

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC CẦN THƠ  
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

-- ୧୦ • 📖 • ୧୩ --



**LUẬN VĂN TỐT NGHIỆP  
NGÀNH CÔNG NGHỆ THÔNG TIN  
CHUYÊN NGÀNH TIN HỌC ỨNG DỤNG**

**Đề tài  
PHÁT TRIỂN ỨNG DỤNG QUẢN LÝ TOUR DU LỊCH  
VỚI REACT NATIVE - PHẦN HỆ BACK-END**

**Sinh viên: Ngô Duy Nam**

**Mã số: B1809485**

**Khóa: 44**

**Cần Thơ, 12/2022**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC CẦN THƠ  
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG  
KHOA TRUYỀN THÔNG ĐA PHƯƠNG TIỆN**

-- ∞ • 📖 • ☸ --



**LUẬN VĂN TỐT NGHIỆP  
NGÀNH CÔNG NGHỆ THÔNG TIN  
CHUYÊN NGÀNH TIN HỌC ỨNG DỤNG**

**Đề tài  
PHÁT TRIỂN ỨNG DỤNG QUẢN LÝ TOUR DU LỊCH  
VỚI REACT NATIVE - PHẦN HỆ BACK-END**

Giảng Viên Hướng Dẫn  
**ThS. Hồ Văn Tú**

Sinh viên thực hiện  
**Ngô Duy Nam**  
**Mã số: B1809485**  
**Khóa: 44**

**Cần Thơ, 12/2022**

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG  
**KHOA TRUYỀN THÔNG ĐA PHƯƠNG TIỆN**

**XÁC NHẬN CHỈNH SỬA LUẬN VĂN  
THEO YÊU CẦU CỦA HỘI ĐỒNG**

Tên luận văn (tiếng Việt và tiếng Anh):

Phát triển ứng dụng quản lý Tour du lịch với React Native -Phân hệ Back-end

Developing a tourism management app using React Native- Module Back-end

Họ tên sinh viên: Ngô Duy Nam

MASV: B1809485

Mã lớp: DI18Y1A1

Đã báo cáo tại hội đồng ngành: Công nghệ thông tin

Ngày báo cáo: 14-12-2022

Hội đồng báo cáo gồm:

1. Ths. Huỳnh Phụng Toàn

Chủ tịch hội đồng

2. Ths. Võ Hải Đăng

Thành viên

3. Ths. Hồ Văn Tú

Thư ký

Luận văn đã được chỉnh sửa theo góp ý của Hội đồng.

Cần Thơ, ngày ..... tháng ..... năm 20..

**Giáo viên hướng dẫn**  
(Ký và ghi họ tên)

## LỜI CẢM ƠN

Luận văn này được hoàn thành tại Trường Công Nghệ Thông Tin & Truyền Thông – Trường Đại học Cần Thơ. Trong suốt thời gian thực hiện đề tài tôi đã nhận được rất nhiều sự giúp đỡ từ gia đình, thầy cô và bạn bè để có thể hoàn thành tốt luận văn lần này.

Lời đầu tiên tôi xin gửi lời cảm ơn chân thành đến gia đình, người thân, những người đã luôn ở bên cạnh tôi ủng hộ, giúp đỡ tôi có thời gian nghiên cứu đề tài và hết lòng hỗ trợ tôi trong suốt thời gian thực hiện đề tài.

Bên cạnh đó, tôi cũng xin gửi lời cảm ơn đến trường Đại học Cần Thơ, trường Công Nghệ Thông Tin & Truyền Thông, nơi đã tạo mọi điều kiện tốt nhất để tôi được học tập, rèn luyện, trau dồi bản thân. Tôi cũng xin gửi lời cảm ơn đến toàn thể quý thầy cô bộ môn cũng như quý thầy cô trong trường đã tận tình chỉ bảo, giảng dạy để tôi có những kiến thức vững chắc làm nền tảng hoàn thành thật tốt đề tài.

Với lòng biết ơn sâu sắc, tôi xin gửi lời cảm ơn đến thầy Hồ Văn Tú – Khoa Tin Học Ứng Dụng – trường Công Nghệ Thông Tin & Truyền Thông, người đã trực tiếp hướng dẫn, chỉ bảo và giúp đỡ tôi trong suốt thời gian thực hiện đề tài này.

Mặc dù đã hết sức cố gắng hoàn thành luận văn nhưng cũng không tránh khỏi những sai sót, kính mong nhận được cảm thông, chỉ bảo tận tình của Thầy. Cuối lời, tôi kính chúc Thầy nhiều sức khỏe, ngày càng thành công trong công việc.

