2

ĐÀO ĐỨC TẤN MSSV: N18DCCN184 CHUYÊN NGÀNH: CÔNG NGHỆ PHẦN MỀM 2018 – 2023 Lớp: D18CQCP02-N

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

------------------------------



**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**Đề tài: “Website quản lý tour du lịch”**

**Người hướng dẫn : ThS. NGUYỄN ANH HÀO**

**Sinh viên thực hiện : ĐÀO ĐỨC TẤN**

**Mã số sinh viên : N18DCCN184  
Lớp : D18CQCP02-N  
Khóa : 2018-2023  
Hệ : ĐẠI HỌC CHÍNH QUY**

**TP.HCM, tháng 8/2022**

TP. HCM

2022

20…

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

---------------------------



**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**Đề tài: “Website quản lý tour du lịch”**

**Người hướng dẫn : ThS. NGUYỄN ANH HÀO**

**Sinh viên thực hiện : ĐÀO ĐỨC TẤN**

**Mã số sinh viên : N18DCCN184  
Lớp : D18CQCP02-N  
Khóa : 2018-2023  
Hệ : ĐẠI HỌC CHÍNH QUY**

**TP.HCM, tháng 8/2022**

**LỜI CÁM ƠN**

Lời đầu tiên, em xin gửi lời tri ân sâu sắc đến các thầy cô trường Học Viện Công Nghệ Bưu Chính Viễn thông cơ sở tại TP.HCM đã tận tình dẫn dắt và truyền đạt cho em rất nhiều kiến thức quý báu trong các học kỳ vừa qua.

Đặc biệt, em xin gửi lời cảm ơn chân thành đến ThS.Nguyễn Anh Hào. Thầy đã hướng dẫn tận tình, truyền đạt kiến thức, chỉ bảo cho em trong suốt thời gian học tập và thực hiện đề tài. Kính chúc thầy và gia đình nhiều sức khoẻ và thành công trong cuộc sống.

Và xin chân thành cảm ơn tất cả các bạn đã luôn sát cánh, giúp đỡ, động viên mình trong những thời điểm khó khăn, tiếp thêm động lực và ý chí giúp vượt qua khó khăn trong suốt quá trình học tập tại trường.

Tuy nhiên, vì thời gian và kiến thức chuyên môn còn hạn chế cùng nhiều yếu tố khách quan khác nên trong quá trình thực hiện đề tài không tránh khỏi những sai sót. Kính mong được lời nhận xét và góp ý của thầy cô, để em có thể hoàn thiện bản thân mình hơn. Một lần nữa, em xin chân thành cảm ơn!

TP.HCM, tháng 8 năm 2022

Sinh viên thực hiện

Đào Đức Tấn

**MỤC LỤC**

[CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI 1](#_Toc112355631)

[1.1 Tổng quan. 1](#_Toc112355632)

[1.2 Tìm hiểu “Website quản lý tour du lịch”. 1](#_Toc112355633)

[1.2.1. Lý thuyết. 1](#_Toc112355634)

[1.2.2. Thực hành. 1](#_Toc112355635)

[1.3 Công cụ sử dụng. 2](#_Toc112355636)

[1.4 Mục tiêu đề tài. 2](#_Toc112355637)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT ĐỀ TÀI 3](#_Toc112355638)

[2.1 Xây dựng hệ thống. 3](#_Toc112355639)

[2.1.1 Giới thiệu Spring. 3](#_Toc112355640)

[2.1.1.1 Giới thiệu Spring Framework. 3](#_Toc112355641)

[2.1.1.2 Mô hình MVC. 3](#_Toc112355642)

[2.1.1.3 Lợi ích của Spring MVC. 5](#_Toc112355643)

[2.1.1.4 Mô hình hoạt động Spring MVC. 6](#_Toc112355644)

[2.1.2 Tổng quan về RESTful API. 7](#_Toc112355645)

[2.1.3 Tổng quan về Reactjs. 7](#_Toc112355646)

[2.1.4 Giới thiệu về hệ quản trị cơ sở dữ liệu MySQL, công cụ PhpMyAdmin và XAMPP 8](#_Toc112355647)

[2.1.4.1 Giới thiệu về hệ quản trị cơ sở dữ liệu MySQL: 8](#_Toc112355648)

[2.1.4.2 Giới thiệu công cụ PhpMyAdmin: 9](#_Toc112355649)

[2.1.4.3 Giới thiệu XAMPP: 9](#_Toc112355650)

[CHƯƠNG 3: KHẢO SÁT HỆ THỐNG 10](#_Toc112355651)

[3.1 Khảo sát hệ thống website quản lý tour du lịch. 10](#_Toc112355652)

[3.2 Các chức năng của hệ thống. 10](#_Toc112355653)

[3.2.1 Yêu cầu chức năng của hệ thống. 10](#_Toc112355654)

[3.2.2 Yêu cầu phi chức năng. 11](#_Toc112355655)

[CHƯƠNG 4: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 12](#_Toc112355656)

[4.1 Mô hình chức năng nghiệp vụ (BFD – Business Function Diagram). 12](#_Toc112355657)

[4.2 Mô hình luồng dữ liệu (DFD – Data Flow Diagram). 13](#_Toc112355658)

[4.3 Thiết lập mô hình thực thể ERD (ERD – Entity Relationship Diagram). 14](#_Toc112355659)

[4.3.1 Xác định các thực thể. 14](#_Toc112355660)

[4.3.2 Mô hình thực thể kết hợp ERD. 14](#_Toc112355661)

[4.3.3 Mô hình dữ liệu quan hệ. 15](#_Toc112355662)

[4.4 Thiết kế dữ liệu. 16](#_Toc112355663)

[4.4.1 Mô hình Diagram. 16](#_Toc112355664)

[4.4.2 Từ điển dữ liệu. 16](#_Toc112355665)

[4.5 Ràng buộc toàn vẹn và tầm ảnh hưởng. 22](#_Toc112355666)

[4.5.1 Ràng buộc toàn vẹn miền giá trị. 22](#_Toc112355667)

[4.5.2 Ràng buộc toàn vẹn liên thuộc tính. 22](#_Toc112355668)

[4.6 Sơ đồ Use case và sơ đồ tuần tự các chức năng. 23](#_Toc112355669)

[4.6.1 Sơ đồ Use case 23](#_Toc112355670)

[4.6.2 Sơ đồ tuần tự 25](#_Toc112355671)

[CHƯƠNG 5: DEMO CHƯƠNG TRÌNH 37](#_Toc112355672)

[CHƯƠNG 6: KẾT LUẬN 44](#_Toc112355673)

[6.1 Kết luận 44](#_Toc112355674)

[6.1.1 Kết quả đạt được 44](#_Toc112355675)

[6.1.2 Hạn chế 44](#_Toc112355676)

[6.2 Hướng phát triển 44](#_Toc112355677)

[CHƯƠNG 7: TÀI LIỆU THAM KHẢO 45](#_Toc112355678)

# CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

## 1.1 Tổng quan.

Hiện tại, Online là một thị trường lớn và đầy tiềm năng đối với kinh doanh. Trong khi đó, người tiêu dùng đang có xu hướng “online hóa” việc mua sắm hàng hoá và dịch vụ để tiết kiệm thời gian, công sức, do đó, các doanh nghiệp cũng đang dần tiến sâu vào thị trường kinh doanh trực tuyến này với mong muốn tăng hiệu suất và doanh thu bán hàng.

Được biết, khi bán hàng theo phương pháp truyền thống rất bất cập trong vấn đề quảng cáo sản phẩm cũng như thống kê doanh thu. Việc thống kê báo cáo doanh thu hoàn toàn dựa trên phần mềm excel, việc này đẫn đến một số vấn đề khá khó khăn như thời gian xuất một báo cáo rất lâu, tốn nhiều thời gian và nhân lực, nhập liệu nếu có nhầm lẫn sẽ không được cảnh báo và xử lý, chẳng hạn như nếu xuất một sản phẩm mà trong kho số lượng tồn của sản phẩm đó không đủ để xuất hay sản phẩm đã hết hàng nhưng khách hàng không được thông báo về tình trạng này, nhân viên cũng không được nhắc để nhập thêm loại hàng ấy. Ngoài ra, khách hàng không chủ động được trong việc đặt trước và mua hàng của shop, mọi việc bán và mua phải diễn ra đồng thời giữa cả bên mua và bên bán.

Do đó, vấn đề xây dựng một shop bán hàng online rất cần thiết. Việc này giúp shop bán hàng quản lý hàng hóa tốt hơn, thống kê báo cáo doanh thu nhanh chóng, chính xác, giảm nhân lực, giảm chi phí và thời gian. Việc quảng cáo sản phẩm và thương hiệu cũng dễ dàng hơn. Về mặt khách hàng, một shop online sẽ giúp họ tiết kiệm thời gian mua sắm cũng như giúp “đặt chỗ” trước những sản phẩm mà họ yêu thích, không sợ bị mua trước hay hết hàng khi họ không đến shop, chủ động trong việc xem và đặt hàng.

Vì các vấn đề đã nêu, em chọn đề tài **“Website quản lý tour du lịch”** nhằm giúp các công ty du lịch cải thiện được vấn đề quản lý, quảng cáo dịch vụ của mình. Đồng thời phù hợp với xu hướng phát triển của thế giới hiện nay.

Chương trình này xây dựng trên nền tảng Spring boot. Với kiến thức còn hạn chế, chắc chắn không thể tránh khỏi những sai sót. Vì vậy, em rất mong nhận được mọi sự góp ý của quý thầy cô để em có thể hoàn thiện đề tài tốt hơn

## 1.2 Tìm hiểu “Website quản lý tour du lịch”.

### 1.2.1. Lý thuyết.

Tìm hiểu về:

- Spring boot.

- RESTful API.

- Reactjs.

### 1.2.2. Thực hành.

Phân tích và thiết kế cơ sở lưu giữ dữ liệu.

Xây dựng website quản lý tour du lịch trực tuyến với các chức năng:

* Thêm/xóa/sửa các danh mục cho admin.
* Phân công hướng dẫn viên.
* Đăng ký, đăng nhập, tạo tài khoản.
* Tra cứu theo tiêu chí.
* Xem danh sách tour, chi tiết tour.
* Đặt tour, đặt câu hỏi, đánh giá tour.
* In hoá đơn, xuất danh sách khách hàng.
* Báo cáo doanh thu.

## 1.3 Công cụ sử dụng.

* PhpMyAdmin.
* Intelliji 2020.1.
* Visual Studio Code.

## 1.4 Mục tiêu đề tài.

Đảm bảo việc quản lý, đặt tour được nhanh chóng, chính xác:

* Giúp nhân viên quản lý tình hình hoạt động của công ty, quản lý các danh mục, phân công hướng dẫn viên, in hoá đơn, thống kê.
* Đảm bảo khách hàng có thông tin chính xác về tour du lịch, đặt tour diễn ra đơn giản.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT ĐỀ TÀI

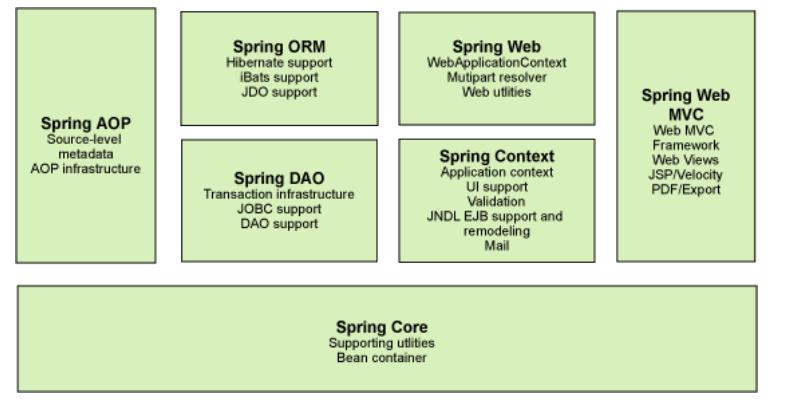
## 2.1 Xây dựng hệ thống.

