) {
        model.addAttribute("welcome", "Hello World!!!");
        return "hello";
    }
}
```

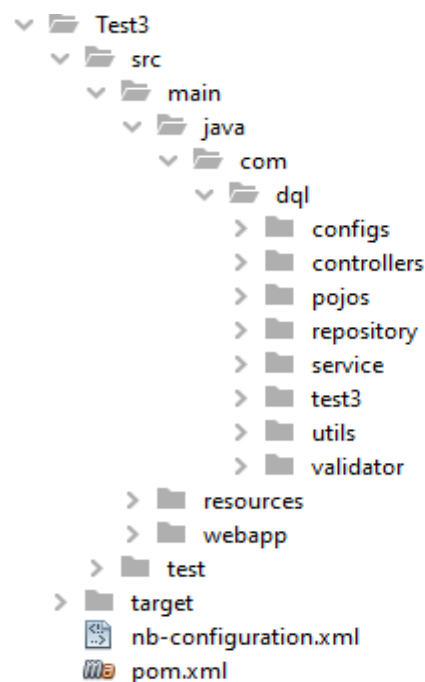
2.1.3.2. Cấu trúc của project

Gồm các gói cấu hình và các tầng xử lý từng nhiệm vụ khác nhau, các tầng này tương tác qua lại với nhau thực hiện các logic và lưu trữ .

- Gói configs: Các lớp trong gói này là các lớp cấu hình cho chương trình
- Gói controllers: Mỗi lớp controller trong gói này tương tác với view, lớp này sử dụng annotation @Controller hoặc @RestController.
- Gói repository: Các lớp trong gói này sử dụng @Repository, các lớp tương tác thẳng với các lớp trong gói pojo, dùng để thực hiện các truy vấn về dữ liệu.
- Gói service: Các lớp trong gói này là các lớp service sử dụng các annotation @Service, là lớp kết nối giữa các lớp controller và các lớp repository.
- Gói utils: Chứa các lớp sử dụng dùng để tính toán chung.
- Gói validator: Chứa các lớp có chức năng kiểm tra điều kiện của dữ liệu.
- Gói pojos: Các lớp trong gói này khi gắn @Entity và @Table tương ứng với các bảng trong cơ sở dữ liệu.



Hình 2.5. Hình cấu trúc thư mục theo projects

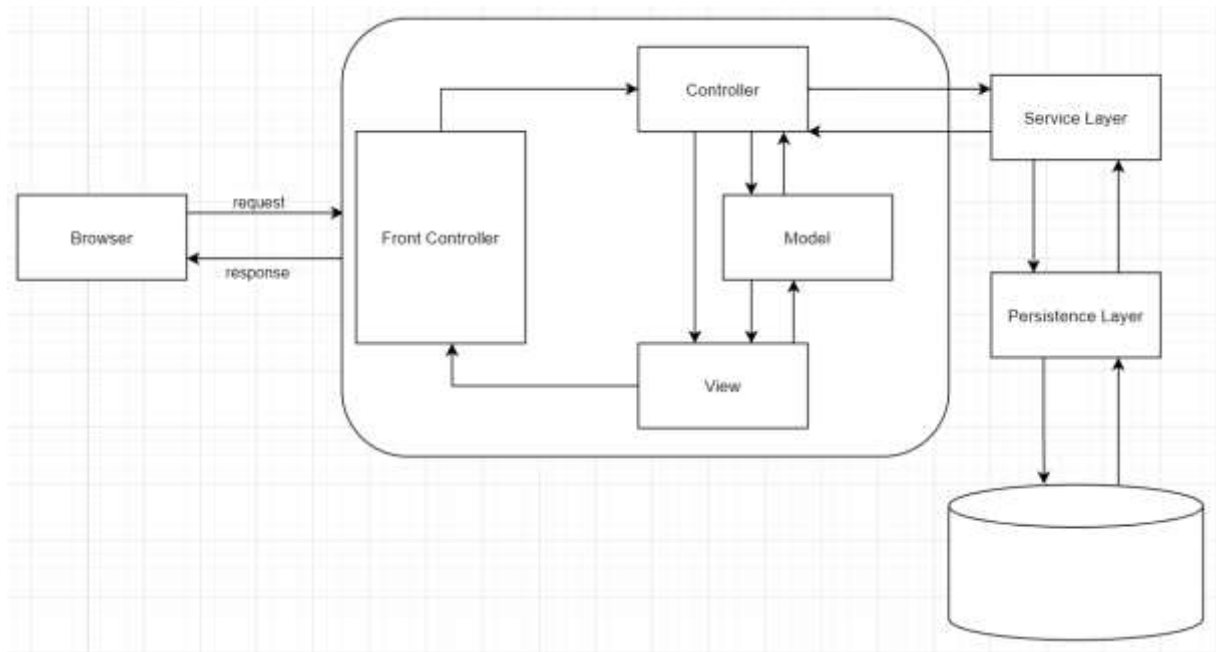


Hình 2.6. Hình cấu trúc thư mục theo files

2.2. Front Controller Design Pattern

Font Controller là nơi tiếp nhận các request, thao tác từ người dùng đồng thời chuyển các request, thao tác này đến các controller tương ứng. Các controller khi nhận các request, thao tác này sẽ thực hiện các phương thức đã viết sẵn và tương tác với tầng Service Layer thực hiện các logic và lưu trữ. Tiếp theo cập nhập dữ liệu cho

model và hiển thị lên View cho người dùng thấy. Dưới đây là các luồng xử lý và mô hình của Front Controller Design Pattern.



Hình 2.7. Các luồng xử lý và mô hình Front Controller Design Pattern

2.3. Cloudinary

Cloudinary là một trang web cung cấp cho phép người dùng quản lý video, hình ảnh bao gồm các chức năng như upload và lưu trữ như là một kho chứa. Với sự tiện lợi như không cần cài một phần mềm nào cả, mọi thao tác upload được xử lý tự động khi ta sử dụng. Bên cạnh đó Cloudinary giúp ta dễ dàng quản lý kho dữ liệu.

2.3.1. Cài đặt Cloudinary

Ta đăng ký và đăng nhập tài khoản trên trang web của Cloudinary. Với url sau <https://cloudinary.com/users/login>. Sau đó ta sử dụng các thông tin của tài khoản đã có sẵn Cloud name, API Key và API Secret để cấu hình và kết nối.

Trước tiên ta thêm dependency của cloudinary vào pom.xml

```
<dependency>
    <groupId>com.cloudinary</groupId>
    <artifactId>cloudinary-http44</artifactId>
    <version>1.29.0</version>
</dependency>
```

2.3.1. Sử dụng Cloudinary

Sau khi thêm dependency ta cấu hình và tạo bean để kết nối với kho chứa của tài khoản trên website.

```
@Bean
public Cloudinary cloudinary() {
    Cloudinary c = new Cloudinary(ObjectUtils.asMap(
        "cloud_name", "xxxxs1mplexxx",
        "api_key", "929613542259299",
        "api_secret", "4Uo-2dgKzAMvs0BR6zLvJJuSvJA",
        "secure", true));
    return c;
}
```

Xử lý để upload ảnh

```
@Autowired
private Cloudinary cloudinary;

public boolean themSuaNguoiDung(NguoiDung nguoiDung) {
    try{Map m = this.cloudinary.uploader().upload(
        nguoiDung.getFile().getBytes(),
        ObjectUtils.asMap("resource_type", "auto"));
        nguoiDung.setAnh((String)m.get("secure_url"));
        return this.nguoiDungRepository
            .themSuaNguoiDung(nguoiDung);
    } catch (IOException ex) {
        System.err.println("Đã xảy ra lỗi!!!");
    }
    return false;
}
```

Thuộc tính nhận dữ liệu upload trong lớp NguoiDung

```

@Entity
@Table(name = "nguoidung")
public class NguoiDung{
    @Transient
    private MultipartFile file;
}

```

Xử lý ở form nhận dữ liệu upload

```

<form:form method="POST"
    modelAttribute="nguoiDung"
    enctype="multipart/form-data">
    <div class="form-group">
        <label for="file">Ảnh đại diện</label>
        <form:input type="file" id="file" path="file"/>
    </div>
    <input type="submit" value="Đăng ký"/>
</form:form>

```

Ta cần cài thêm các dependency sau để hỗ trợ việc upload nhiều dữ

```

<dependency>
    <groupId>commons-fileupload</groupId>
    <artifactId>commons-fileupload</artifactId>
    <version>1.4</version>
</dependency>
<dependency>
    <groupId>commons-io</groupId>
    <artifactId>commons-io</artifactId>
    <version>2.6</version>
</dependency>

```

Các dependency trên hỗ trợ việc cho phép phía client gửi các tập tin và dữ liệu đến server đó là các Multipart request, lớp CommonsMultipartResolver cho phép một request được phép chứa nhiều phần. Dưới đây là lớp CommonsMultipartResolver.

```
@Bean
public CommonsMultipartResolver multipartResolver() {
    CommonsMultipartResolver resolver =
        new CommonsMultipartResolver();
    resolver.setDefaultEncoding("UTF-8");
    return resolver;
}
```

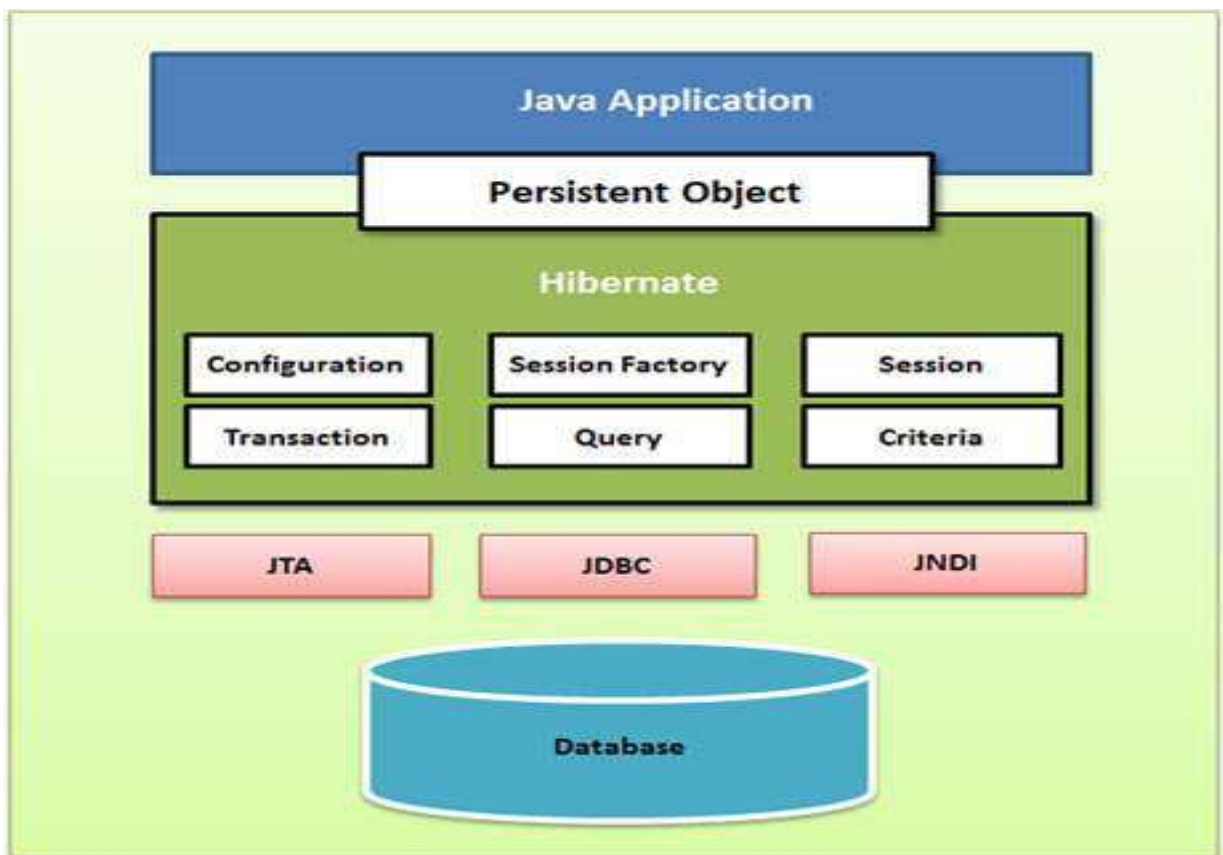
Xử lý các form nhận dữ liệu ở file jsp như sau