Xin chân thành cảm ơn!

Cần Thơ, Ngày 05 Tháng 12 Năm 2022  
Sinh viên thực hiện

Ngô Duy Nam

**MỤC LỤC**

MỤC LỤC.....	i
DANH MỤC HÌNH .....	iii
DANH MỤC BẢNG.....	v
DANH MỤC KÝ HIỆU, CHỮ VIẾT TẮT .....	vi
TÓM TẮT .....	vii
ABSTRACT .....	viii
PHẦN 1: TỔNG QUAN .....	1
1. ĐẶT VẤN ĐỀ.....	1
2. TÓM TẮT LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ .....	1
3. MỤC TIÊU ĐỀ TÀI.....	2
4. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU .....	2
5. PHƯƠNG PHÁP NGUYÊN CỨU .....	2
5.1. Về lý thuyết.....	2
5.2. Về kỹ thuật .....	3
6. CƠ SỞ LÝ THUYẾT.....	3
6.1. JavaScript .....	3
6.2. Framework React Native .....	3
6.3. NodeJS và ExpressJS .....	3
6.4. Axios .....	4
6.5. Rest API .....	4
6.6. Mô hình MVC .....	5
6.7. Hệ quản trị cơ sở dữ liệu MongoDB.....	6
7. BỐ CỤC QUYỀN LUẬN VĂN .....	7
PHẦN 2: NỘI DUNG VÀ KẾT QUẢ THỰC HIỆN .....	9
1. MÔ TẢ TỔNG QUAN .....	9
1.1. Đặc tả hệ thống.....	9

1.2. Môi trường vận hành.....	9
1.3. Các yêu cầu chức năng.....	10
2. THIẾT KẾ VÀ CÀI ĐẶT.....	15
2.1. Tổng quan hệ thống.....	15
2.2. Mô hình phân rã tổng quát .....	17
2.3. Thiết kế dữ liệu .....	20
2.4. Cấu trúc dự án .....	24
3. KẾT QUẢ THỰC HIỆN.....	28
3.1. Chức năng đăng nhập hệ thống.....	28
3.2. Chức năng xem danh sách bài đăng của người dùng là khách hàng.....	29
3.3. Chức năng chi tiết bài đăng của người dùng là khách hàng .....	30
3.4. Chức năng thêm Tour mới của người dùng là khách hàng .....	30
3.5. Chức năng chỉnh sửa bài đăng của người dùng là khách hàng .....	32
3.6. Chức năng thống kê của người dùng là khách hàng .....	33
3.7. Chức năng quản lý nhân viên.....	34
3.8. Chức năng thêm một nhân viên mới .....	34
3.9. Chức năng quản lý khách hàng .....	35
3.10. Chức năng quản lý đối tác.....	35
PHẦN 3: KẾT LUẬN .....	36
1. KẾT QUẢ ĐẠT ĐƯỢC.....	36
2. CÁC ĐIỀU HẠN CHẾ.....	36
3. HƯỚNG PHÁT TRIỂN .....	36
TÀI LIỆU THAM KHẢO.....	38

**DANH MỤC HÌNH**

Hình 1.1. Mô hình Rest API.....	4
Hình 1.2. Mô hình MVC .....	5
Hình 1.3. Hệ quản trị cơ sở dữ liệu MongoDB .....	6
Hình 1.4. Phân cấp dữ liệu trong một document .....	7
Hình 2.1. Sơ đồ tổng quan hệ thống.....	15
Hình 2.2. Sơ đồ phân rã chức năng. ....	17
Hình 2.3. Sơ đồ use case Khách hàng .....	18
Hình 2.4. Sơ đồ use case Nhân viên.....	18
Hình 2.5. Sơ đồ use case Quản lý .....	19
Hình 2.6. Mô hình MLD .....	19
Hình 2.7. Cấu trúc thư mục server .....	24
Hình 2.8. Cấu trúc thư mục khởi tạo model.....	25
Hình 2.9. Schema Tour .....	25
Hình 2.10. Cấu trúc thư mục View.....	26
Hình 2.11. Giao diện đăng nhập.....	28
Hình 2.12. Giao diện xem danh sách bài đăng.....	29
Hình 2.13. Giao diện chi tiết bài đăng .....	30
Hình 2.14. Giao diện thêm bài đăng .....	30
Hình 2.15. Giao diện danh sách tour sau khi thêm .....	31
Hình 2.16. Giao diện chi tiết bài đăng sau khi thêm mới.....	31
Hình 2.17. Giao diện chỉnh sửa bài đăng.....	32
Hình 2.18. Giao diện chi tiết bài đăng sau khi chỉnh sửa .....	32
Hình 2.19. Giao diện phía di động sau khi chỉnh sửa .....	33
Hình 2.20. Giao diện thống kê .....	33
Hình 2.21. Chức năng quản lý nhân viên.....	34
Hình 2.22. Chức năng thêm một nhân viên mới .....	34