### 2.1.1 Giới thiệu Spring.

### 2.1.1.1 Giới thiệu Spring Framework.

Spring Framework là một framework mã nguồn mở được viết bằng Java. Nó được xem như là một giải pháp kiến trúc tốt nhất của Java EE hiện nay. Theo thiết kế, framework này giảm nhẹ công việc kỹ thuật cho lập trình viên Java, để họ tập trung sâu vào các công việc nghiệp vụ của ứng dụng. Đồng thời cung cấp một giải pháp toàn diện để họ thực hiện ứng dụng một cách tiện lợi nhất, chặt chẽ nhất đồng thời dễ dàng bảo trì, bảo dưỡng sau này.

Các module chính: Spring được tổ chức thành 7 modules:



*Hình 1: Các module của Spring*

## 2.1.1.2 Mô hình MVC.

**Spring Core**

Core package là phần lõi của framework, cung cấp những đặc tính IoC (Inversion of Control) và DI (Dependency Injection). BeanFactory đảm nhận việc sản sinh và móc nối sự phụ thuộc giữa các đối tượng trong file cấu hình.

**Spring Context/Application Context**

Phía trên của Core package là Context package - cung cấp cách để truy cập đối tượng. Context package kết thừa các đặc tính từ bean package và thêm vào chức năng đa ngôn ngữ (I18N), truyền sự kiện, resource-loading,...

**Spring AOP (Aspect Oriented Programming).**

Spring AOP module tích hợp chức năng lập trình hướng khía cạnh vào Spring framework thông qua cấu hình của nó. Spring AOP module cung cấp các dịch vụ quản lý giao dịch cho các đối tượng trong bất kỳ ứng dụng nào sử dụng Spring. Với Spring AOP chúng ta có thể tích hợp declarative transaction management vào trong ứng dụng mà không cần dựa vào EJB component. Spring AOP module cũng đưa lập trình metadata vào trong Spring. Sử dụng cái này chúng ta có thể thêm annotation (chú thích) vào source code để hướng dẫn Spring và làm thế nào để thực hiện các phương thức sự kiện đã được cài đặt sẵn.

**Spring DAO (Data Access Object)**

DAO package cung cấp cho tầng JDBC, bỏ bớt những coding dài dòng của JDBC và chuyển đổi mã lỗi được xác định bởi database vendor. JDBC package cung cấp cách lập trình tốt như declarative transaction management. Tầng JDBC và DAO đưa ra một cây phân cấp exception để quản lý kết nối đến database, điều khiển exception và thông báo lỗi được ném bởi vendor của database. Tầng exception đơn giản điều khiển lỗi và giảm khối lượng code mà chúng ta cần viết như mở và đóng kết nối. Module này cũng cung cấp các dịch vụ quản lý giao dịch cho các đối tượng trong ứng dụng Spring.

**Spring ORM (Object Relational Mapping)**

ORM package cung cấp tầng tích hợp với object-relational mapping API bao gồm: JDO, Hibernate, iBatis. Sử dụng ORM package bạn có thể sử dụng tất cả các object-relational mapping đó kết hợp với tất cả các đặc tính của Spring như declarative transaction management.

**Spring Web module.**

Spring Web package cung cấp đặc tính của web như: chức năng file-upload, khởi tạo IoC container sử dụng trình lắng nghe serlvet và web-oriented application context. Nằm trên application context module, cung cấp context cho các ứng dụng web. Spring cũng hỗ trợ tích hợp với Struts, JSF và Webwork. Web module cũng làm giảm bớt các công việc điều khiển nhiều request và gắn các tham số của request vào các đối tượng domain.

**Spring MVC Framework.**

Spring Framework là một ứng dụng mã nguồn mở phổ biến để phát triển ứng dụng Java EE dễ dàng hơn. Nó là một container gồm Web Framework (tiếp nhận và xử lý yêu cầu, chia sẽ dữ liệu…) và Web View (quản lý giao diện). MVC Framework được cài đặt đầy đủ các đặc tính của MVC pattern để xây dựng các ứng dụng Web. Các thành phần gồm View (JSP, Velocity, Tiles và generation of PDF và Excel file…), Model (domain model) và Controller (chứa các xử lý yêu cầu).

## 2.1.1.3 Lợi ích của Spring MVC.

* Tất cả các framework đã được tích hợp rất tốt vào Spring.
* Hoạt động rất tốt khi áp dụng theo kiến trúc MVC.
* Sử dụng cơ chế plug-in.
* Kết hợp rất tốt với các O/R (object-relational) Mapping frameworks như là

Hibernate.

* Dễ Testing ứng dụng.
* Ít phức tạp hơn so với các framework khác.
* Cộng đồng người sử dụng rất nhiều, nhiều sách mới được xuất bản.

### 2.1.1.4 Mô hình hoạt động Spring MVC.

*Hình 2: Mô hình MVC*

**Model**

Model gồm các lớp java có nhiệm vụ:

* Biểu diễn data và cho phép đọc/ghi data thông qua các phương thức getter/setter theo qui ước trong JavaBean.
* Buộc dữ liệu form giao diện tức là nhận dữ liệu từ tham số và cung cấp dữ liệu

để

trình bày lên giao diện.

* Thi hành các yêu cầu (tính toán, kết nối CSDL …)
* Trả về các giá trị tính toán theo yêu cầu của Controller

**View**

Bao gồm các mã tương tự như JSP, HTML, CSS, XML, Javascript, JSON… để hiển thị giao diện người dùng, các dữ liệu trả về từ Model thông qua Controller…

**Controller**

Đồng bộ hoá giữa View và Model. Tức là với một trang JSP này thì sẽ tương ứng với lớp java nào để xử lý nó và ngược lại, kết quả sẽ trả về trang jsp nào. Nó đóng vai trò điều tiết giữa View và Model.

Như vậy, chúng ta có thể tách biệt được các mã java ra khỏi mã html. Do vậy, nó đã giải quyết được các khó khăn sự phụ thuộc nghiệp vụ lẫn nhau giữa các thành viên tham gia dự án. Người thiết kế giao diện và người lập trình java có thể mang tính chất độc lập tương đối.

Việc kiểm lỗi hay bảo trì sẽ dễ dàng hơn, việc thay đổi các theme của trang web cũng dễ dàng hơn.

## 2.1.2 Tổng quan về RESTful API.

REST là viết tắt của Representational State Transfer. RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource. RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile…) khác nhau giao tiếp với nhau. Chức năng quan trọng nhất của REST là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE…) và cách định dạng các URL cho ứng dụng web để quản các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API.

## 2.1.3 Tổng quan về Reactjs.

Reactjs là một thư viện Javascript mã nguồn mở hỗ trợ xây dựng các thành phần giao diện nhanh gọn và tiện lợi. Bình thường các lập trình viên sẽ nhúng javascript vào code HTML thông qua các attribute như AngularJS nhưng với Reactjs làm việc như một thư viện cho phép nhúng HTML vào javascript thông qua JSX. Qua đó bạn có thể dễ dàng lồng các đoạn HTML vào trong JSX làm cho các component dễ hiểu và dễ sử dụng hơn.

Làm gì cũng phải có nguyên nhân, người dùng sử dụng Reactjs ngày càng nhiều thì Reactjs phải mang lại những lợi ích tốt.

* ReactJS giúp cho việc viết các đoạn code Javascript sẽ trở nên dễ dàng hơn vì nó sử dụng một cú pháp đặc biệt đó chính là cú pháp JSX. Thông qua JSX cho phép nhúng code HTML và Javascript.
* ReactJS cho phép Developer phá vỡ những cấu tạo UI phức tạp thành những component độc lập. Dev sẽ không phải lo lắng về tổng thể ứng dụng web, giờ đây Developer dễ dàng chia nhỏ các cấu trúc UI/UX phức tạp thành từng component đơn giản hơn.
* Đi kèm với ReactJS là rất nhiều các công cụ phát triển giúp cho việc debug code một cách dễ dàng hơn.
* Một trong những ưu điểm nữa của ReactJS đó là sự thân thiện với SEO. Hầu như các JS Frameworks không thân thiện với các tìm kiếm mặc dù đã được cải thiện nhiều nhưng dưới sự hỗ trợ của các render dữ liệu trả về dưới dạng web page giúp cho SEO chuẩn hơn.

## 2.1.4 Giới thiệu về hệ quản trị cơ sở dữ liệu MySQL, công cụ PhpMyAdmin và XAMPP

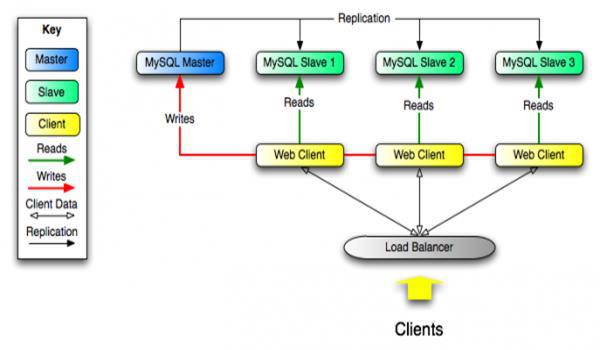
### 2.1.4.1 Giới thiệu về hệ quản trị cơ sở dữ liệu MySQL:

* **Giới thiệu:**

A picture containing logo

Description automatically generated**MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở** (gọi tắt là RDBMS) hoạt động theo **mô hình client-server**. Với **RDBMS** là viết tắt của **Relational Database Management System**. MySQL được tích hợp apache, PHP. MySQL quản lý dữ liệu thông qua các cơ sở dữ liệu. Mỗi cơ sở dữ liệu có thể có nhiều bảng quan hệ chứa dữ liệu. MySQL cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL. MySQL được phát hành từ thập niên 90s.

*Hình 4 Hệ quản trị cơ sở dữ liệu MySQL*

* **Cách thức hoạt động của MySQL:**

Cách vận hành chính trong môi trường MySQL như sau:

* MySQL tạo ra bảng để lưu trữ dữ liệu, định nghĩa sự liên quan giữa các bảng đó.
* Client sẽ gửi yêu cầu SQL bằng một lệnh đặc biệt trên MySQL.
* Ứng dụng trên server sẽ phản hồi thông tin và trả về kết quả trên máy client.

### 2.1.4.2 Giới thiệu công cụ PhpMyAdmin:

PhpMyAdmin là phần mềm mã nguồn mở được viết bằng ngôn ngữ PHP giúp quản trị cở sở dữ liệu MySQL thông qua giao diện web**.** Sử dụng phpMyadmin người dùng có thể thực hiện được nhiều tác vụ khác nhau như khi sử dụng cửa sổ dòng lệnh. Các tác vụ này bao gồm việc tạo, cập nhật và xoá các cơ sở dữ liệu, các bảng, các trường, dữ liệu trên bảng, phân quyền và quản lý người dùng, ...

### 2.1.4.3 Giới thiệu XAMPP:

Phần mềm XAMPP là một loại ứng dụng phần mềm khá phổ biến và thường hay được các lập trình viên sử dụng để xây dựng và phát triển các dựa án website theo ngôn ngữ PHP. XAMPP được sử dụng cho mục đích nghiên cứu, phát triển website qua Localhost của máy tính cá nhân. XAMPP được ứng dụng trong nhiều lĩnh vực từ học tập đến nâng cấp, thử nghiệm Website của các lập trình viên.