```
<form:form method="POST"
    modelAttribute="nguoiDung"
    enctype="multipart/form-data">
    ...
    <div class="form-group">
        <label for="tenTour">Tên tour</label>
        <form:input type="text" id=" tenTour "
            path=" tenTour "/>
    </div>
    <div class="form-group">
        <label for="file">Hình ảnh</label>
        <form:input type="file" id="file" path="file"/>
    </div>
    ...
    <input type="submit" value="Thêm tour"/>
</form:form>
```

2.4. Kiến trúc Hibernate

Hibernate là một framework mã nguồn mở, cung cấp các giải pháp về ORM [4] hỗ trợ cho việc map các objects của Java với cơ sở dữ liệu. Các ORM này giúp cho lập trình viên thay vì quan tâm tới các bảng trong cơ sở dữ liệu, giờ đây lập trình viên chỉ cần tập trung tương tác với các đối tượng, xử lý logic hướng đối tượng và không cần phải viết các truy vấn SQL như trước nữa. [5]

Kiến trúc Hibernate được xây dựng dựa trên 4 tầng bao gồm Java Application, Hibernate, backend api và database. Hình bên dưới là kiến trúc Hibernate.



Hình 2.8. Hình kiến trúc Hibernate

(Nguồn ảnh: <https://viettuts.vn/hibernate/kien-truc-hibernate>)

2.4.1. Các đối tượng trong kiến trúc Hibernate

2.4.1.1. Configuration

Đối tượng này được chúng ta tạo đầu tiên, nó đại diện cho một tập tin cấu hình hoặc thuộc tính để hibernate sử dụng và nó chỉ tạo một lần duy nhất trong lúc khởi động ứng dụng. Đây là cấu hình để tạo kết nối với database. [5]

Trước tiên ta thêm dependency sau

```
<dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-core</artifactId>
    <version>5.5.6.Final</version>
</dependency>
```

Tạo bean và cung cấp thông tin để kết nối

```
@Autowired
private Environment env;

@Bean
public DataSource dataSource() {
    DriverManagerDataSource dataSource =
        new DriverManagerDataSource();
    dataSource.setDriverClassName(env.getProperty
        ("hibernate.connection.driverClass"));
    dataSource.setUrl(env.getProperty
        ("hibernate.connection.url"));
    dataSource.setUsername(env.getProperty
        ("hibernate.connection.username"));
    dataSource.setPassword(env.getProperty
        ("hibernate.connection.password"));

    return dataSource;
}
```

Thông tin kết nối

```
hibernate.connection.driverClass=com.mysql.cj.jdbc.Driver
hibernate.connection.url=jdbc:mysql://localhost:3306/dulic
hldb
hibernate.connection.username=root
hibernate.connection.password=123456
```


2.4.1.2. SessionFactory

Đối tượng này dùng để khởi tạo ra các đối tượng Session, mỗi SessionFactory ứng với mỗi cơ sở dữ liệu, vì vậy muốn sử dụng nhiều cơ sở dữ liệu ta cần tạo số SessionFactory tương ứng. [5]

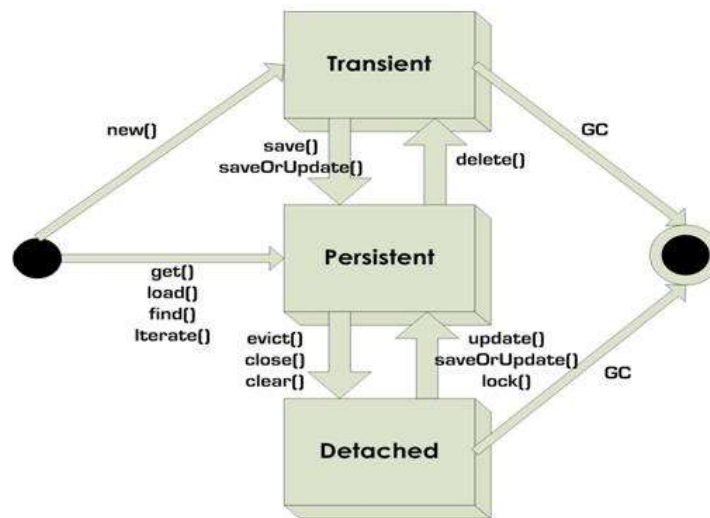
2.4.1.3. Session

Đối tượng này sử dụng cho việc thêm, sửa, xóa và các tương tác với cơ sở dữ liệu bằng kết nối vật lý thông qua các lớp đối tượng của Java được ánh xạ(persistent object). Các đối tượng ánh xạ(persistent object) sẽ tồn tại một trong ba trạng thái sau: transient (vừa được tạo ra), persistent (đã kết hợp với một session), detached(khi session được đóng). [5] [6]

```
public long slBinhLuan(int tourId) {  
    Session session = this.sessionFactory  
                        .getObject().getCurrentSession();  
    Query query = session.createQuery  
                    ("Select Count(*)  
                     From BinhLuan Where tour.id=:id");  
    query.setParameter("id", tourId);  
  
    return Long.parseLong(query  
                          .getSingleResult().toString());  
}
```

Đây là sơ đồ chuyển động trạng thái của các đối tượng ánh xạ(persistent object)

Persistence Lifecycle



Hình 2.9. Sơ đồ chuyển động trạng thái của các persistent object

(Nguồn ảnh: <https://lastdemon.wordpress.com/cac-trang-thai-cua-doi-tuong-trong-hibernate/>)

2.4.1.4. Các đối tượng khác

- Transaction: đối tượng này làm việc với hibernate bằng các giao tác.
- Query: các query sử dụng các truy vấn SQL hoặc HQL để tạo các đối tượng và lấy dữ liệu từ cơ sở dữ liệu.
- Criteria: các criteria sử dụng để tạo và truy vấn dữ liệu có điều kiện.

Dưới đây là truy vấn dữ liệu của HQL

```
public long slNguoiDung() {
    Session session = this.sessionFactory
        .getObject().getCurrentSession();
    Query q = session.createQuery
        ("Select Count(*) From NguoiDung");
    return Long.parseLong
        (q.getSingleResult().toString());
}
```

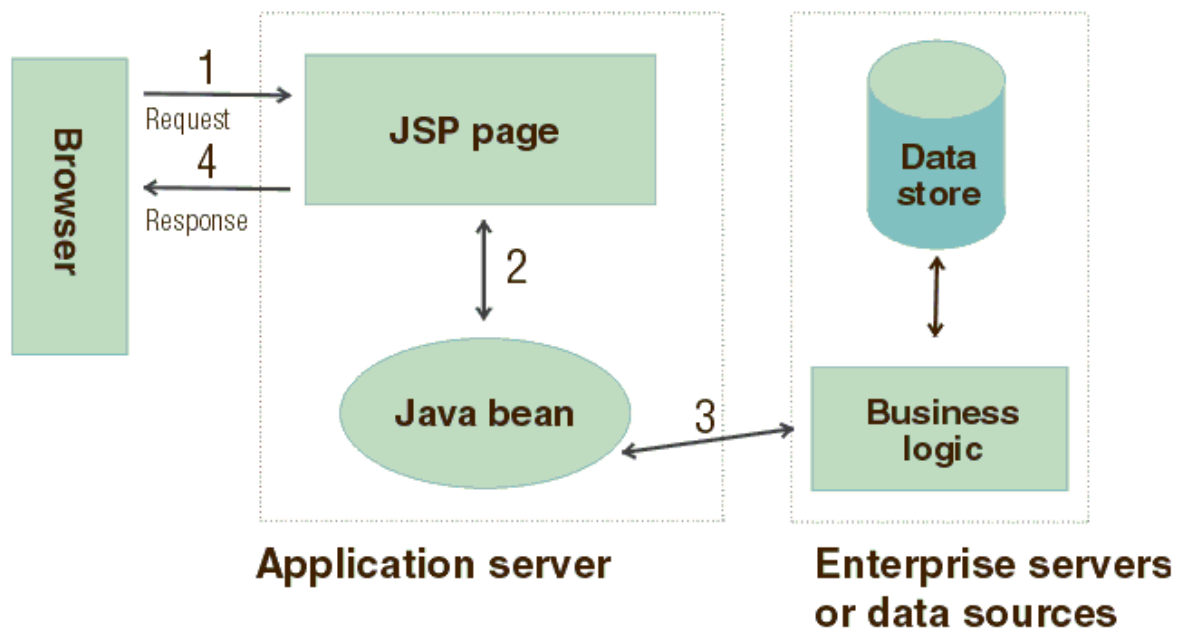
Dưới là truy dữ liệu của Criteria