## DANH MỤC HÌNH

---

Hình 2.23. Chức năng quản lý khách hàng .....	35
Hình 2.24. Chức năng quản lý đối tác.....	35



**DANH MỤC BẢNG**

Bảng 1. Bảng chức năng đăng ký.....	10
Bảng 2. Bảng chức năng đăng nhập.....	10
Bảng 3. Bảng chức năng xem bài đăng của tour.....	11
Bảng 4. Bảng chức năng tìm kiếm tour.....	11
Bảng 5. Bảng chức năng tạo bài đăng tour .....	12
Bảng 6. Bảng chức năng xem trang cá nhân.....	12
Bảng 7. Bảng chức năng thích bài đăng.....	13
Bảng 8. Bảng chức năng thay đổi mật khẩu.....	13
Bảng 9. Bảng chức đặt mua tour .....	14
Bảng 10. Bảng chức năng xem lịch sử hóa đơn.....	14
Bảng 11. Bảng chức năng xem chi tiết bài đăng tour .....	15
Bảng 12. Collection Người dùng .....	20
Bảng 13. Collection tour .....	21
Bảng 14. Collection Loại Tour.....	22
Bảng 15. Collection Hóa đơn.....	23

**DANH MỤC KÝ HIỆU, CHỮ VIẾT TẮT**

<b>Ký hiệu/ chữ viết tắt</b>	<b>Diễn giải</b>
DOM	Document Object Model
API	Application Programming Interfaces
UI/UX	User Interface/User Experience
HTTP	Hypertext Transfer Protocol
URL	Uniform Resource Locator
HTML	Hyper Text Markup Language
REST API	Representational State Transfer Application Programming Interfaces
XML	Extensible Markup Language
JSON	JavaScript Object Notation
MVC	Model-View-Controller
CSDL	Cơ sở dữ liệu

### TÓM TẮT

Đề tài “Phát triển ứng dụng quản lý Tour du lịch với React Native - Phần hệ Back-end” được xây dựng nhằm phục vụ khách hàng có nhu cầu tìm kiếm các gói tour du lịch, các địa điểm tham quan dễ dàng và nhanh chóng. Với các đối tượng sử dụng là: khách hàng có đăng ký tài khoản, đối tác, nhân viên và quản lý. Để tiện lợi khi đi du lịch thì khách hàng đã đăng kí sẽ được mua vé tour du lịch tìm kiếm chi tiết địa điểm cần đến, lưu trữ các điểm du lịch mà mình yêu thích và quản lý tài khoản cá nhân. Bên cạnh đó, tôi đã xây dựng website quản trị để người quản lý dễ dàng quản lý thông tin của khách hàng cũng như đối tác, danh mục các tour du lịch, hóa đơn mua bán, doanh thu, xem báo cáo thống kê khi đăng nhập bằng tài khoản có quyền quản lí. Về công nghệ phát triển thì tôi nhận thấy React Native có thể ứng dụng được tốt nhất đối với tôi, tuyệt vời hơn khi sử dụng cũng với NodeJS – một môi trường runtime chạy JavaScript đa nền tảng có mã nguồn mở, và hệ quản trị cơ sở dữ liệu NoSQL – MongoDB.

Nội dung đề tài gồm ba phần:

**Phần giới thiệu:** Đặt vấn đề, mục tiêu đề tài, phạm vi nghiên cứu, định nghĩa các khái niệm, phương pháp.

**Phần nội dung:** Đưa ra giải pháp thực hiện đề tài, đặc tả hệ thống, vẽ các mô hình, sơ đồ chức năng, lưu đồ giải quyết vấn đề.

**Phần kết luận:** Thực hiện việc nhận xét kết quả đạt được, nêu lên ưu điểm cũng như nhược điểm và hướng phát triển của đề tài.