# CHƯƠNG 3: KHẢO SÁT HỆ THỐNG

## 3.1 Khảo sát hệ thống website quản lý tour du lịch.

Một công ty đặt tour du lịch trực tuyến được tổ chức như sau:

* Công ty có nhiều nhân viên. Thông tin của một nhân viên bao gồm mã nhân viên, họ tên, địa chỉ, số điện thoại, email, mật khẩu, quyền….
* Khách hàng của công ty có thông tin bao gồm: mã khách hàng để phân biệt, họ tên, giới tính, số điện thoại, mật khẩu email, trạng thái …
* Mỗi khách hàng và nhân viên đều có riêng 1 tài khoản đăng nhập là số điện thoại và mật khẩu. Mỗi nhân viên và khách hàng chỉ thuộc một quyền và một quyền sẽ có nhiều người dùng
* Công ty có nhiều tour. Một tour bao gồm những thông tin như mã tour, tiêu đề tour, hướng dẫn viên, ngày khởi hành, ngày kết thúc, giá tiền, mô tả, loại tour, …
* Một tour có nhiều địa danh thăm quan, 1 địa danh thăm quan có thể được nhiều tour đến. Mỗi địa danh sẽ có thông tin bao gồm mã địa danh, tên địa danh, hình ảnh.
* Một tour có nhiều phương tiện di chuyển, 1 phương tiện di chuyển có thể được nhiều tour sử dụng. Mỗi phương tiện sẽ có thông tin bao gồm mã phương tiện (biển số xe), tên phương tiện, tên tài xế, loại xe.
* Một tour có nhiều dịch vụ, 1 dịch vụ có thể có trong nhiều tour. Mỗi dịch vụ sẽ có thông tin bao gồm mã dịch vụ, tên dịch vụ.
* Một tour có 1 hướng dẫn viên, 1 hướng dẫn viên có thể hướng dẫn nhiều tour.
* Một tour có thể có nhiều khách đặt, một khách có thể đặt nhiều tour khác nhau.
* Một khách hàng có thể đánh giá tour của mình đã đặt sau khi tour đã kết thúc, một tour có thể được nhiều khách hàng đánh giá.
* Một khách hàng có thể đặt câu hỏi ở nhiều tour, một tour có thể được nhiều khách hàng đặt câu hỏi.

## 3.2 Các chức năng của hệ thống.

Đề tài xây dựng website quản lý tour du lịch có các chức năng của hệ thống như sau:

### 3.2.1 Yêu cầu chức năng của hệ thống.

Công việc của người quản lý:

* Thêm, xóa, sửa các danh mục.
* In hoá đơn.
* Phân công hướng dẫn viên.
* Khoá/mở khoá tài khoản khách hàng.
* Trả lời các câu hỏi của khách hàng.
* Báo cáo doanh thu.

Công việc của hướng dẫn viên:

* Đăng nhập
* Xem thông tin cá nhân.
* Chỉnh sửa thông tin.
* Xem danh sách tour mà mình được phân công.
* Xem danh sách khách hàng của tour.

Công việc của người mua hàng:

* Đăng ký tài khoản/Đăng nhập.
* Xem thông tin cá nhân.
* Chỉnh sửa thông tin cá nhân.
* Xem thông tin tour.
* Lọc tour.
* Đặt tour/ huỷ tour.
* Đánh giá tour/ đặt câu hỏi.

Đối tượng tham gia hệ thống:

* Người quản lý
* Khách hàng.
* Hướng dẫn viên.

### 3.2.2 Yêu cầu phi chức năng.

* Hệ thống đáng tin cậy, chính xác, giao diện thân thiện, truy cập dữ liệu nhanh chóng.
* Đảm bảo tính bảo mật cho người điều hành hệ thống.
* Phải có tính linh hoạt cao

# CHƯƠNG 4: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## 4.1 Mô hình chức năng nghiệp vụ (BFD – Business Function Diagram).

Chức năng là công việc mà tổ chức cần làm và được phân theo nhiều mức từ tổng hợp đến chi tiết.

Mỗi chức năng có một tên duy nhất, các chức năng khác nhau phải có tên khác nhau. Để xác định tên cho các chức năng một cách chính xác và cụ thể, có thể bàn luận và nhất trí với người sử dụng.

Hình thức biểu diễn:

Tên chức năng

*Mỗi chức năng được phân rã thành các chức năng con. Các chức năng con có quan hệ phân cấp với chức năng cha.*

*Chart, box and whisker chart

Description automatically generatedBiểu diễn mối quan hệ phân cấp chức năng như sau:*

Diagram

Description automatically generatedHình 4.1. Mô hình phân rã chức năng phía frontend được biểu diễn thành hình cây phân cấp

Hình 4.2. Mô hình phân rã chức năng phía backend được biểu diễn thành hình cây phân cấpDiagram

Description automatically generated

## 4.2 Mô hình luồng dữ liệu (DFD – Data Flow Diagram).

Mô hình luồng dữ liệu là một công cụ mô tả quan hệ thông tin giữa các công việc.

Biểu đồ luồng dữ liệu gồm có 5 yếu tố chính:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Chức năng** | **Luồng dữ liệu** | **Kho dữ liệu** | **Tác nhân ngoài** | **Tác nhân trong** |
| Định nghĩa | Nhiệm vụ xử lý thông tin | Thông tin vào/ra một chức năng xử lý | Nơi lưu trữ thông tin trong một thời gian | Người hay tổ chức ngoài hệ thống có giao tiếp với hệ thống | Một chức năng hay một hệ con của hệ thống nhưng được mô tả ở trang khác |
| Tên đi kèm | Động từ  (+ bổ ngữ) | Danh từ  (+ tính từ) | Danh từ  (+ tính từ) | Danh từ | Động từ |
| Biểu đồ |  | Tên | Tên | Tên | Tên |
| Ví dụ |  | Hóa đơn đã xác nhận chi | Đơn hàng | Nhân viên | Thanh toán |

#### Bảng 4.1. Các đối tượng cơ bản sử dụng để thiết kế biểu đồ luồng dữ liệu.

Diagram

Description automatically generated

#### Hình 4.3. Biểu đồ mức ngữ cảnh

## 4.3 Thiết lập mô hình thực thể ERD (ERD – Entity Relationship Diagram).

### 4.3.1 Xác định các thực thể.

* **Tour** (idTour, title, dateStart, dateEnd, price, description, type)
* **Place** (idPlace, name, image)
* **Vehicle** (idVehicle, name, driverName, type)
* **Employee** (idEmployee, name, address, email, phone, password, role)
* **Customer** (idCustomer, name, gender, email, phone, password, status)
* **Service** (idService, name)

### 4.3.2 Mô hình thực thể kết hợp ERD.

Diagram

Description automatically generated

#### Hình 4.4. ERD.

### 4.3.3 Mô hình dữ liệu quan hệ.

* **tour** (idTour, title, idEmployee, dateStart, dateEnd, price, description, type)
* **place** (idPlace, name, image)
* **vehicle** (lisencePlate, name, driverName, type)
* **employee** (idEmployee, name, address, email, phone, password, role)
* **customer** (idCustomer, name, gender, email, phone, password, status)
* **service** (idService, name)
* **booking** (idCustomer, idTour, dateBooking, quantity, totalPrice, judge)
* **ask** (idCustomer, idTour, question, answer)
* **placetour** (idTour, idPlace)
* **vehicletour** (idTour, idPlace)
* **servicetour** (idTour, idService)

## 4.4 Thiết kế dữ liệu.

### 4.4.1 Mô hình Diagram.

Graphical user interface

Description automatically generated

#### Hình 4.5. Diagram

### 4.4.2 Từ điển dữ liệu.

* Bảng: PLACE

**PLACE** (IDPLACE, NAME, IMAGE)

Graphical user interface, application

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDPLACE | Varchar(4) | Mã địa danh |  | X |  | X |
| 2 | NAME | Varchar(50) | Tên địa danh |  |  |  | X |
| 3 | IMAGE | Varchar(200) | Đường dẫn đến hình ảnh |  |  |  | X |

* Bảng: VEHICLE

**VEHICLE** (IDVEHICLE, NAME, DRIVERNAME, TYPE)

Graphical user interface, application

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDVEHICLE | Varchar(10) | Mã phương tiện |  | X |  | X |
| 2 | NAME | Varchar(50) | Tên phương tiện |  |  |  | X |
| 3 | DRIVERNAME | Varchar(50) | Tên tài xế |  |  |  |  |
| 4 | TYPE | Varchar(15) | Loại xe |  |  |  |  |

* Bảng: SERVICE

**SERVICE** (IDSERVICE, NAME)

Graphical user interface, application

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDSERVICE | Varchar(4) | Mã dịch vụ |  | X |  | X |
| 2 | NAME | Varchar(50) | Tên dịch vụ |  |  |  | X |

* Bảng: CUSTOMER

**CUSTOMER** (IDCUSTOMER, NAME, GENDER, EMAIL, PHONE, PASSWORD, STATUS)

Graphical user interface, text, application

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDCUSTOMER | Varchar(4) | Mã khách hàng |  | X |  | X |
| 2 | NAME | Varchar(50) | Họ tên |  |  |  |  |
| 3 | GENDER | Varchar(11) | Giới tính |  |  |  |  |
| 4 | EMAIL | Varchar(50) | Email | X |  |  |  |
| 5 | PHONE | Varchar(10) | Số điện thoại | X |  |  |  |
| 6 | PASSWORD | Varchar(50) | Mật khẩu |  |  |  |  |
| 7 | STATUS | Varchar(10) | Trạng thái tài khoản |  |  |  |  |

* Bảng: EMPLOYEE

**EMPLOYEE** (IDEMPLOYEE, NAME, ADDRESS, EMAIL, PHONE, PASSWORD, ROLE)

Graphical user interface, text

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDEMPLOYEE | Varchar(4) | Mã nhân viên |  | X |  | X |
| 2 | NAME | Varchar(50) | Tên nhân viên |  |  |  |  |
| 3 | ADDRESS | Varchar(50) | Địa chỉ |  |  |  |  |
| 4 | EMAIL | Varchar(50) | Email | X |  |  |  |
| 5 | PHONE | Varchar(10) | Số điện thoại | X |  |  |  |
| 6 | PASSWORD | Varchar(50) | Mật khẩu |  |  |  |  |
| 7 | ROLE | Varchar(10) | Quyền |  |  |  |  |

* Bảng: BOOKING

**BOOKING** (IDCUSTOMER, IDTOUR, DATEBOOKING, QUANTITYPEOPLE, TOTALPRICE, JUDGE)

Graphical user interface, text, application

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDCUSTOMER | Varchar(4) | Mã khách hàng |  | X |  | X |
| 2 | IDTOUR | Varchar(4) | Mã tour |  | X |  | X |
| 3 | DATEBOOKING | Date | Ngày đặt |  |  |  | X |
| 4 | QUANTITYPEOPLE | Int(2) | Số lượng người tham gia |  |  |  | X |
| 5 | TOTALPRICE | Float | Tổng tiền |  |  |  | X |
| 6 | JUDGE | Varchar(200) | Đánh giá |  |  |  |  |

* Bảng: ASK

**ASK** (IDCUSTOMER, IDTOUR, QUESTION, ANSWER)

Graphical user interface, text, application, chat or text message

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDCUSTOMER | Varchar(4) | Mã khách hàng |  | X |  | X |
| 2 | IDTOUR | Varchar(4) | Mã tour |  | X |  | X |
| 3 | QUESTION | Varchar(200) | Câu hỏi |  |  |  | X |
| 4 | ANSWER | Varchar(200) | Câu trả lời |  |  |  |  |

* Bảng: PLACETOUR

**PLACETOUR** (IDTOUR, IDPLACE)

Graphical user interface, application, chat or text message

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDTOUR | Varchar(4) | Mã tour |  | X |  | X |
| 2 | IDPLACE | Varchar(4) | Mã địa danh |  | X |  | X |

* Bảng: VEHICLETOUR

**VEHICLETOUR** (IDTOUR, IDVEHICLE)

Graphical user interface, application

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDTOUR | Varchar(4) | Mã tour |  | X |  | X |
| 2 | IDVEHICLE | Varchar(10) | Mã phương tiện |  | X |  | X |