```
public List<NguoiDung> dsNguoiDung(String taiKhoan) {
    Session session = this.sessionFactory
        .getObject().getCurrentSession();
    CriteriaBuilder builder = session
        .getCriteriaBuilder();
    CriteriaQuery<NguoiDung> query = builder
        .createQuery(NguoiDung.class);
    Root root = query.from(NguoiDung.class);
    query = query.select(root);

    if(taiKhoan != null){
        Predicate p = builder.like(root
            .get("taiKhoan").as(String.class),
            String.format("%%%s%", taiKhoan));
        query = query.where(p);
    }
    Query q = session.createQuery(query);
    return q.getResultList();
}
```

2.5. JSP

JSP là một công nghệ dùng để xây dựng các ứng dụng web. Nó giúp cho lập trình viên có thể viết code java trong HTML bằng việc sử dụng các thẻ của JSP. Nó giúp cho việc lập trình dễ dàng và thuận tiện hơn. [7]



Hình 2.10. Vị trí của JSP trong một ứng dụng

(Nguồn ảnh: <https://viettuts.vn/jsp>)

Ta cần thêm các thẻ định dạng sau để sử dụng

```

<%@ taglib prefix="c"
    uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="sec"
    uri="http://www.springframework.org/security/tags"%>
<%@taglib prefix="form"
    uri="http://www.springframework.org/tags/form" %>
  
```

Các thẻ được sử dụng nhiều

Bảng 2.1. Bảng các thẻ trong JSP

c:forEach	Lặp qua các phần tử trong thẻ.
c:if	Câu lệnh so sánh.
c:url	Tạo một đường dẫn với các tham số truy vấn.
sec:authorize	Kiểm tra quyền.
form:form	Tạo thẻ form.
form:input	Ô nhập dữ liệu.

form:select	Danh sách lựa chọn.
form:option	Bên trong select, giá trị tùy chỉnh trong danh sách

2.6. Bootstrap

2.6.1. Giới thiệu Bootstrap

Bootstrap là một framework mã nguồn mở và miễn phí. Hiện tại đang được rất nhiều lập trình viên sử dụng để xây dựng website, giúp cho quá trình xây dựng website trở nên dễ dàng và nhanh chóng hơn dựa vào các thành phần được thiết kế sẵn như navbar, tables, modal, dropdown, grids, Với các thuộc tính được thiết kế sẵn như màu sắc, kích thước, chiều cao, chiều rộng, ... làm giảm thời gian thiết kế website nhưng vẫn sáng tạo và đẹp mắt không bị rập khuôn theo một mẫu thiết kế nào có sẵn. Tuy rằng còn nhiều hạn chế như chưa thật sự hoàn thiện, tốc độ tối ưu chưa được đánh giá cao, nhưng nhờ vào khả năng thích ứng với các trình duyệt, dễ dàng sử dụng, dễ dàng điều chỉnh giữa các thiết bị khác nhau và tiện lợi nó vì vậy Bootstrap được rất đông người sử dụng. [8]

2.6.2. Cài đặt và sử dụng Bootstrap

2.6.2.1. Cài đặt

Có 2 cách để cài đặt và sử dụng Bootstrap phổ biến:

- Tải Bootstrap về máy: Ở cách này ta phải tải Bootstrap về tại trang <https://getbootstrap.com>, sau khi tải về và giải nén thành công ta có hai thư mục css và js. Sau đó ta copy là hai thư mục trên vào thư mục tương ứng với loại file. Trong thẻ head của file jsp ta sử dụng Bootstrap tạo thẻ link chỉ đến file bootstrap.css và trong thẻ body tạo thẻ script chỉ đến file bootstrap.js. [8]

```
<link rel="stylesheet" type="text/css"
      href="<c:url value="/css/bootstrap.css"/>" />

<script type="text/javascript"
      src="<c:url value="/js/bootstrap.js"/>"></script>
```

- Sử dụng CDN: Ở cách này đơn giản hơn và không cần phải tải bất cứ file nào hết. Ta chỉ cần thêm các thẻ sau vào phần head của file jsp ta sử dụng Bootstrap sẽ hoàn thành việc cài đặt Bootstrap. [8]

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/
bootstrap@4.6.0/dist/css/bootstrap.min.css"/>

<script src="https://cdn.jsdelivr.net/npm/
bootstrap@4.6.0/dist/js/bootstrap.min.js"></script>
```

2.6.2.2. Sử dụng Bootstrap

Vì các thuộc tính của Bootstrap được thiết sẵn nên chúng được sử dụng bằng cách gọi tên các lớp có sẵn của Bootstrap. Dưới đây là một đoạn code cơ bản để ta hiểu sơ qua cách sử dụng Bootstrap.

```
<div class="container">
  <div class="row">
    <div class="col-sm-4">
      <h3>Cột 1</h3>
    </div>
    <div class="col-sm-8">
      <h3>Cột 2</h3>
    </div>
  </div>
</div>
```

Grid System hiển thị dưới dạng lưới, bao gồm 12 cột(col) trong một class row nó hỗ trợ việc tương thích với nhiều giao diện. Grid System gồm có 5 loại class: [8]

- .col-: Kích thước thiết bị cực nhỏ, chiều rộng của thiết bị < 576px.
- .col-sm-: Kích thước thiết bị nhỏ, chiều rộng của thiết bị ≥ 576 px.
- .col-md-: Kích thước thiết bị trung bình, chiều rộng của thiết bị ≥ 768 px.
- .col-lg-: Kích thước thiết bị lớn, chiều rộng của thiết bị ≥ 992 px.
- .col-xl-: Kích thước thiết bị cực lớn, chiều rộng của thiết bị ≥ 1200 px.

Các class cơ bản được sử dụng trong Bootstrap

Bảng 2.2. Các class cơ bản của Bootstrap

Tên class	Mô tả	Ví dụ
containers	Thùng chứa	container, container-fluid
buttons	Các nút	btn, btn-primary, btn-dark, ...
text-alignment	Căn chỉnh văn bản	text-left, text-center, ...
text-transform	Chuyển đổi kiểu văn bản	text-uppercase
margin	Khoảng cách giữa viền và phần tử tiếp xúc	mt-2, mb-3, mr-1, ml-5 (có giá trị auto và từ 0 tới 5)
padding	Khoảng cách giữa nội dung và viền	mr-3, ml-2, mt-4, mb-1 (có giá trị auto và từ 0 tới 5)

Chương 3. **HỆ THỐNG ĐẶT TOUR DU LỊCH**

Chương này nhằm giới thiệu chi tiết về hệ thống đặt tour du lịch bao gồm các phần về phân tích các chức năng của hệ thống, cách hoạt động của chúng bên cạnh đó còn trình bày các giao diện của hệ thống và cách thiết kế cơ sở dữ liệu.

3.1. Phân tích hệ thống

3.1.1. Mô tả chức năng

Hệ thống đặt tour du lịch là một website được xây dựng với mục đích hạn chế thời gian lãng phí, thuận tiện trong việc quản lý và nâng cấp, giảm bớt chi phí và hiệu quả hơn so với việc đặt tour truyền thống. Hệ thống được xây dựng từ các chức năng cơ bản kết hợp với nhau. Dưới đây là giới thiệu và mô tả sơ về các chức năng trong hệ thống.

3.1.1.1. Chức năng chung

Chức năng chung là phần bao gồm các chức năng dành cho tất cả người sử dụng hệ thống thao tác được. Có một số chức năng như bình luận, thanh toán thì bắt buộc người dùng phải đăng nhập tài khoản mới có thể thao tác được.

- Chức năng đăng ký: Chức năng này là nơi để người dùng tạo nhập thông tin cá nhân của mình ứng với một tài khoản có vai trò trong hệ thống và vai trò mặc định khi đăng ký là khách hàng(USER).
- Chức năng đăng nhập: Chức năng này dùng để hệ thống xác thực và phân quyền cho người dùng như khách hàng(USER), nhân viên(MANAGER), admin(ADMIN).
- Chức năng tìm kiếm tour: Chức năng này giúp người dùng tìm kiếm tên của một tour bất kỳ, khi đó hệ thống sẽ lọc ra danh sách phù hợp với tên tour người dùng tìm kiếm.
- Chức năng bình luận: Chức năng này chỉ cho phép người dùng đã có tài khoản trong hệ thống được phép để lại bình luận của mình trong chi tiết tour.
- Chức năng thanh toán: Chức năng này chỉ cho phép người dùng đã đăng nhập vào hệ thống và lưu hóa đơn các tour người dùng đã đặt.

3.1.1.2. Chức năng dành cho nhân viên và admin

Đây là những chức năng dành riêng cho những người dùng có vai trò nhân viên hoặc admin trong hệ thống. Vì là các chức năng quản lý và tác động đến dữ liệu riêng của hệ thống nên những chức năng này sẽ hiển thị khi nhân viên hoặc admin đăng nhập thành công.

- Chức năng thêm, sửa tour: Chức năng này có nhiệm vụ giúp các nhân viên, admin thêm hoặc chỉnh sửa dữ liệu của tour trong hệ thống.
- Chức năng quản lý người dùng: Chức năng này giúp cho việc xem, tìm kiếm, chỉnh sửa, xóa thông tin của người dùng trong hệ thống.
- Chức năng quản lý đặt tour: Chức năng này bao gồm các danh sách hóa đơn của tất cả người dùng đã đặt. Nó hỗ trợ cho việc kiểm tra tính đúng đắn và kiểm soát được số lượng người đã đặt tour.