**Từ khóa:** MongoDB, NodeJS, React Native, du lịch.

### ABSTRACT

The thesis "Developing a tourism management app using React Native - Module Front-end" was built to serve customers who need to find tour packages, sightseeing easily and quickly. With the user objects are: customers with registered accounts, partners, and managers. For convenience when traveling registered customers will be able to buy tour tickets search for details of places to go, save the favorite tourist destinations and manage personal account. In addition, I have built an admin website for managers to easily manage information of customers as well as partners, list of tours, purchase and sale invoices, revenue, view statistical reports when logging in with an account with management authorize. In terms of development technology, I find React Native to be the best framework for me, even better when used with Node JS - a cross-platform JavaScript runtime environment with open source code, and an administration system NoSQL database - MongoDB.

The content of topic consists of three parts:

**Introduction:** Give the problem, objectives of the topic and research scope.

**Content:** Defining concepts, methods, providing solutions to implement the topic, specifying the system, drawing models, functional diagrams, pseudo-problem flowcharts.

**Conclusion:** Reviewing the results achieved, giving the advantages as well as disadvantages and development direction of the topic.

**Keywords:** Travel, MongoDB, NodeJS, React Native.

### PHẦN 1: TỔNG QUAN

#### 1. ĐẶT VẤN ĐỀ

Với công nghệ được phát triển mạnh mẽ như hiện nay, nhu cầu sử dụng các thiết bị tiện ích ngày càng tăng cao, điện thoại di động đang ngày càng phổ biến hơn nhờ sự tiện lợi, linh hoạt, không bị giới hạn bởi không gian và thời gian. Làm cho chất lượng cuộc sống ngày càng được nâng cao, con người càng có nhiều nhu cầu hơn về đời sống, kết nối cộng đồng và đặc biệt là nhu cầu về du lịch và nghỉ dưỡng. Do đó thị trường hoạt động du lịch đã có nhiều sự thay đổi vượt bậc. Vào các ngày lễ tết các ứng dụng di động đóng vai trò quan trọng trong việc kích thích chi tiêu của người dùng. Sự tăng trưởng này chủ yếu được thúc đẩy bởi số lượng điện thoại di động đang ngày càng tăng cao. Các ứng dụng thương mại điện tử lần lượt ra đời và ngày càng hoạt động mạnh mẽ.

Trong lĩnh vực du lịch, nó đã trở thành xu hướng chung của toàn cầu, chiếm vị trí chiến lược trong nền kinh tế quốc gia nhiều nước trên thế giới. Cùng với sự phát triển mạnh mẽ của công nghệ thông tin và sự bùng nổ Internet, hoạt động thương mại điện tử áp dụng trong du lịch đã phát triển nhanh chóng, mang lại giá trị gia tăng cao hơn cho ngành du lịch. Bên cạnh đó vấn đề đặt ra là người kinh doanh làm thế nào để quảng bá các chuyến du lịch vừa tiếp cận khách hàng dễ dàng mà không tốn quá nhiều chi phí, còn đối với người dùng thì làm thế nào để dễ dàng tìm được một công ty uy tín để tham khảo các tour du lịch mà không cần tốn quá nhiều công sức, thời gian cho việc di chuyển và tìm kiếm. Song song với tốc độ phát triển ngành du lịch như hiện nay, thì những tiện ích cũng như dịch vụ của để phục vụ cho nhu cầu của du khách cũng phải phát triển.

Chính vì những lẽ đó, chúng tôi quyết định thực hiện đề tài “Phát triển ứng dụng quản lý Tour du lịch với React Native - Phân hệ Back-end” để bám theo hướng phát triển của xu thế hiện đại trong thời buổi công nghiệp 4.0 như hiện nay. Giúp cho du khách có thể chủ động hơn trong việc đặt và tìm kiếm tour du lịch phù hợp với nhu cầu, và ý muốn của du khách nhất.

#### 2. TÓM TẮT LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ

Hiện nay, đã có nhiều ứng dụng đặt tour du lịch. Các ứng dụng đó đã tạo nên nhiều lợi thế về công nghệ, giúp khách hàng có thể nhận thông báo tự động về lịch trình thông qua điện thoại hoặc e-mail, thanh toán trực tuyến thông qua các tài khoản ngân hàng, lên lịch hoặc hủy tour một cách chủ động, giúp khách hàng có thể đặt

được nhiều gói tour cùng một lúc. Song với những tiện ích trên, việc tạo ra giao diện thân thiện với người dùng thì các ứng dụng trên còn gặp nhiều hạn chế, làm cho khách hàng khó đi đến mục đích mà họ cần.