* Bảng: SERVICETOUR

**SERVICETOUR** (IDTOUR, IDSERVICE)

Graphical user interface, application

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDTOUR | Varchar(4) | Mã tour |  | X |  | X |
| 2 | IDSERVICE | Varchar(4) | Mã dịch vụ |  | X |  | X |

* Bảng: TOUR

**TOUR** (IDTOUR, TITLE, IDEMPLOYEE, DATESTART, DATEEND, PRICE, DESCRIPTION, TYPE)

Graphical user interface, text, application

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu | Diễn giải | Khóa duy nhất | Khóa chính | Khóa ngoại | Not null |
| 1 | IDTOUR | Varchar(4) | Mã tour |  | X |  | X |
| 2 | TITLE | Varchar(200) | Tiêu đề tour |  |  |  | X |
| 3 | IDEMPLOYEE | Varchar(4) | Mã hướng dẫn viên |  |  | X |  |
| 4 | DATESTART | Date | Ngày khởi hành |  |  |  |  |
| 5 | DATEEND | Date | Ngày kết thúc |  |  |  |  |
| 6 | PRICE | Float | Giá tiền |  |  |  |  |
| 7 | DESCRIPTION | Varchar(300) | Mô tả |  |  |  |  |
| 8 | TYPE | Varchar(200) | Loại tour |  |  |  |  |

## 4.5 Ràng buộc toàn vẹn và tầm ảnh hưởng.

### 4.5.1 Ràng buộc toàn vẹn miền giá trị.

Ràng buộc 1:

* Bối cảnh: TOUR
* Mô tả:
* Giá tiền > 0
* Nội dung:
* ∀ t ∈ TOUR / t.PRICE > 0
* Tầm ảnh hưởng

|  |  |  |  |
| --- | --- | --- | --- |
| Quan hệ | Thêm | Xóa | Sửa |
| TOUR | + | - | +(PRICE) |

Ràng buộc 2:

* Bối cảnh: BOOKING
* Mô tả:
* số lượng >0
* tổng tiền >0
* Nội dung:
* ∀ b ∈ BOOKING / b.QUANTITYPEOPLE > 0
* ∀ b ∈ BOOKING / b.TOTALPRICE > 0
* Tầm ảnh hưởng

|  |  |  |  |
| --- | --- | --- | --- |
| Quan hệ | Thêm | Xóa | Sửa |
| BOOKING | + | - | +(QUANTITYPEOPLE) |
| BOOKING | + | - | +(TOTALPRICE) |

### 4.5.2 Ràng buộc toàn vẹn liên thuộc tính.

Ràng buộc 3:

* Bối cảnh: TOUR
* Mô tả:
* Ngày kết thúc phải sau ngày bắt đầu khuyến mãi
* Nội dung:
* ∀ t ∈ TOUR / t.DATEEND > d.DATESTART
* Tầm ảnh hưởng

|  |  |  |  |
| --- | --- | --- | --- |
| Quan hệ | Thêm | Xóa | Sửa |
| TOUR | + | - | +(DATESTART) |
| TOUR | + | - | +(DATEEND) |

## 4.6 Sơ đồ Use case và sơ đồ tuần tự các chức năng.

### 4.6.1 Sơ đồ Use case

Diagram

Description automatically generated

#### Hình 4.6. Use case quản lý

#### Diagram Description automatically generatedHình 4.7. Use case nhân viên

Diagram

Description automatically generated

#### Hình 4.8. Use case khách hàng

### 4.6.2 Sơ đồ tuần tự

Diagram

Description automatically generated

#### Diagram Description automatically generatedHình 4.9. Sơ đồ tuần tự chức năng quản lý nhân viên

#### Hình 4.10. Sơ đồ tuần tự chức năng quản lý khách hàng

#### Diagram Description automatically generatedDiagram Description automatically generatedHình 4.11. Sơ đồ tuần tự chức năng quản lý địa điểm

#### Hình 4.12. Sơ đồ tuần tự chức năng quản lý phương tiện

Graphical user interface

Description automatically generated

#### Hình 4.13. Sơ đồ tuần tự chức năng phản hồi câu hỏi

Diagram

Description automatically generated

#### Hình 4.14. Sơ đồ tuần tự chức năng quản lý dịch vụ

Graphical user interface, diagram

Description automatically generated with medium confidence

#### Hình 4.15 Sơ đồ tuần tự chức năng phân hướng dẫn viên

Diagram

Description automatically generated

#### Hình 4.16. Sơ đồ tuần tự chức năng in hoá đơn

Diagram

Description automatically generated

#### Diagram Description automatically generatedHình 4.17. Sơ đồ tuần tự chức năng chỉnh sửa thông tin

#### Hình 4.18. Sơ đồ tuần tự chức năng quản lý tour

#### Diagram Description automatically generatedHình 4.19. Sơ đồ tuần tự chức năng đổi mật khẩu

Diagram

Description automatically generated

#### Hình 4.20. Sơ đồ tuần tự chức năng đăng xuất

Diagram

Description automatically generated

#### Diagram Description automatically generated Hình 4.21. Sơ đồ tuần tự chức năng đăng nhập

*Hình 4.22. Sơ đồ tuần tự chức năng thống kê doanh thu*

#### Diagram Description automatically generatedGraphical user interface Description automatically generatedHình 4.23. Sơ đồ tuần tự chức năng lập danh sách khách

#### Hình 4.24. Sơ đồ tuần tự chức năng đặt tour

#### Graphical user interface Description automatically generated with medium confidenceHình 4.25. Sơ đồ tuần tự chức năng xem danh sách tour

Graphical user interface

Description automatically generated

#### Hình 4.26. Sơ đồ tuần tự chức năng xem danh sách khách đặt tour

Graphical user interface, diagram

Description automatically generated

#### Diagram Description automatically generatedHình 4.27. Sơ đồ tuần tự chức năng xem chi tiết tour

#### Hình 4.28. Sơ đồ tuần tự chức năng đăng ký

Diagram

Description automatically generated

#### Hình 4.29. Sơ đồ tuần tự chức năng đặt câu hỏi

Diagram

Description automatically generated

#### Hình 4.30. Sơ đồ tuần tự chức năng đánh giá tour

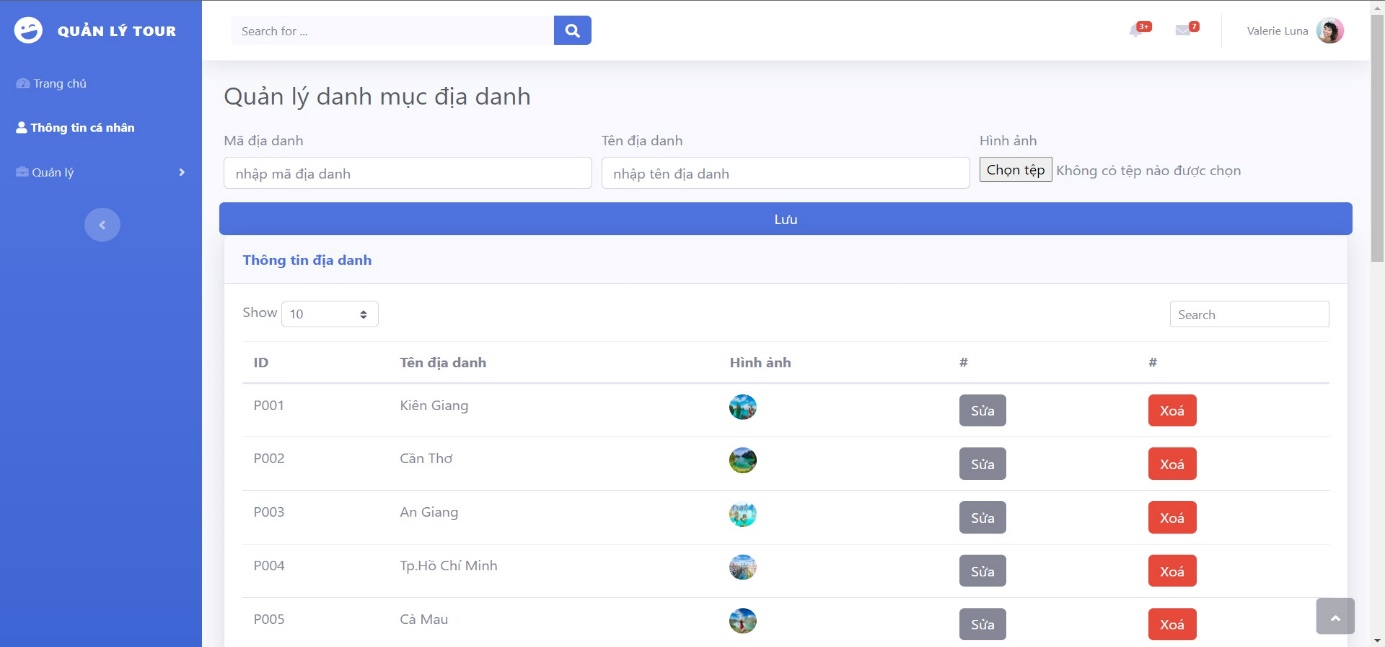
#### Diagram Description automatically generatedHình 4.31. Sơ đồ tuần tự chức năng tìm kiếm