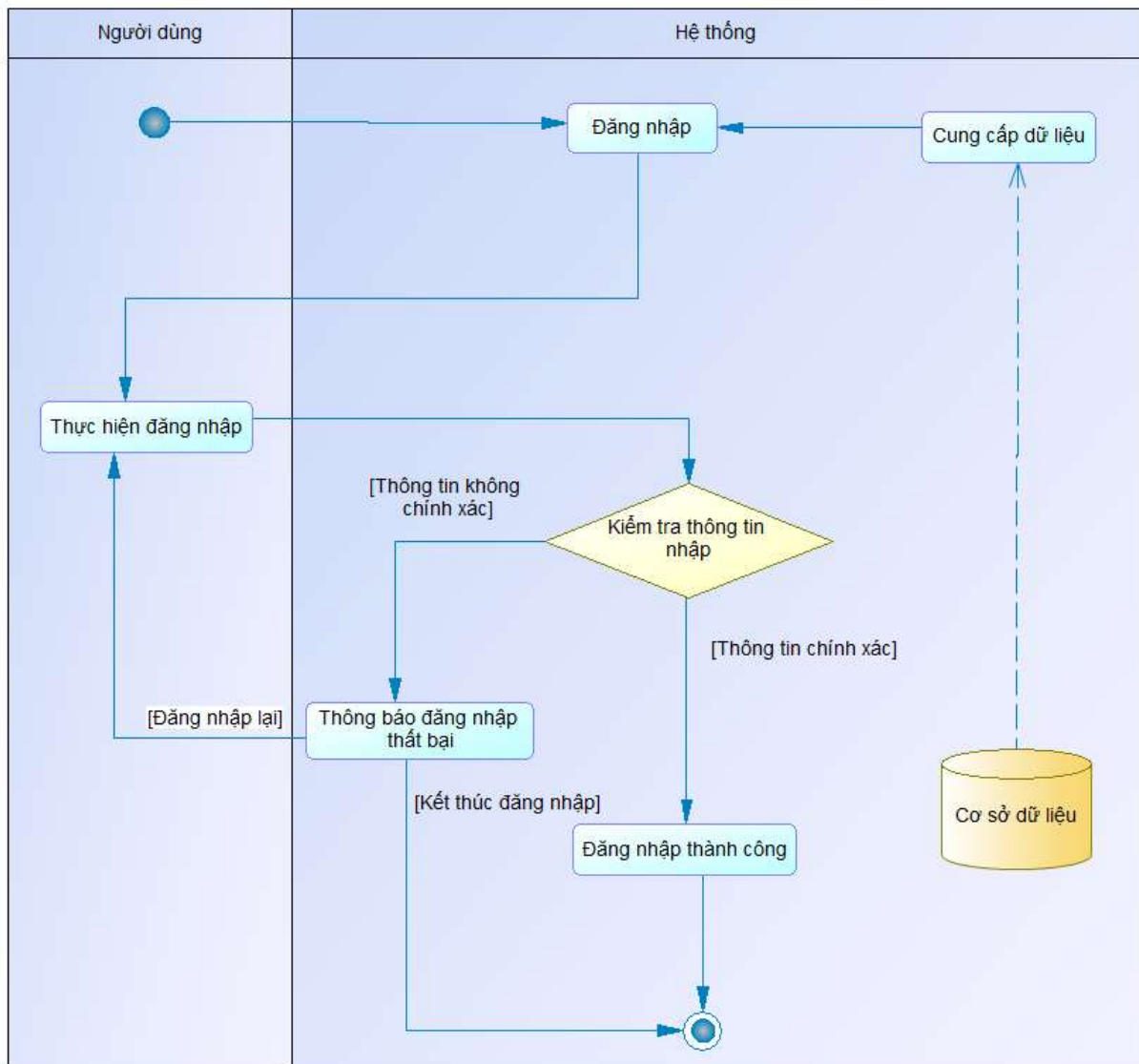
3.1.1.3. Chức năng dành cho admin

Ngoài những chức năng cơ bản trên dưới đây là những chức năng chỉ dành cho admin và tránh việc lạm dụng quyền bừa bãi.

- Chức năng thống kê: Nhiệm vụ của chức năng này là thống kê doanh thu của từng tour và hiển thị theo dạng biểu đồ đường, nó giúp cho việc nhận xét và so sánh dễ dàng và hiệu quả hơn.
- Chức năng xóa tour, xóa người dùng bất kỳ: Chức năng này chỉ cho phép admin mới xóa được.
- Chức năng thêm người dùng: chỉ có admin có quyền thêm một tài khoản với vai trò tự chọn nhằm tránh việc lạm dụng cấp quyền không phù hợp.

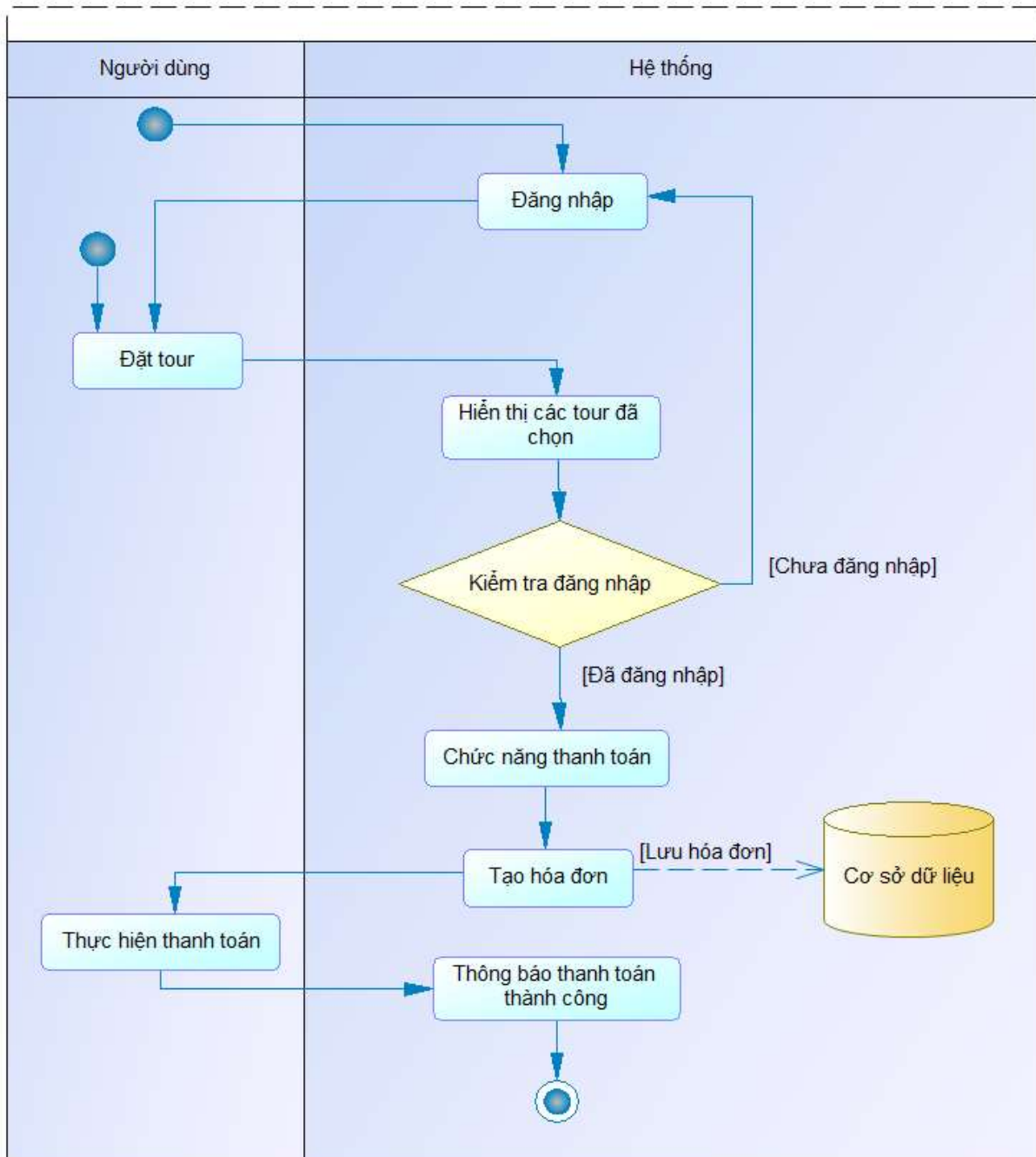
3.1.2. Sơ đồ hoạt động

3.1.2.1. Chức năng đăng nhập



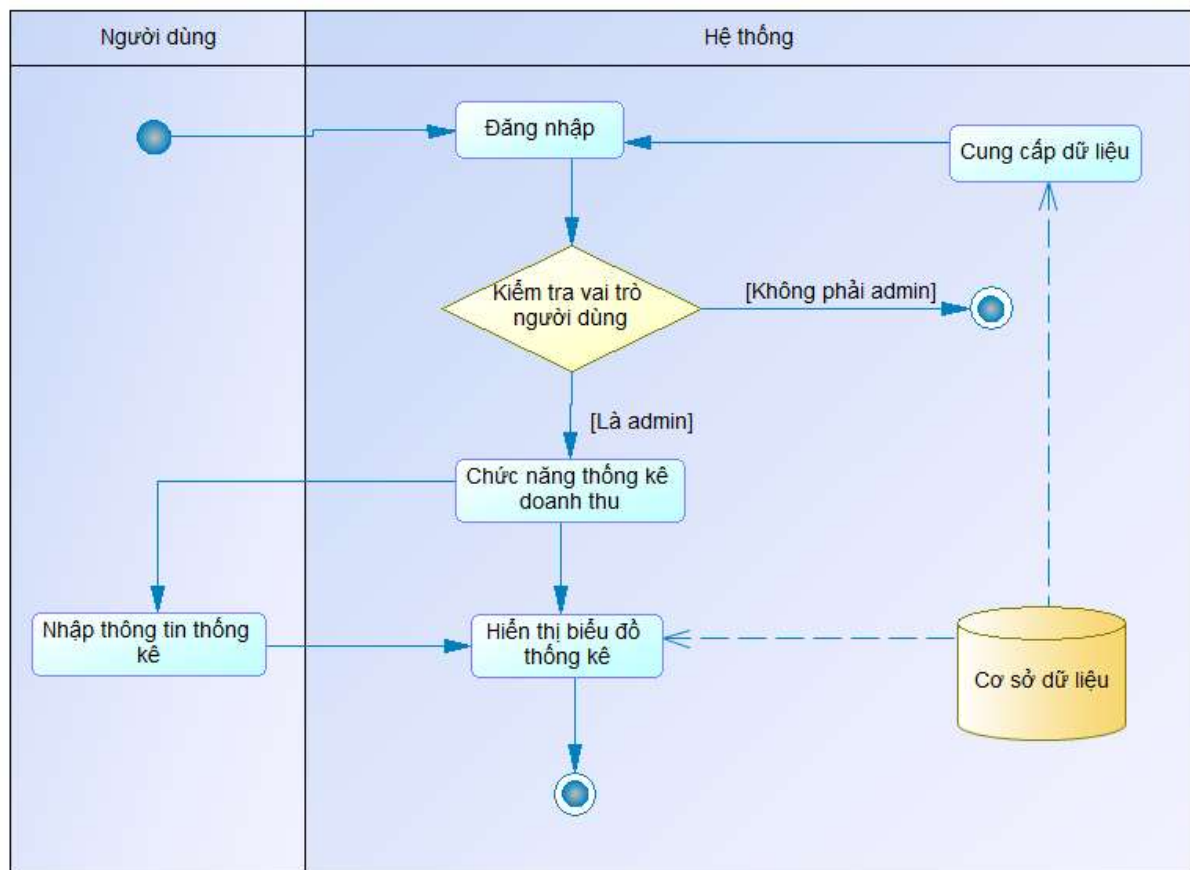
Hình 3.1. Sơ đồ hoạt động chức năng đăng nhập

3.1.2.3. Chức năng thanh toán



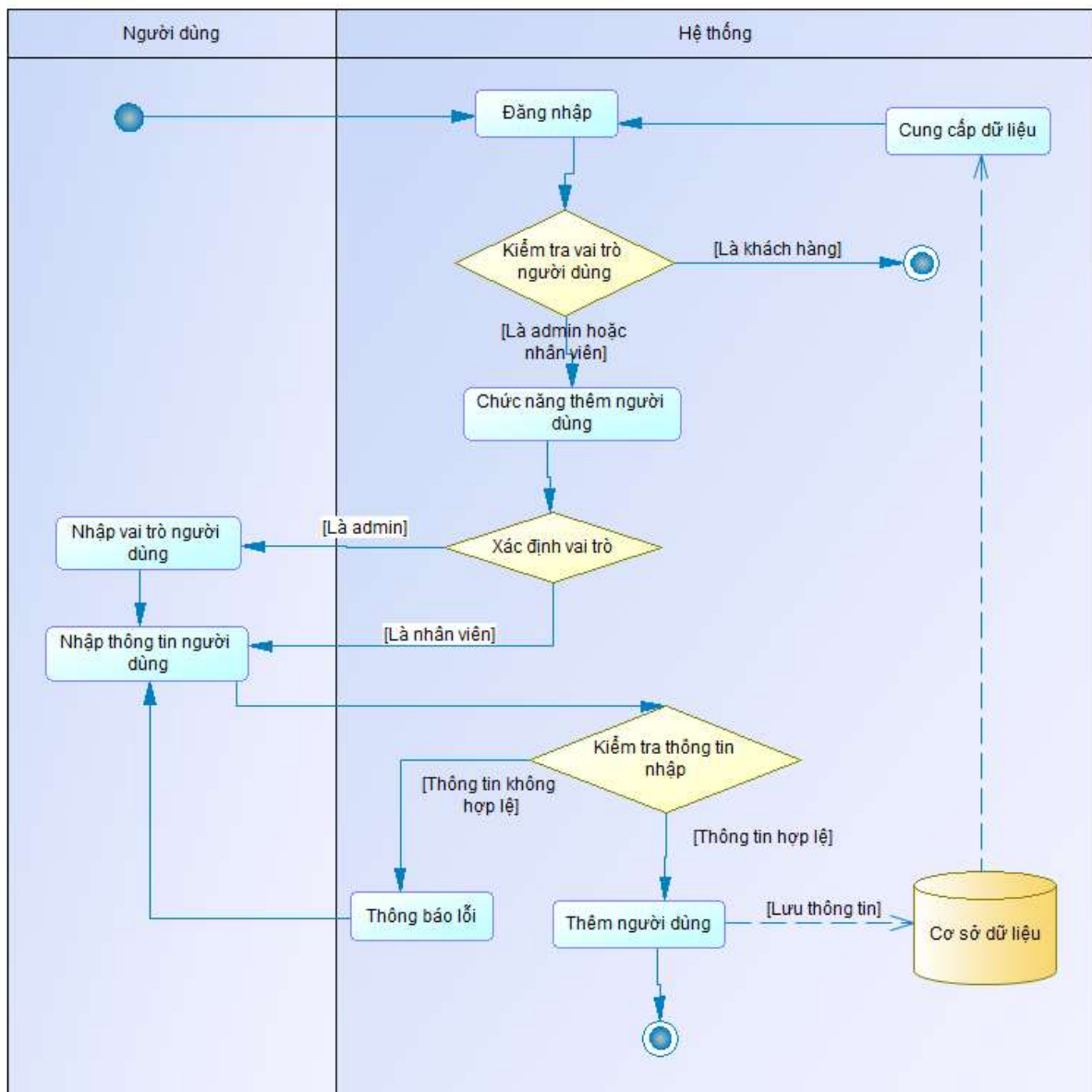
Hình 3.3. Sơ đồ hoạt động chức năng thanh toán

3.1.2.4. Chức năng thống kê doanh thu



Hình 3.4. Sơ đồ hoạt động chức năng thống kê doanh thu

3.1.2.5. Chức năng thêm người dùng



Hình 3.5. Sơ đồ hoạt động chức năng thêm người dùng

3.2. Giao diện hệ thống

Giao diện của hệ thống là bộ mặt của hệ thống, nó là tập hợp các bố cục hình ảnh và chữ viết giúp cho người dùng cảm thấy đẹp mắt và quan trọng nó cũng dễ dàng sử dụng các chức năng của hệ thống.

3.2.1. Giao diện chung

Phần này là giao diện dành cho toàn bộ người dùng khi sử dụng hệ thống đặt tour du lịch. Có một số giao diện mà người dùng không cần phải đăng nhập nhưng vẫn có thể sử dụng được.

3.2.1.1. Đăng nhập

Đây là giao diện của chức năng đăng nhập, chức năng này có nhiệm vụ cho người dùng đăng nhập với tài khoản của mình đã có, giúp hệ thống biết được thông tin người đăng nhập là ai và có vai trò gì trong hệ thống. Phục vụ cho tất cả các đối tượng từ người dùng, nhân viên và cả admin.



Hình 3.6. Hình giao diện đăng nhập

3.2.1.2. Đăng ký

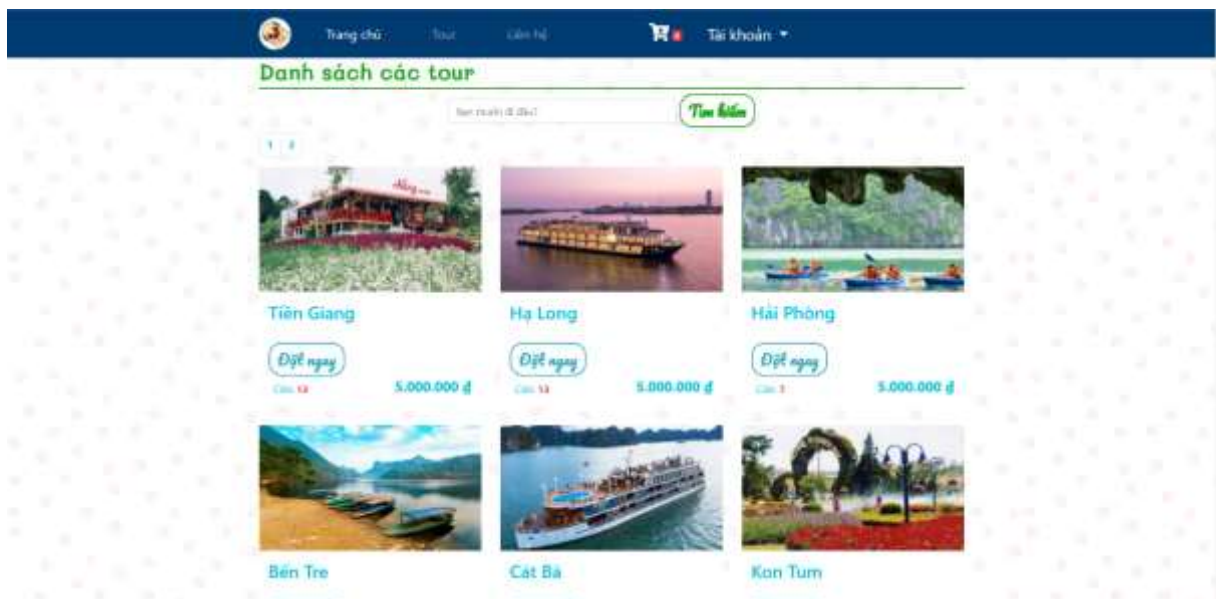
Đây là giao diện đăng ký tài khoản, chức năng này cho phép người dùng đăng ký thông tin của mình trên hệ thống, giúp người dùng có thể sử dụng các chức năng khác của hệ thống, chỉ khi người dùng có vai trò trong hệ thống mới sử dụng được.



Hình 3.7. Hình giao diện đăng ký

3.2.1.3. Danh sách các tour

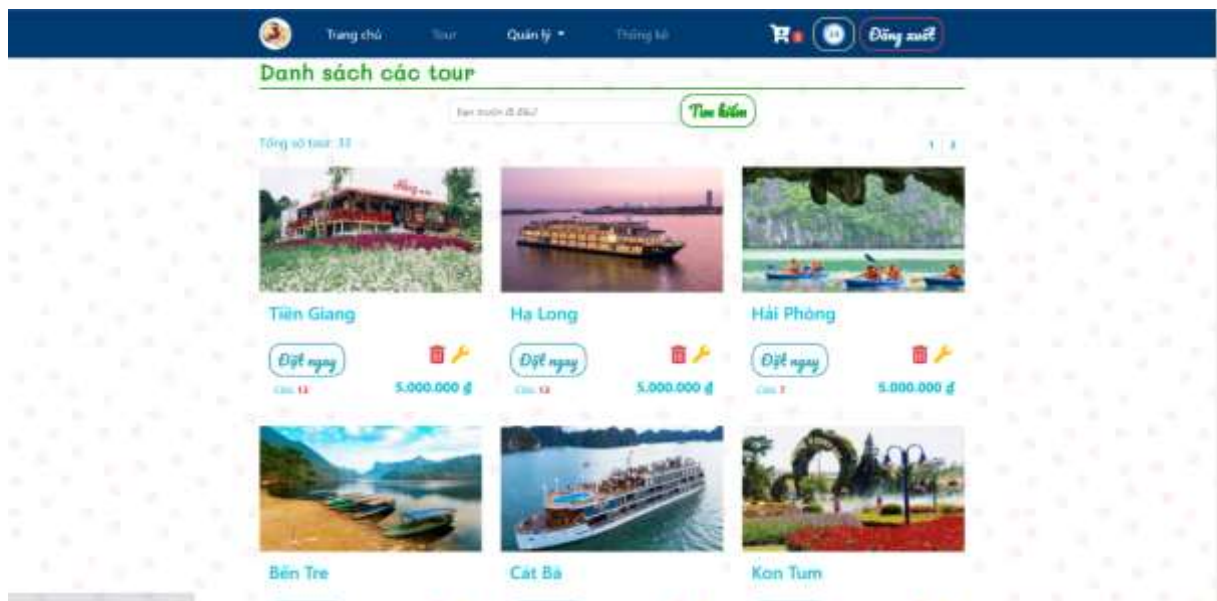
Danh sách tour là một giao diện giúp người dùng tìm kiếm các tour có trên hệ thống, bên cạnh đó đây là nơi người dùng đặt tour du lịch. Còn khi ta click vào một tour bất kì ta có thể chuyển sang giao diện chi tiết tour đó.