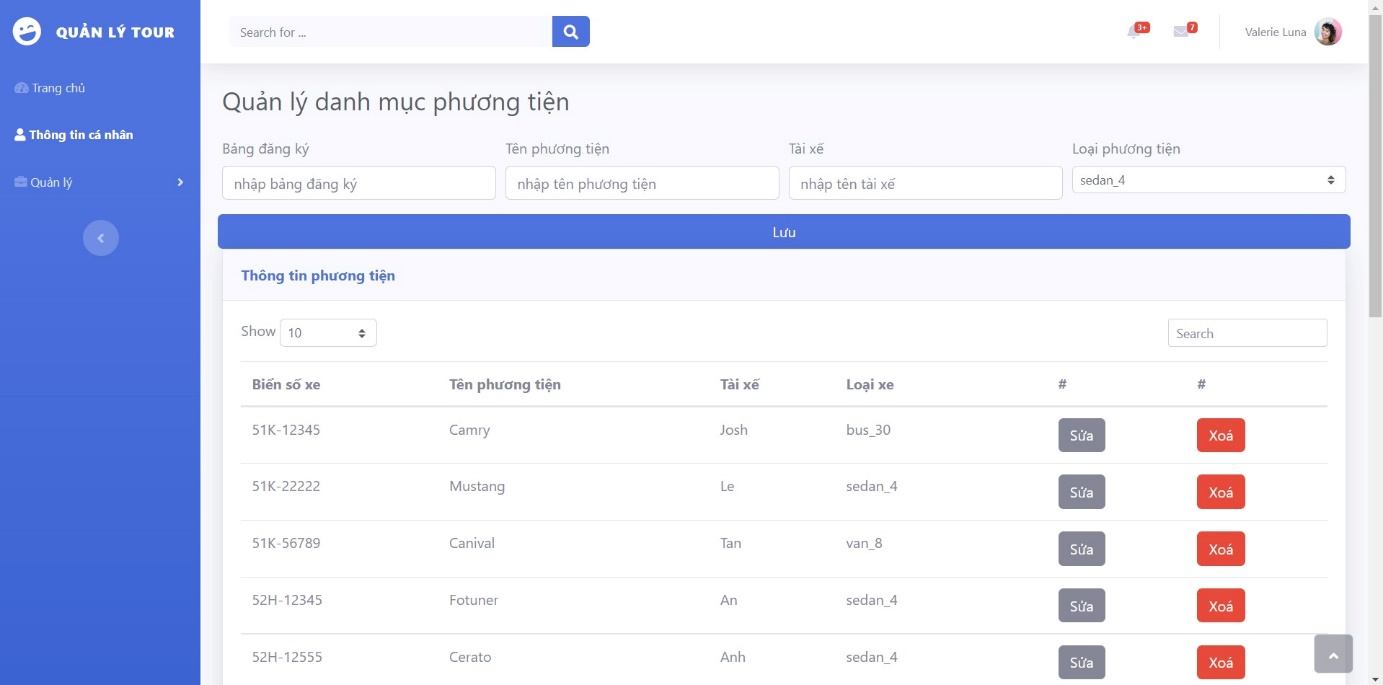
# CHƯƠNG 5: DEMO CHƯƠNG TRÌNH

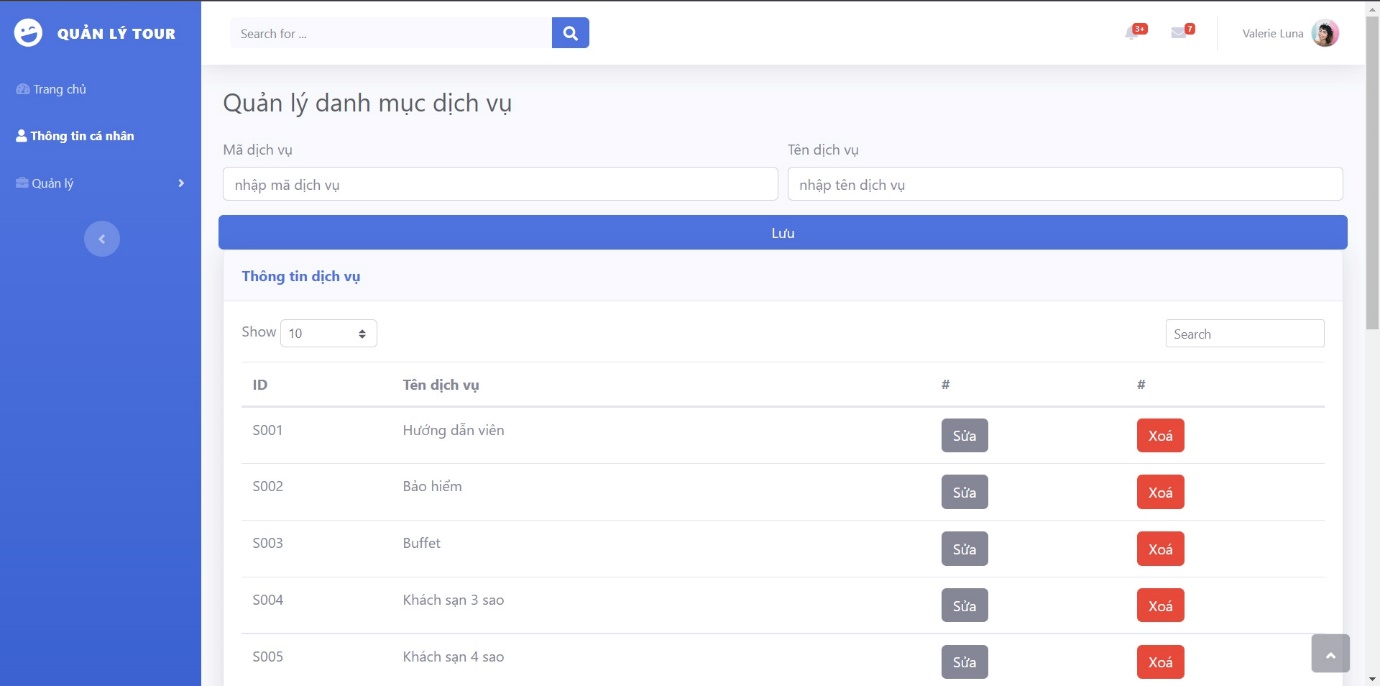
Hình 5.2: Giao diện thông tin cá nhân trang web quản lý

Hình 5.1: Giao diện đăng nhập trang web quản lý

Hình 5.3: Giao diện quản lý địa danh trang web quản lý





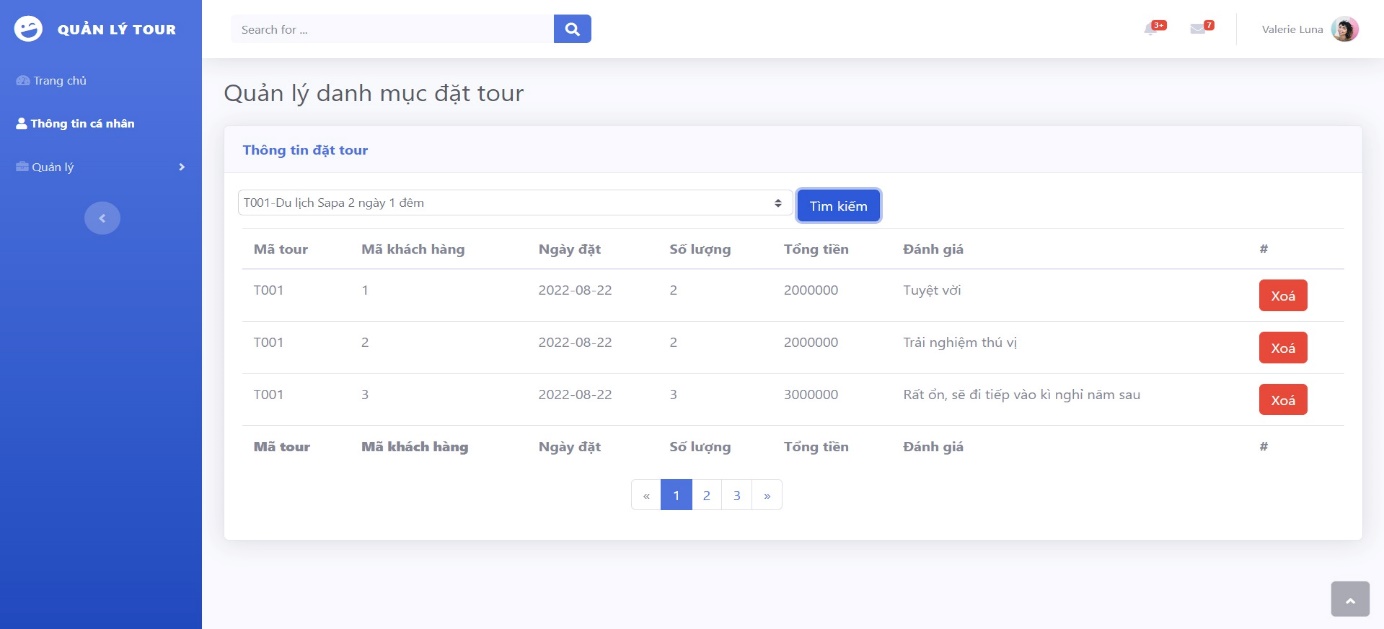
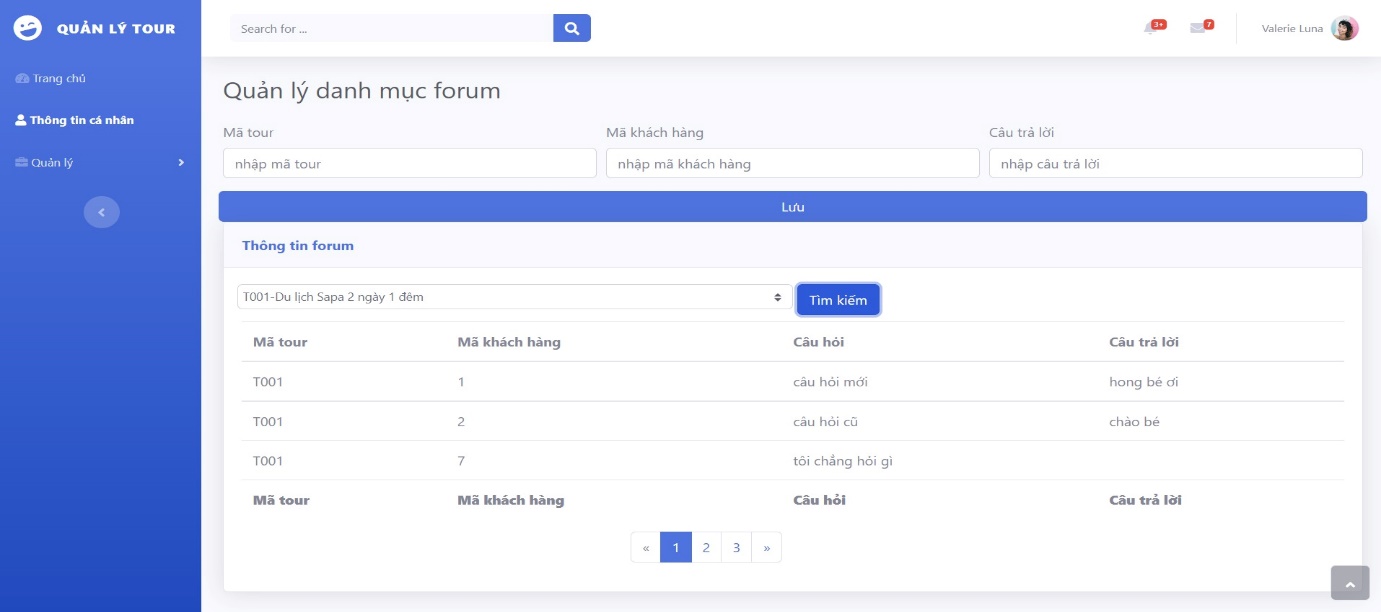