Hình 3.8. Hình giao diện danh sách tour 1 (người dùng)

Ở hình 3.9 bên dưới là giao diện danh sách tour của nhân viên, admin nó có các thêm các chức năng dành cho nhân viên hoặc admin, biểu tượng chiếc cờ lê màu vàng là chức năng sửa tour ở chức năng này thì cả nhân viên và admin đều có quyền chỉnh

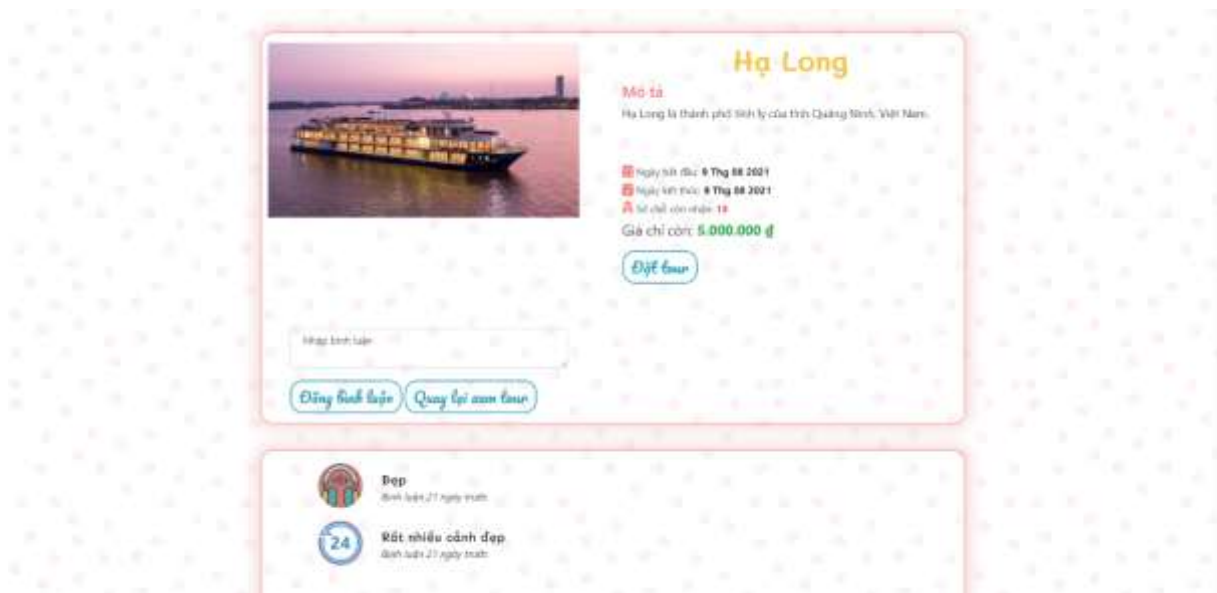
sửa, bên cạnh chiếc cờ lê là biểu tượng thùng rác đây là chức năng xóa tour, chức năng này chỉ có người dùng với vai trò là admin thì mới có thể xóa được.



Hình 3.9. Hình giao diện danh sách tour 2 (nhân viên và admin)

3.2.1.4. Chi tiết tour

Giao diện chi tiết này không những cung cấp cho người dùng biết được các thông tin của tour đó và đặt tour, nó còn là nơi để người dùng xem và để lại những bình luận, nhận xét cho tour du này. Nó giúp người dùng có trải nghiệm tốt hơn khi sử dụng.



Hình 3.10. Hình giao diện chi tiết tour

3.2.1.5. Thanh toán

Giao diện này cho phép người dùng xem lại tất cả các tour mà mình đã chọn trước đó, bên cạnh việc tăng giảm số lượng một tour bất kì người dùng có thể xóa bỏ các lựa chọn không phù hợp.

Mã tour	Tên tour	Số lượng đặt/Số lượng còn	Đơn giá	
49	Đà Nẵng	<input type="text" value="1"/> /7	899.000 đ	Xóa
3	Hải Phòng	<input type="text" value="1"/> /7	1.299.000 đ	Xóa
7	Bến Tre	<input type="text" value="3"/> /21	1.200.000 đ	Xóa
56	Cao Bằng	<input type="text" value="1"/> /10	1.230.000 đ	Xóa
57	Đà Nẵng 2	<input type="text" value="1"/> /19	2.399.000 đ	Xóa
12	Cát Bà	<input type="text" value="1"/> /13	1.990.000 đ	Xóa

Tổng tiền: 11.417.000 đ

Đặt ngay

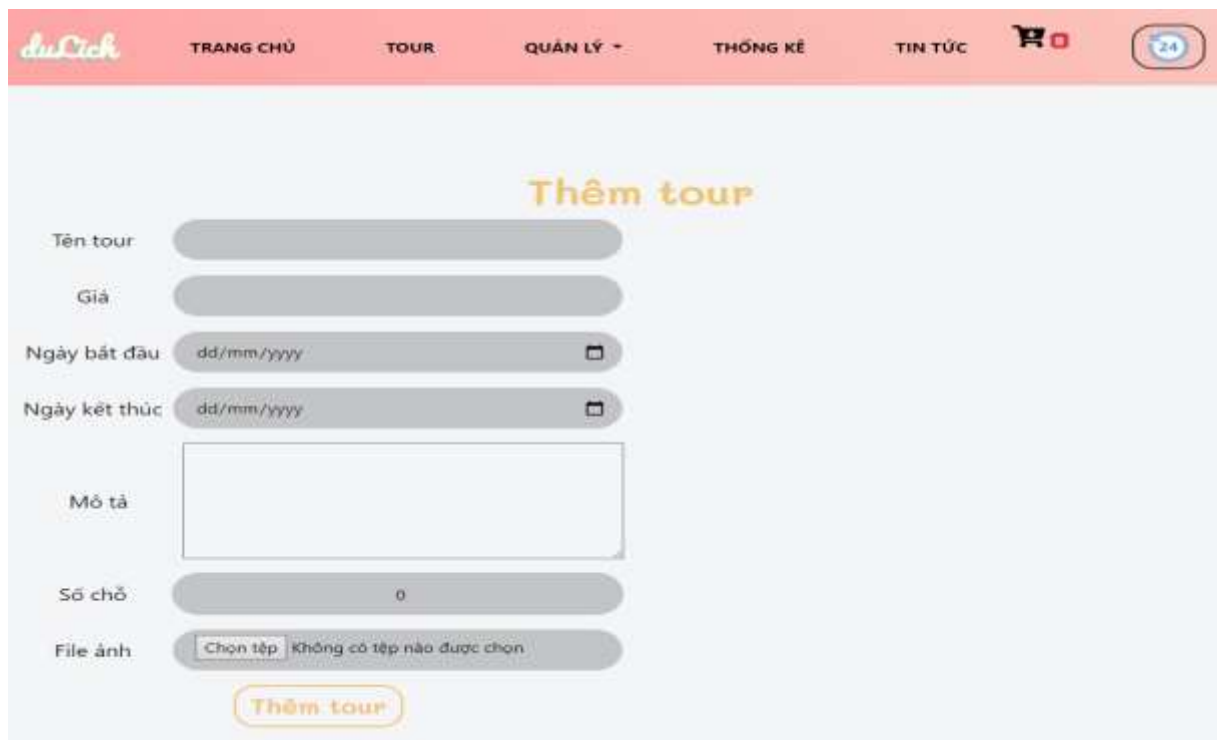
Hình 3.11. Hình giao diện thanh toán

3.2.2. Giao diện chung của admin và nhân viên

Người dùng có vai trò admin hay nhân viên có thể thao tác các chức năng thêm, sửa, quản lý người dùng, ... trong hệ thống. Những chức năng này giúp cho việc quản lý, chỉnh sửa trở nên dễ dàng hơn.

3.2.2.1. Thêm, sửa tour

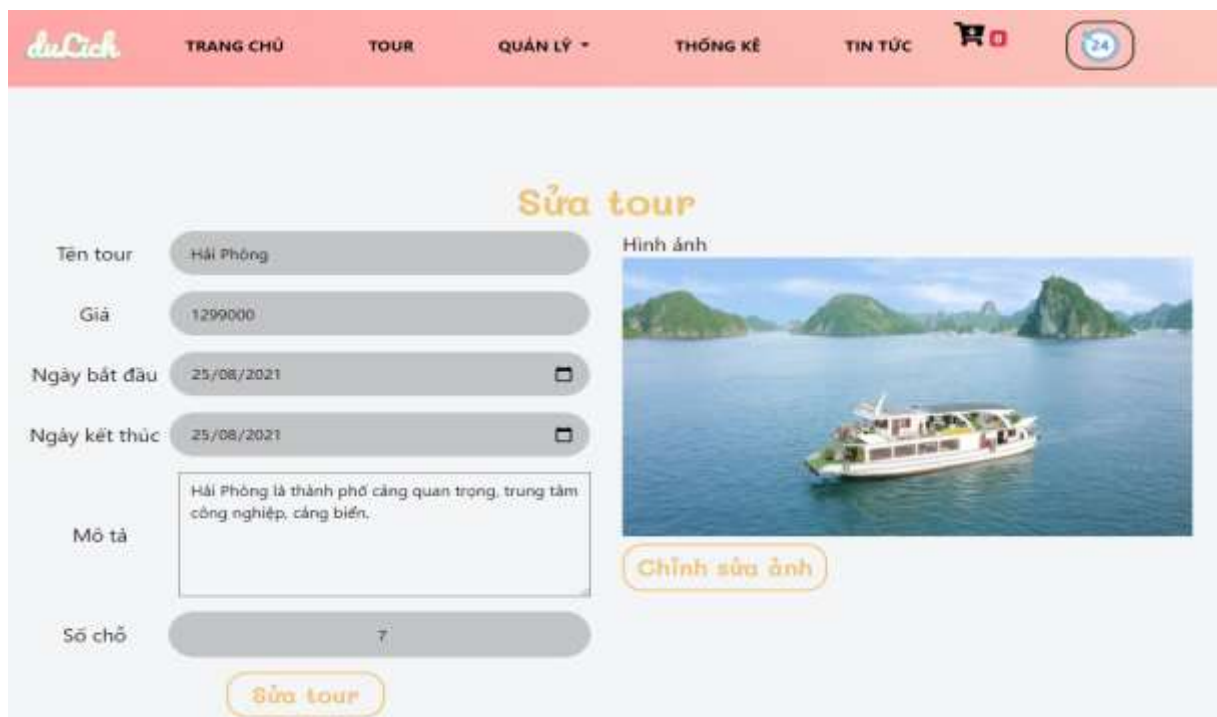
Hình 3.12 bên dưới là giao diện giúp nhân viên hoặc admin thêm mới thông tin của một tour du lịch.



The screenshot shows the 'Thêm tour' (Add tour) form in the duLich system. The form is titled 'Thêm tour' in orange text. It contains several input fields: 'Tên tour' (Tour name), 'Giá' (Price), 'Ngày bắt đầu' (Start date) with a date picker icon, 'Ngày kết thúc' (End date) with a date picker icon, 'Mô tả' (Description) with a text area, 'Số chỗ' (Number of seats) with a numeric input field showing '0', and 'File ảnh' (Image file) with a 'Chọn tệp' (Choose file) button and the text 'Không có tệp nào được chọn' (No files selected). At the bottom, there is an orange button labeled 'Thêm tour'.

Hình 3.12. Hình giao diện thêm tour

Hình 3.13 là giao diện có chức năng sửa thông tin của một tour du lịch.



The screenshot shows the 'Sửa tour' (Edit tour) form in the duLich system. The form is titled 'Sửa tour' in orange text. It contains several input fields: 'Tên tour' (Tour name) with the value 'Hải Phòng', 'Giá' (Price) with the value '1299000', 'Ngày bắt đầu' (Start date) with the value '25/08/2021' and a date picker icon, 'Ngày kết thúc' (End date) with the value '25/08/2021' and a date picker icon, 'Mô tả' (Description) with the text 'Hải Phòng là thành phố cảng quan trọng, trung tâm công nghiệp, cảng biển.', and 'Số chỗ' (Number of seats) with the value '7'. To the right of the form, there is a section titled 'Hình ảnh' (Image) with a placeholder image of a boat on the water. Below the image is an orange button labeled 'Chỉnh sửa ảnh' (Edit image). At the bottom, there is an orange button labeled 'Sửa tour'.

Hình 3.13. Hình giao diện sửa tour

3.2.2.2. Quản lý người dùng

Giao diện này là nơi cho phép quản lý tất cả tài khoản của người dùng, nó là một bảng bao gồm các thông tin cơ bản của người dùng, bên cạnh đó còn có những chức năng như sửa đổi thông tin của một tài khoản, xóa một tài khoản, thêm một người dùng và tìm kiếm một tài khoản bất kì trong hệ thống.



Hình 3.14. Hình giao diện quản lý người dùng

3.2.2.3. Quản lý người dùng đặt tour

Giao diện này là một bảng danh sách người dùng đăng ký đặt tour. Nó cho phép nhân viên hay admin xem chi tiết các tour được đăng ký đặt.



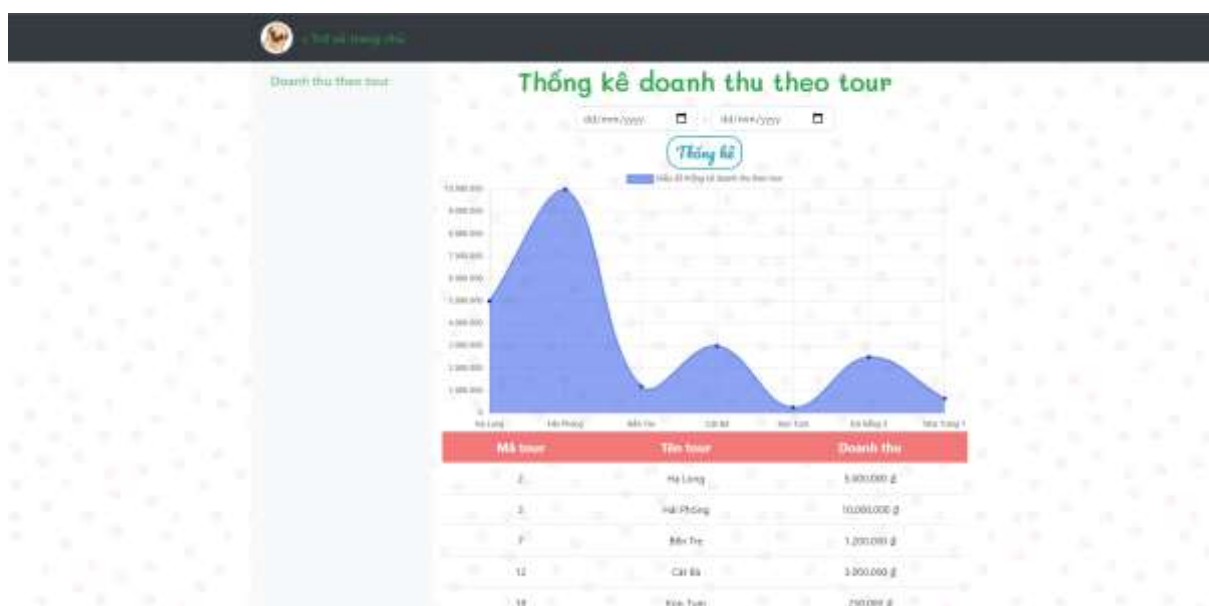
Hình 3.15. Hình giao diện quản lý người dùng đặt tour

3.2.3. Giao diện admin

Giao diện những chức năng đặc biệt chỉ dành cho người dùng có vai trò admin trong hệ thống.

3.2.3.1. Thống kê

Đây là nơi người dùng có vai trò admin dùng để thống kê doanh thu của các tour theo tháng, quý hay năm. Với biểu đồ dòng đẹp mắt, nó làm cho việc so sánh doanh thu giữa các tour dễ dàng và vô cùng đẹp mắt.



Hình 3.16. Hình giao diện thống kê

3.2.3.2. Thêm người dùng

Giao diện này chỉ có người dùng có vai trò admin mới có quyền thêm một người dùng mới với một vai trò được tùy chọn. Nó giúp cho việc thêm mới một nhân viên hay admin linh hoạt, nhanh chóng hơn.



Hình 3.17. Hình giao diện thêm người dùng

3.3. Thiết kế cơ sở dữ liệu

3.3.1. Các bảng

Bảng 3.1. Các bảng cơ sở dữ liệu

Tên bảng	Mô tả
tour	Lưu toàn bộ thông tin của tour du lịch
người dùng	Lưu thông tin của tất cả tài khoản đã đăng ký, dùng để đăng nhập và phân quyền cho người dùng của hệ thống
hóa đơn	Lưu trữ các thông tin người dùng đã đặt tour
chi tiết hóa đơn	Lưu trữ các thông tin chi tiết của hóa đơn người dùng đã đặt tour
biên luận	Lưu trữ tất cả các biên luận người dùng đã để lại trong các tour

3.3.2. Chi tiết các bảng dữ liệu

3.3.2.1. Bảng tour

Bảng này có một khóa chính là id (Mã tour). Khi nhân viên hoặc admin thêm hoặc sửa dữ liệu một tour vào hệ thống, thông tin của tour đó sẽ được lưu trong bảng

tour. Danh sách tour cũng truy xuất dữ liệu từ bảng này để hiển thị cho người dùng thao tác vào các chức năng khá như tìm kiếm, xem, sửa, xóa.

Bảng 3.2. Bảng cơ sở dữ liệu tour

Tên thuộc tính	Kiểu dữ liệu	Mô tả	Thông tin thêm
id	INT	Mã tour	Tự tăng(Khóa chính)
tenTour	VARCHAR(45)	Tên tour	NULL
ngayBD	DATETIME	Ngày bắt đầu	NULL
ngayKT	DATETIME	Ngày kết thúc	NULL
soCho	INT	Số lượng chỗ	0
gia	DECIMAL(10,0)	Giá tour	NOT NULL
anh	VARCHAR(100)	Hình ảnh tour	NULL
moTa	VARCHAR(500)	Mô tả tour	NULL

3.3.2.2. Bảng người dùng

Bảng này có một khóa chính là id (Mã người dùng). Và một thuộc tính đặc biệt là vaiTro(Vai trò người dùng) thuộc tính này giúp hệ thống phân quyền. Bảng này là nơi lưu trữ thông tin người dùng của hệ thống, nó đơn giản là một nơi để người dùng đăng ký tài khoản hoặc đăng nhập, hệ thống sẽ truy xuất dữ liệu để phân quyền cho người dùng khi tham gia vào hệ thống.

Bảng 3.3. Bảng cơ sở dữ liệu người dùng

Tên thuộc tính	Kiểu dữ liệu	Mô tả	Thông tin thêm
id	INT	Mã người dùng	Tự tăng(Khóa chính)
ho	VARCHAR(50)	Họ người dùng	NOT NULL
ten	VARCHAR(50)	Tên người dùng	NOT NULL
email	VARCHAR(45)	Email người dùng	NULL
sdt	VARCHAR(45)	Số điện thoại	NOT NULL
taiKhoan	VARCHAR(45)	Tên tài khoản	NOT NULL
matKhau	VARCHAR(200)	Mật khẩu	NOT NULL
anh	VARCHAR(100)	Hình đại diện	NULL
diaChi	VARCHAR(100)	Địa chỉ người dùng	NULL

vaiTro	VARCHAR(45)	Vai trò người dùng	NOT NULL(USER)
--------	-------------	--------------------	----------------

3.3.2.3. Bảng hóa đơn

Người dùng đặt tour và thanh toán thành công thì hóa đơn được tạo và lưu vào bảng hóa đơn. Khi hóa đơn tạo nó sẽ lấy mã người dùng đang đăng nhập để xác định người dùng nào đặt tour. Vì thế bảng này có một khóa chính là id (Mã hóa đơn) và một khóa ngoại là idNguoiDung (mã người dùng).

Bảng 3.4. Bảng cơ sở dữ liệu hóa đơn

Tên thuộc tính	Kiểu dữ liệu	Mô tả	Thông tin thêm
id	INT	Mã hóa đơn	Tự tăng(Khóa chính)
tongTien	DECIMAL(10,0)	Tổng tiền	NOT NULL
ngayMua	DATETIME	Ngày lập hóa đơn	NULL
idNguoiDung	INT	Mã người dùng	NOT NULL (Khóa ngoại)

3.3.2.4. Bảng chi tiết hóa đơn

Khi người dùng đặt tour và tiến hành thanh toán thành công thì chi tiết hóa đơn được tạo và lưu danh sách các tour người dùng đã đặt. Id (Mã chi tiết hóa đơn) là khóa chính, có idTour (Mã tour) là khóa ngoại, để xác định chi tiết hóa đơn nào của hóa đơn nào thì bảng có một thuộc tính là idHoaDon (Mã hóa đơn) cũng là khóa ngoại.