Hình 5.5: Giao diện quản lý dịch vụ trang web quản lý

Hình 5.4: Giao diện quản lý phương tiện trang web quản lý

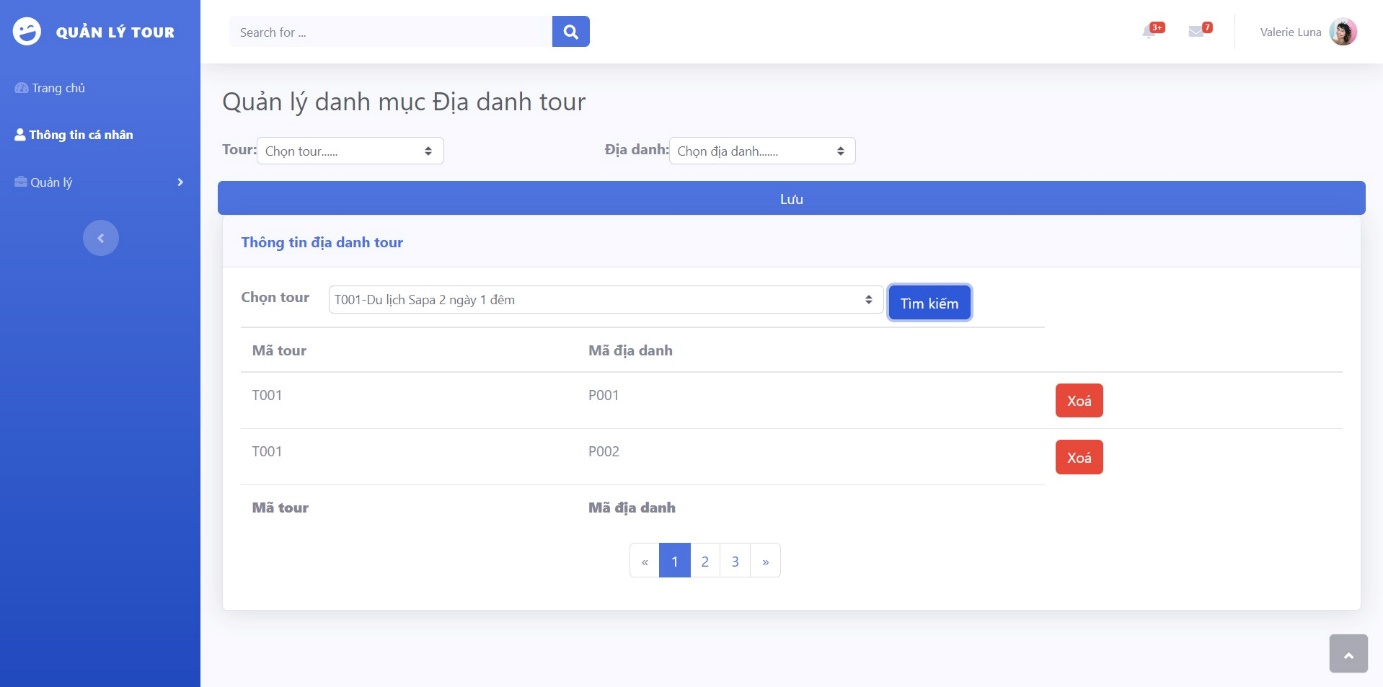


Hình 5.6: Giao diện quản lý tour trang web quản lý

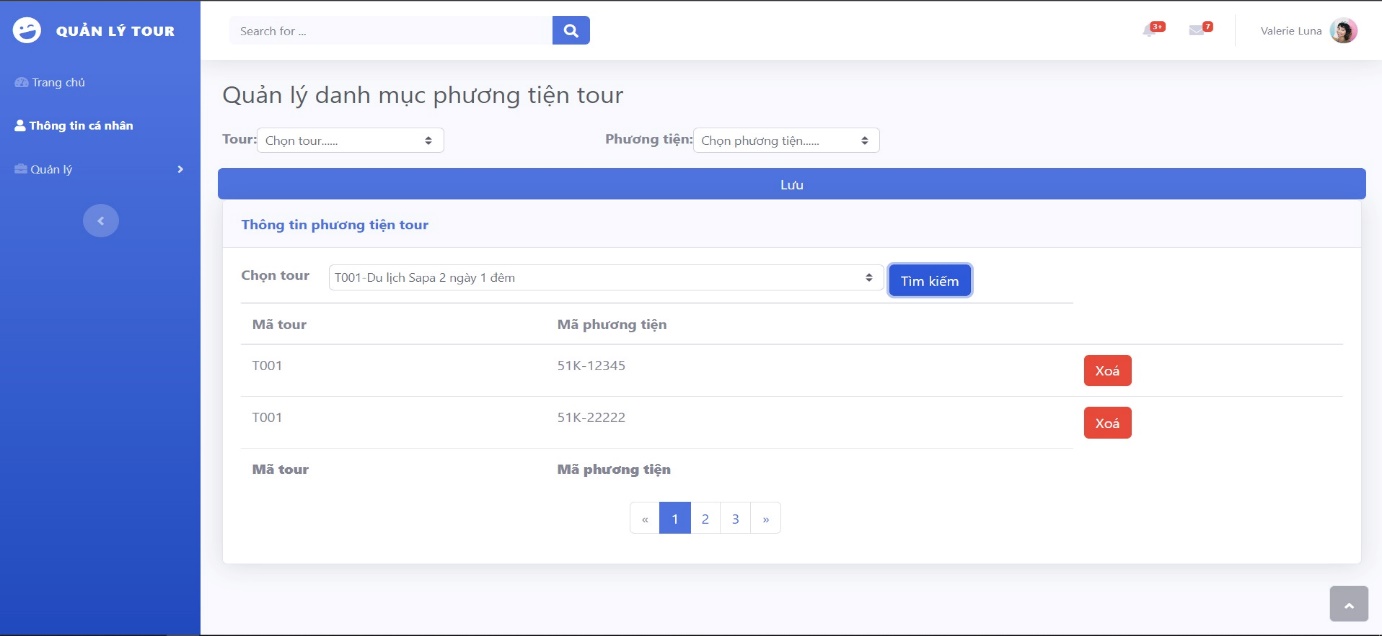


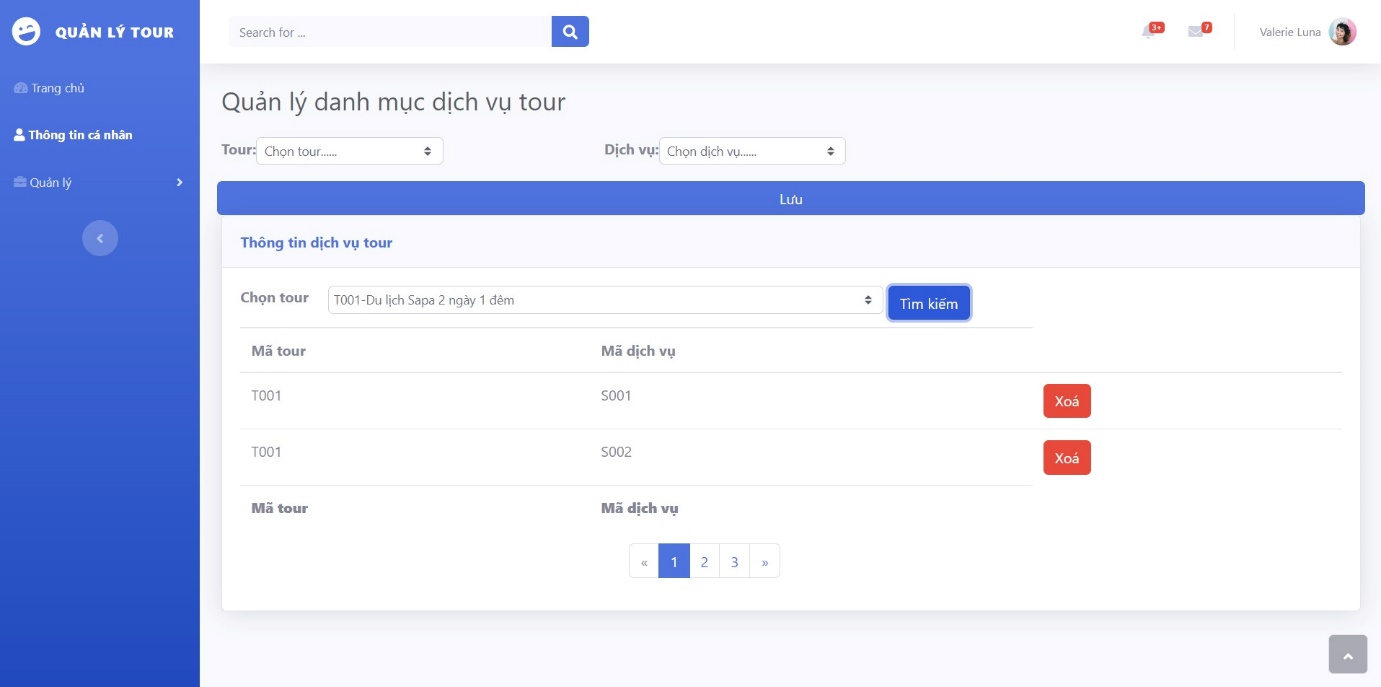
Hình 5.8: Giao diện quản lý forum trang web quản lý

Hình 5.7: Giao diện quản lý booking trang web quản lý



Hình 5.9: Giao diện quản lý địa danh -tour trang web quản lý

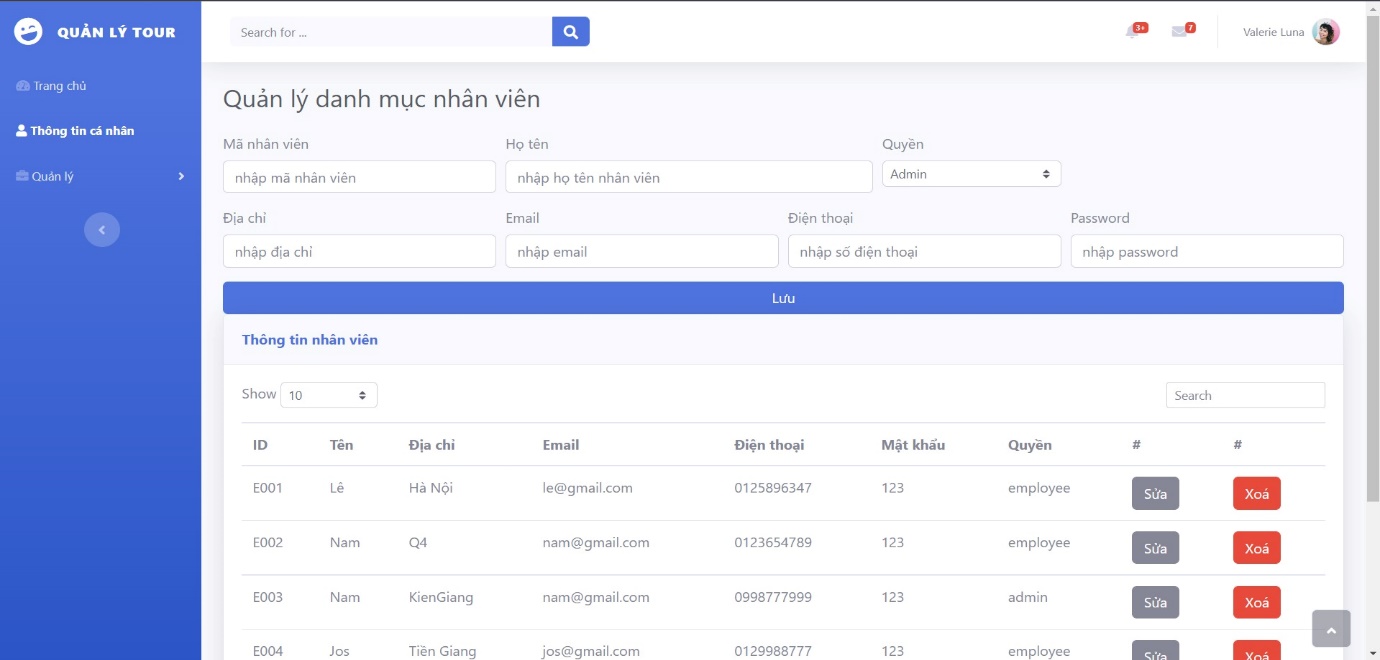


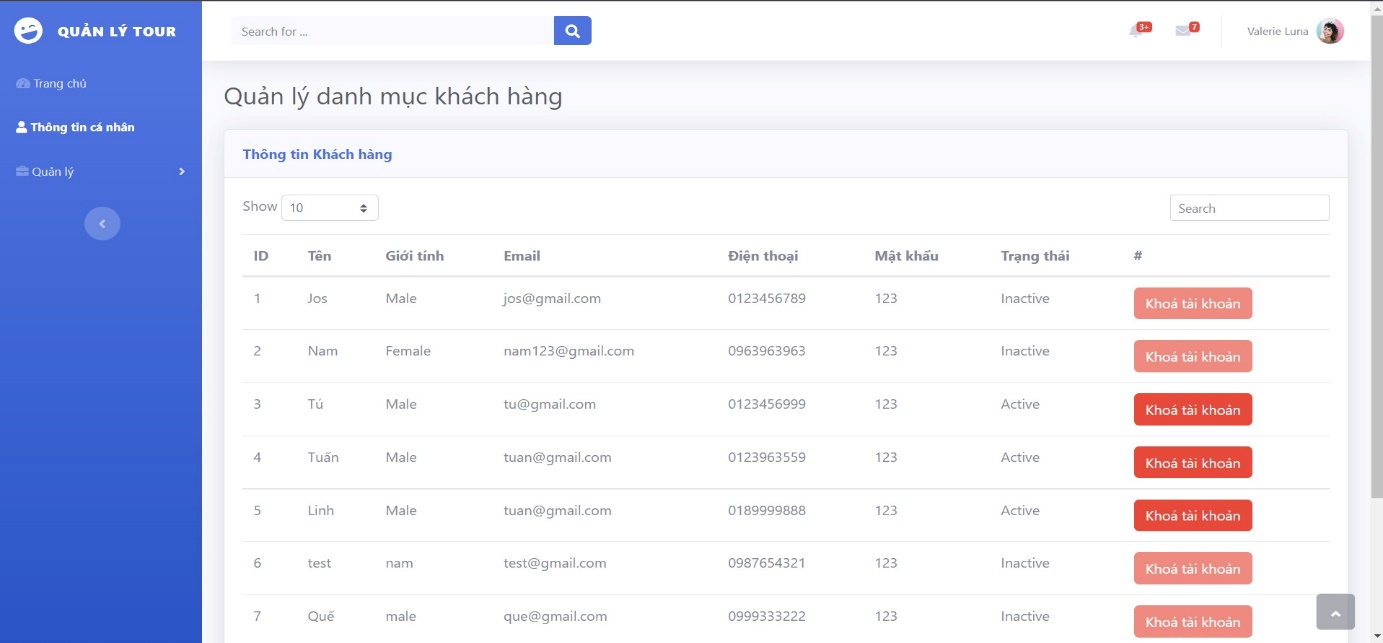


Hình 5.12: Giao diện quản lý nhân viên trang web quản lý

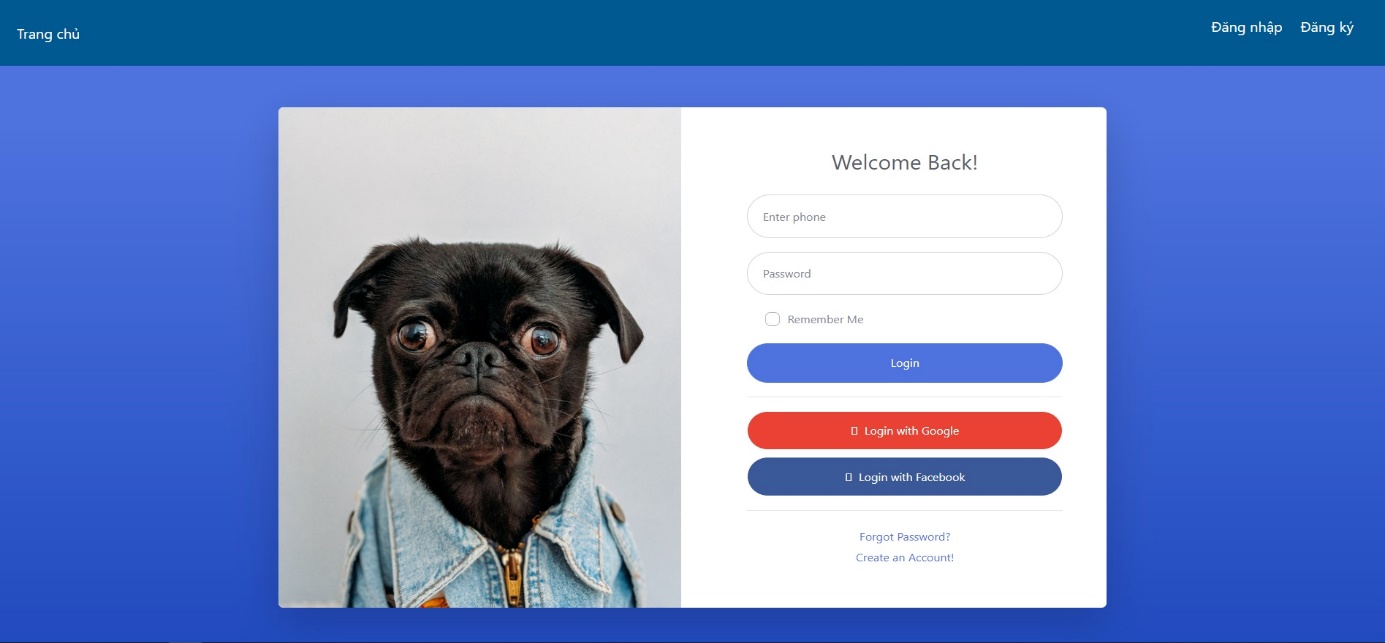
Hình 5.11: Giao diện quản lý dịch vụ -tour trang web quản lý

Hình 5.10: Giao diện quản lý phương tiện -tour trang web quản lý

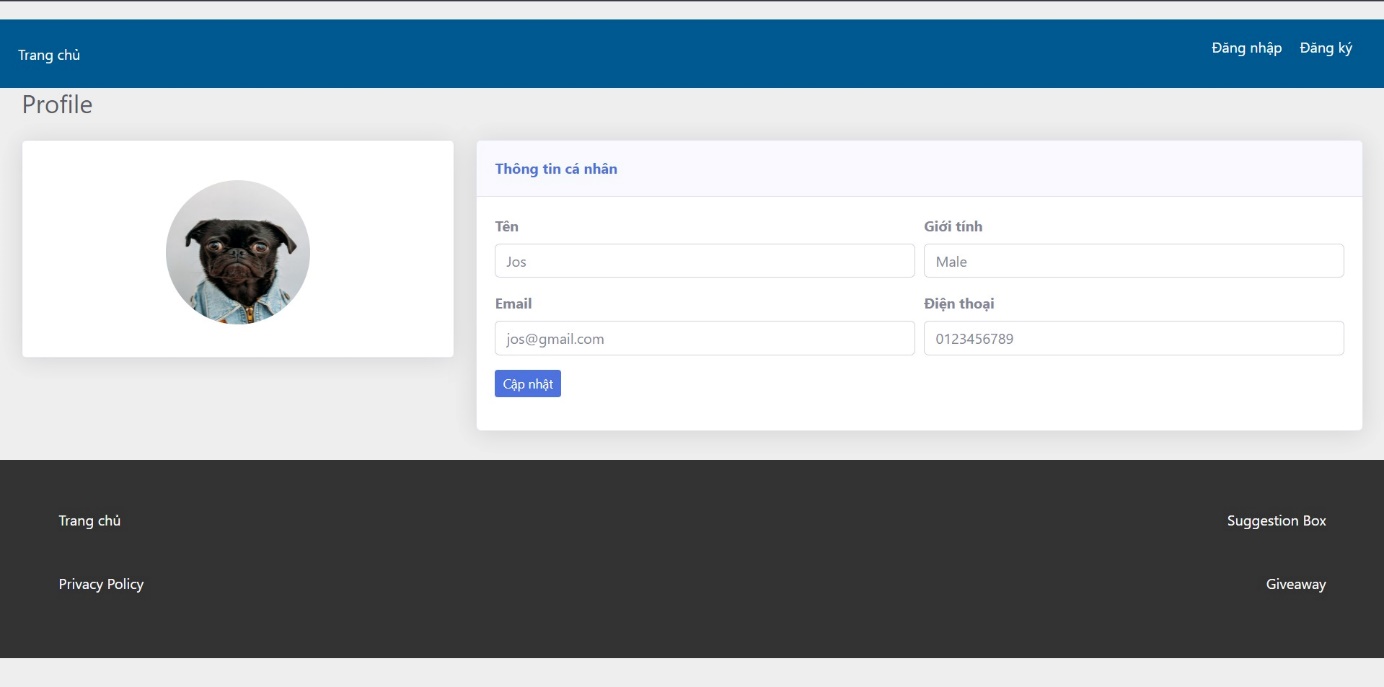




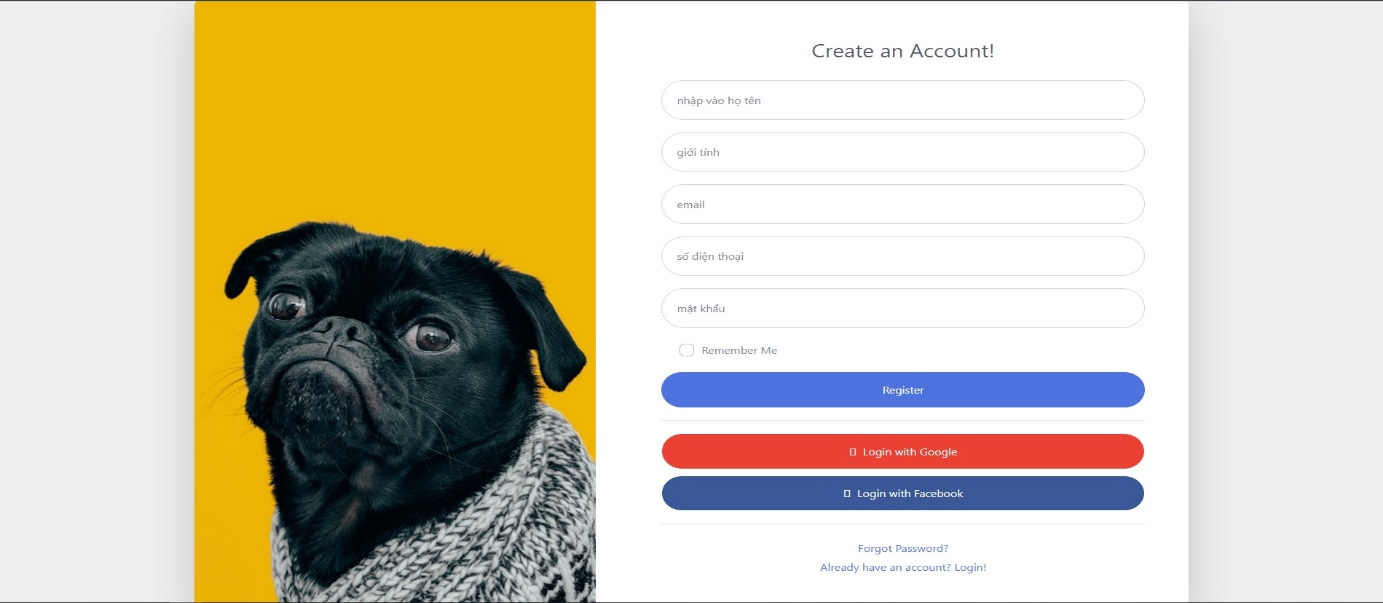
Hình 5.13: Giao diện quản lý Khách hàng trang web quản lý



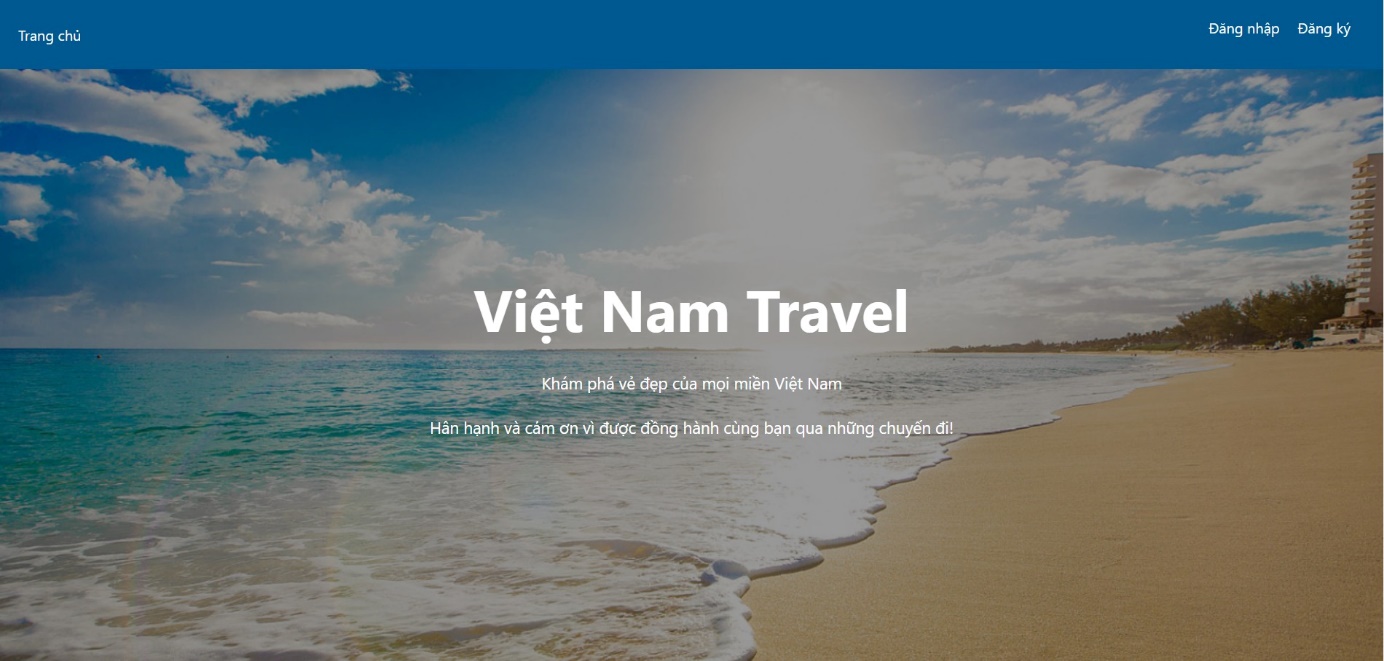
Hình 5.14: Giao diện đăng nhập trang web đặt tour