Bảng 3.5. Bảng cơ sở dữ liệu chi tiết hóa đơn

Tên thuộc tính	Kiểu dữ liệu	Mô tả	Thông tin thêm
id	INT	Mã chi tiết hóa đơn	Tự tăng(Khóa chính)
soLuong	INT	Số lượng tour	NOT NULL
gia	DECIMAL(10,0)	Giá tour	NOT NULL
idHoaDon	INT	Mã hóa đơn	NOT NULL (Khóa ngoại)
idTour	INT	Mã tour	NOT NULL (Khóa ngoại)

3.3.2.5. Bảng bình luận

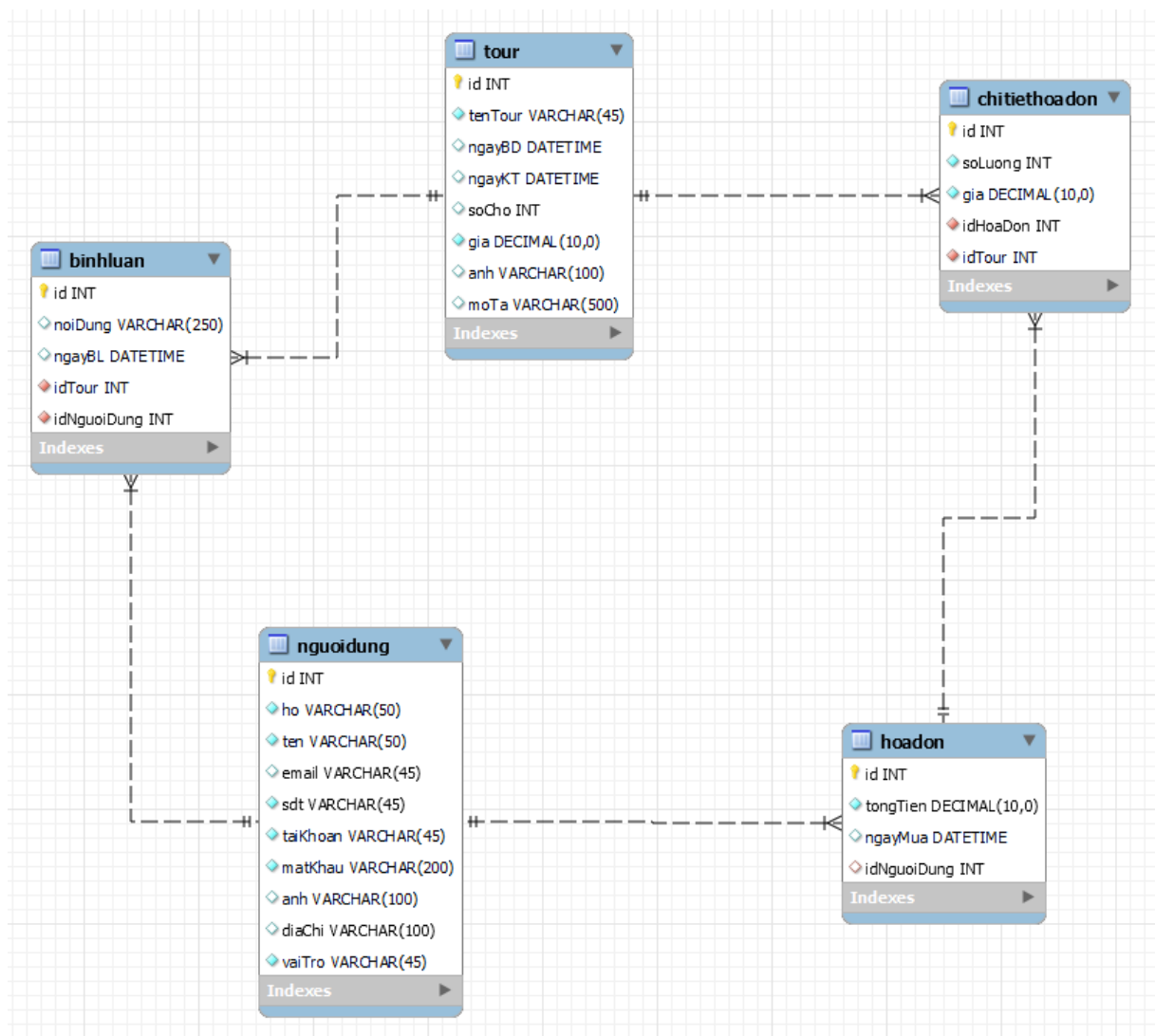
Bảng này lưu lại tất cả bình luận của người dùng để lại, id (Mã bình luận) là khóa chính, để xác định bình luận nào của tour nào, bảng có thuộc tính idTour (Mã tour) là khóa ngoại lưu thông tin mã tour, idNguoiDung (Mã người dùng) là thuộc tính giúp xác định người dùng nào để lại bình luận cho tour nào.

Bảng 3.6. Bảng cơ sở dữ liệu bình luận

Tên thuộc tính	Kiểu dữ liệu	Mô tả	Thông tin thêm
id	INT	Mã bình luận	Tự tăng(Khóa chính)
noiDung	VARCHAR(250)	Nội dung bình luận	NULL
ngayBL	DATETIME	Ngày bình luận	NULL
idTour	INT	Mã tour	NOT NULL (Khóa ngoại)
idNguoiDung	INT	Mã người dùng	NOT NULL (Khóa ngoại)

3.3.3. Mô hình thực thể kết hợp

Một tour có thể có một hoặc nhiều bình luận và một tour cũng có thể có một hoặc nhiều chi tiết hóa đơn. Một hóa đơn gồm nhiều hoặc một chi tiết hóa đơn. Một người dùng có một hoặc nhiều hóa đơn, một hóa đơn chỉ thuộc một người dùng. Một bình luận chỉ thuộc một tour và một người dùng.



Hình 3.18. Mô hình thực thể kết hợp

Chương 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Chương này tổng hợp lại những vấn đề đã đạt được và đưa ra những kết luận cho hệ thống bên cạnh đó còn đưa ra các mặt hạn chế và hướng phát triển để ứng dụng cập nhật và bổ sung thêm những vấn đề ứng dụng còn thiếu.

4.1. Kết quả đã đạt được

Qua khóa luận này em đã được tìm hiểu và nghiên cứu sâu hơn về cấu trúc và luồng xử lý của công nghệ Spring MVC từ đó phát triển được một hệ thống Website tour du lịch với các chức năng cơ bản. Hiện tại website đã cơ bản hoàn thành giao diện và các chức năng cơ bản. Nó đáp ứng được nhu cầu tìm kiếm địa điểm du lịch đẹp, đặc biệt của nước ta cùng với các thông tin, hình ảnh minh họa. Giúp cho người dùng có thêm thông tin và bên cạnh đó người dùng cũng có thể xem được những bình luận của những người đã từng đi, nó là y kiến chân thật của chính khách hàng tự trải nghiệm nên cũng giúp người dùng an tâm phần nào, ngoài ra người dùng có thể trực tiếp đặt tour để trải nghiệm với mức giá vừa phải và đặc biệt tránh được các trường hợp lừa đảo. Ứng dụng còn có các chức năng hỗ trợ các công việc của các quản lý, nó được tích hợp thành các chức năng như, quản lý người dùng, quản lý tour, quản lý đặt tour, thống kê.

4.2. Hạn chế

Bên cạnh những chức năng đã thực hiện được trong ứng dụng. Việc thực hiện dự án với thời gian có hạn và còn một số chức năng chưa đầy đủ tính năng, ngoài ra em còn thiếu kinh nghiệm trong việc học hỏi kiến thức mới đối với bản thân. Vì vậy việc hoàn thành khóa luận còn gặp một số hạn chế như sau:

4.2.1. Về mặt khách hàng

- Đăng ký thông tin chưa kiểm tra thông tin người dùng nhập.
- Chưa lưu các tour người dùng đã đặt khi người dùng đăng xuất.
- Chưa xem lại lịch sử các tour mình đã đặt.
- Không có nhiều hình ảnh, thông tin cho người dùng tham khảo về tour.
- Không có chức năng lọc tour theo phân khúc giá, số lượng, tour nổi bật.
- Chưa có tính năng chỉnh sửa thông tin người dùng và đổi mật khẩu.

- Chưa loại vé có vé cho trẻ em và người lớn.
- Thiếu phương tiện đi lại của tour.
- Không thay đổi, trả tour được.

4.2.2. Về mặt nhân viên và admin

- Thêm tour chưa kiểm tra thông tin nhập.
- Chưa có lọc thông tin người dùng.
- Chức năng chỉnh sửa thông tin người dùng chưa có.
- Không thêm được nhiều tour, nhiều hình ảnh của một tour cùng một lúc được.
- Còn thiếu nhiều thống kê, chưa xuất được file thống kê, báo cáo.

4.3. Hướng phát triển

Bên cạnh cần khắc phục những mặt thiếu sót ở mục hạn chế, trong tương lai website sẽ được nâng cấp theo thời gian để cải thiện việc trải nghiệm của người dùng, ngoài ra website sẽ cập nhật thêm những tính năng mới hứa hẹn sẽ giúp cho website ổn định và đáp ứng đủ nhu cầu của người dùng.

Những tính năng sẽ được website cập nhập và nâng cấp:

- Các hạn chế đã nói trên.
- Thanh toán trực tuyến: người dùng có thể thanh toán trực tuyến bằng momo zalo pay hoặc các ví điện tử khác.
- Danh sách tour được nổi bật, các tour được nhiều bình luận.
- Có thêm nhiều loại thống kê khác.

Những tính năng trên sẽ được nghiên cứu và xây dựng một cách hợp lý. Vì vậy sẽ cần có thời gian để hoàn thiện các chức năng được kể trên. Website này mang một phong cách khác so với các website có mặt trên thị trường, Các cập nhập sắp tới hứa hẹn sẽ mang đến trải nghiệm tốt hơn cho người dùng.

TÀI LIỆU THAM KHẢO

(Theo chuẩn IEEE)

- [1] Đ. Nghĩa, "Nhu cầu thể kế website tăng cao," 23 11 2011. [Online]. Available: <https://matbao.com/vi-VN/Tin-tuc/Tin-tuc-cong-ty/Nhu-cau-thiet-ke-Website-tang-cao.aspx>.
- [2] B. K. Hòa, "Để du lịch là ngành kinh tế mũi nhọn," Để du lịch là ngành kinh tế mũi nhọn, 05 08 2021. [Online]. Available: <https://tinhuylkhanhhoa.vn/tin-chi-tiet/id/10361/De-du-lich-la-nganh-kinh-te-mui-nhon>.
- [3] "Tổng quan về Spring Framework," [Online]. Available: <https://spring.io/projects/spring-framework>.
- [4] hibernate, "Your relational data. Objectively.," [Online]. Available: <http://hibernate.org/orm/>.
- [5] J. L. D. M. Joseph B. Ottinger, Beginning Hibernate: For Hibernate 5, 2016.
- [6] G. K. a. G. G. Christian Bauer, Java Persistence with Hibernate, Second Edition, October 2015.
- [7] viettuts, "JSP là gì? - học JSP cần chuẩn bị những gì?," [Online]. Available: <https://viettuts.vn/jsp>.
- [8] M. n. n. c. n. p. t. t. G. "Bootstrap 4," 2010. [Online]. Available: <https://getbootstrap.com/docs/4.2/getting-started/introduction/>.
- [9] H. B. Tours, "Halong Bay Tours," [Online]. Available: <https://www.halongbaytours.com/>.
- [10] Ivivu, "Ivivu," [Online]. Available: <https://www.ivivu.com/du-lich/>.
- [11] Vietravel, "travel," [Online]. Available: <https://travel.com.vn/>.
- [12] Q. B. Group, "Hướng dẫn css animation scroll hiệu ứng đẹp cho trang web khi lăn

chuột | Scroll down css," 10 10 2021. [Online]. Available:
<https://www.youtube.com/watch?v=LOZFQRiLQO4>.

[13] kigiri, "it-swarm-vi.com," 16 09 2014. [Online]. Available: <https://www.it-swarm-vi.com/vi/javascript/trinh-cao-mau-ngau-nhien/967183954/>.