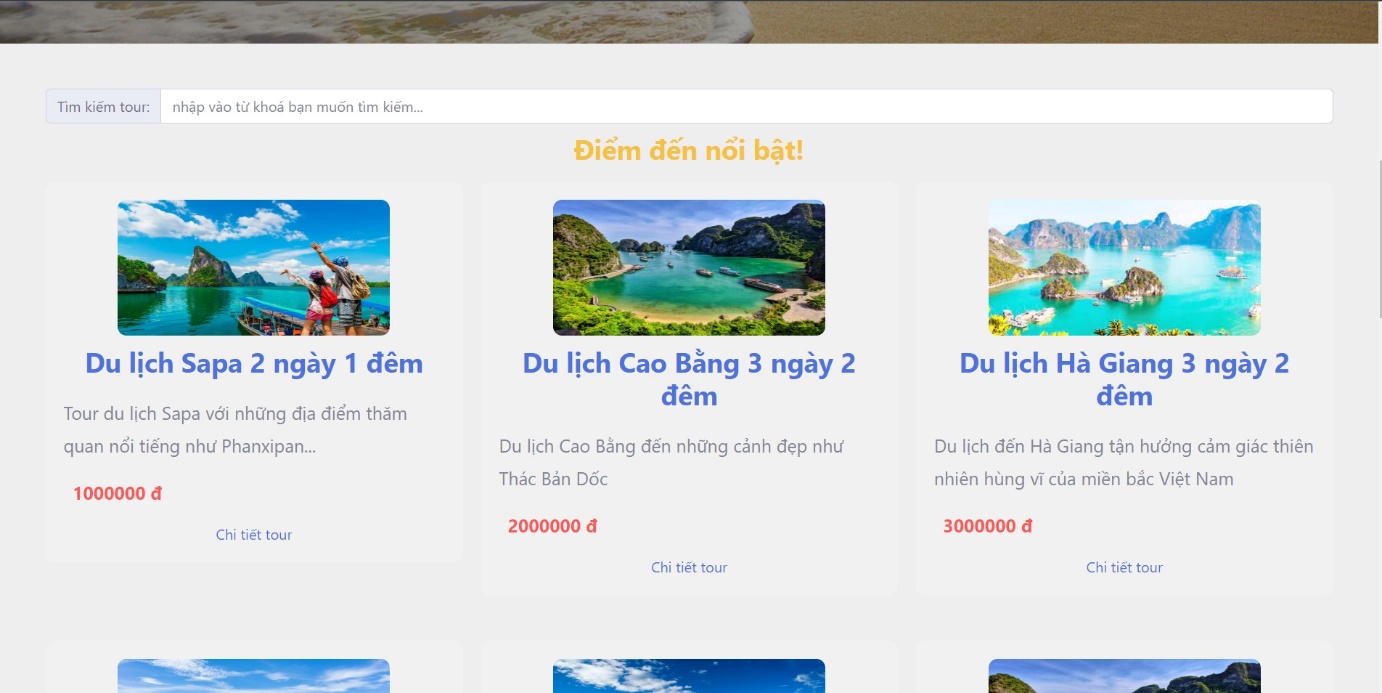


Hình 5.15: Giao diện thông tin cá nhân trang web đặt tour

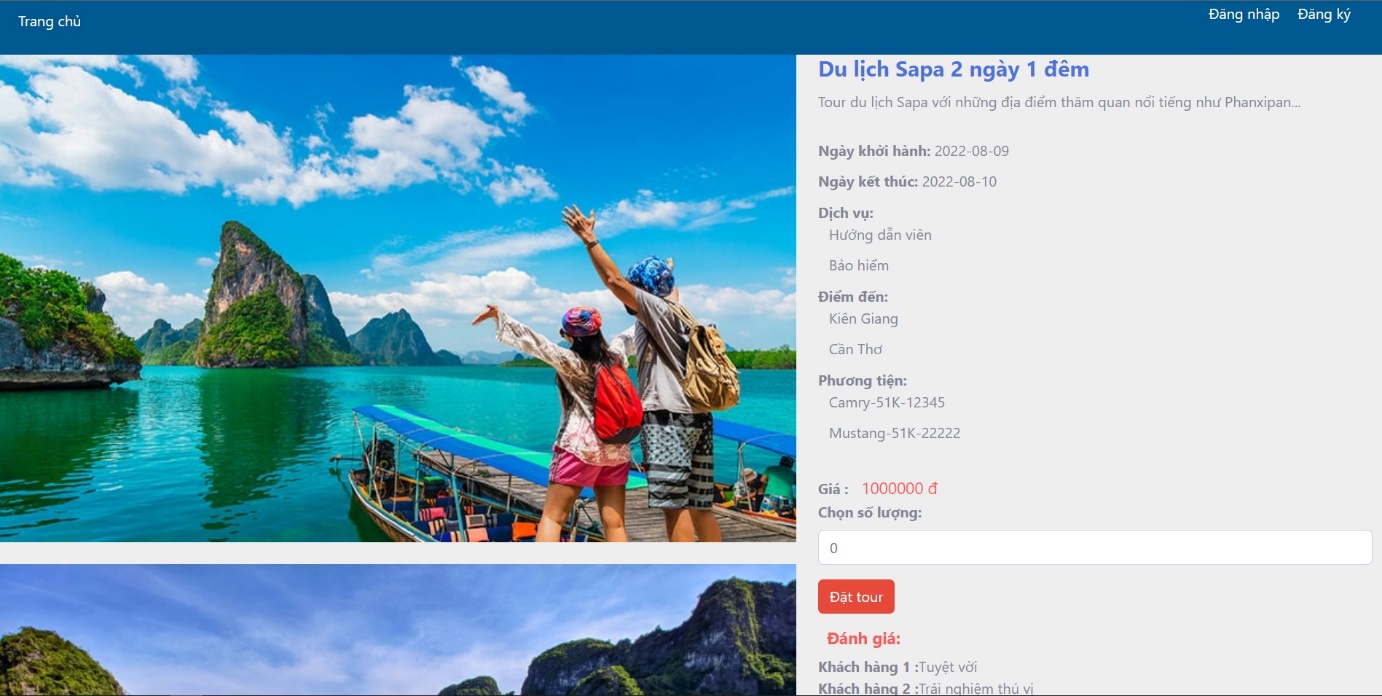
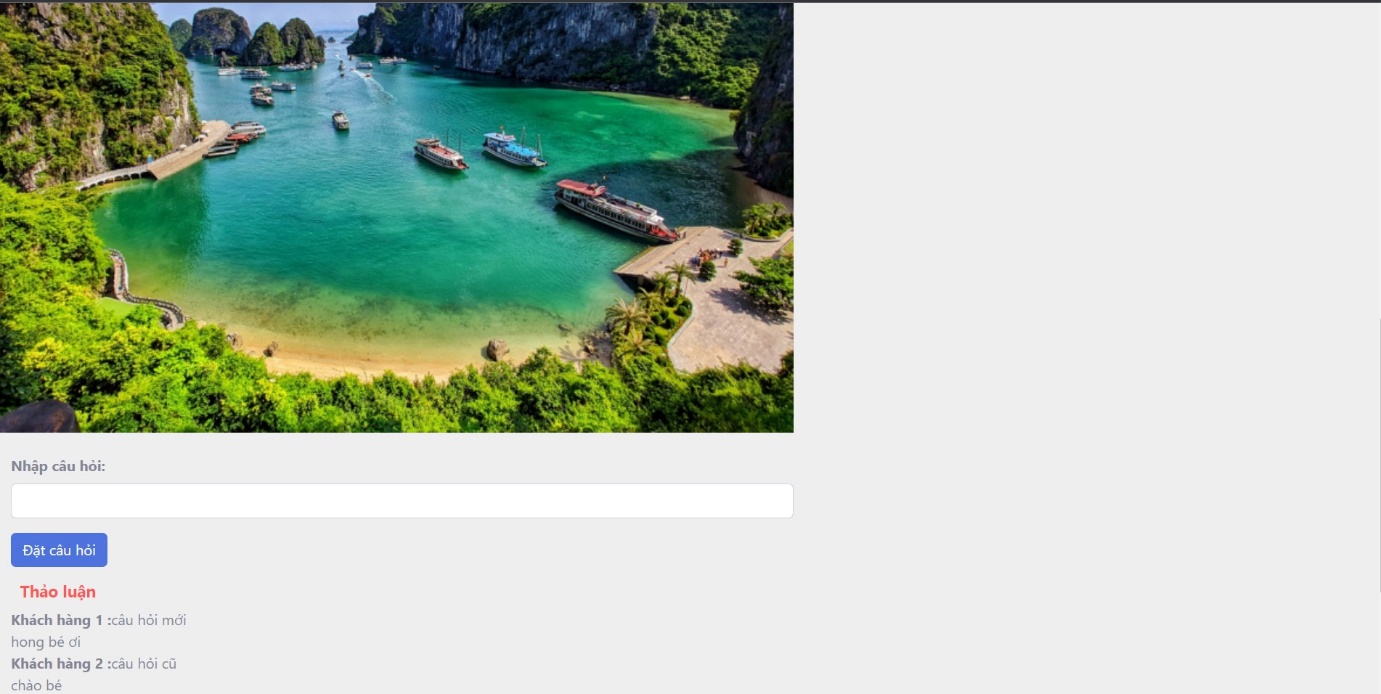


Hình 5.16: Giao diện đăng kí trang web đặt tour





Hình 5.17: Giao diện trang chủ trang web đặt tour



Hình 5.18: Giao diện chi tiết tour trang web đặt tour

# CHƯƠNG 6: KẾT LUẬN

## 6.1 Kết luận

### 6.1.1 Kết quả đạt được

* Tìm hiểu về Reactjs, Spring Boot, REST API để xây dựng trang web.
* Áp dụng xây dựng trang web quản lý đặt tour.
* Xây dựng được giao diện web site cho người quản trị và người dùng.
* Thực hiện được đa số các chức năng quản lý, người dùng cơ bản.

### 6.1.2 Hạn chế

* Giao diện chưa tối ưu.
* Còn thiếu nhiều chức năng nghiệp vụ.
* Quản lý hình ảnh còn nhiều hạn chế.
* Chưa có chức năng thống kê doanh thu.
* Phân quyền người dùng chưa đảm bảo kỹ thuật

## 6.2 Hướng phát triển

* Tiếp tục bổ sung các chức năng còn thiếu, sửa những chức năng đang bị lỗi.
* Tiếp tục phát triển các chức năng mới để có thể áp dụng rộng rãi cho nhiều đối tượng người dùng.

# CHƯƠNG 7: TÀI LIỆU THAM KHẢO

1. TOPDEV- Giới thiệu về Reactjs- <https://topdev.vn/blog/reactjs-nhung-dieu-ban-can-phai-biet/>
2. VIBLO- Bắt đầu với Reactjs- <https://viblo.asia/p/bat-dau-voi-reactjs-phan-1-3P0lPOE8Zox>
3. W3SCHOOL- Mysql tutorial- <https://www.w3schools.com/MySQL/default.asp>
4. Spring.io- Spring boot- <https://spring.io/projects/spring-boot>
5. Baeldung- Learn Spring boot- <https://www.baeldung.com/spring-boot>
6. Stack Overflow- fix bug- <https://stackoverflow.com/>