### 3. MỤC TIÊU ĐỀ TÀI

Đề tài nhằm tạo ra một ứng dụng phục vụ nhu cầu thông tin và đặt tour du lịch của du khách. Giải quyết được những khó khăn giữa doanh nghiệp và khách hàng đối với hình thức đặt tour truyền thống gặp phải. Giúp khách hàng thuận tiện hơn khi đặt tour, doanh nghiệp có thể chủ động trong việc tạo ra tour mới. Doanh nghiệp dễ dàng quảng cáo tour mới đến với khách hàng. Từ đó tiết kiệm được nhiều thời gian, chi phí của doanh nghiệp cũng như khách hàng.

### 4. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

Đề tài tập trung nghiên cứu về đối tượng chính là những khách hàng tiềm năng về du lịch, các doanh nghiệp đang kinh doanh loại hình dịch vụ du lịch. Tập trung nghiên cứu về cách thức tìm kiếm tour du lịch của khách hàng có nhu cầu, và doanh nghiệp quảng bá các gói du lịch đến khách hàng như thế nào? Từ đó, giải quyết vấn đề đặt tour và thanh toán tour của du khách. Tạo ra giao diện thân thiện, dễ sử dụng cho khách hàng, cũng như dễ dàng quản lý cho người quản trị.

### 5. PHƯƠNG PHÁP NGUYÊN CỨU

Dựa trên những tài liệu và nghiên cứu liên quan sẵn có. Từ đó, tổng hợp, đề xuất ra mô hình, giải pháp phù hợp với thực tiễn. Tìm hiểu, nắm rõ về mặt lý thuyết cũng như cách hoạt động của framework React Native, NodeJS, Mongoose, ExpressJS, JavaScript. Sau đó, xây dựng ứng dụng đặt tour du lịch phù hợp với khách hàng và doanh nghiệp.

#### 5.1. Về lý thuyết

Tìm hiểu về các loại hình du lịch theo tour, và cách vận hành, giao tiếp giữa doanh nghiệp và khách hàng để đưa ra mô hình phù hợp nhằm cho ra đời một ứng dụng tốt nhất có thể.

Tìm hiểu ngôn ngữ Javascript, framework React Native, NodeJS, Mongoose, ExpressJS và các thư viện hỗ trợ xây dựng giao diện cho ứng dụng.

Tìm hiểu mô hình client server, Redux để hỗ trợ cho việc truy cập, lưu trữ dữ liệu, giúp ứng dụng mượt mà hơn.

### 5.2. Về kỹ thuật

Sử dụng ngôn ngữ Javascript cùng với sự hỗ trợ của framework React Native để xây dựng giao diện và xử lý các chức năng của ứng dụng.

Dùng thư viện Mongoose, NodeJS, framework ExpressJS để xây dựng server lưu trữ và xử lý dữ liệu từ client.

## 6. CƠ SỞ LÝ THUYẾT

### 6.1. JavaScript

JavaScript [8] là một ngôn ngữ kịch bản đa nền tảng, hướng đối tượng được sử dụng để làm cho các trang web tương tác (ví dụ: có hình ảnh động phức tạp, nút có thể nhấp, menu bật lên, v.v.). Ngoài ra còn có các phiên bản phía máy chủ nâng cao hơn của JavaScript như Nodejs, cho phép bạn thêm nhiều chức năng hơn vào trang web hơn là tải xuống các tệp (như hợp tác thời gian thực giữa nhiều máy tính). Bên trong môi trường máy chủ (ví dụ: trình duyệt web), JavaScript có thể được kết nối với các đối tượng của môi trường để cung cấp kiểm soát chương trình cho chúng.

### 6.2. Framework React Native

React Native [3] là một khung JavaScript để viết các ứng dụng di động thực, tự hiển thị cho IOS và Android. Nó dựa trên React, thư viện JavaScript của Facebook để xây dựng giao diện người dùng, nhưng thay vì nhắm mục tiêu trình duyệt, nó nhắm mục tiêu các nền tảng di động.

React Native không thao tác với DOM thông qua DOM ảo, mà React Native thực hiện quá trình xử lý nền trực tiếp ngay trên thiết bị đầu cuối cũng như giao tiếp cùng với nền tảng góp phần qua một cầu trung gian, theo đợt và bất đồng bộ.

Những thành phần React sẽ bao bọc mã gốc cũng như tương tác cùng với API gốc thông qua mô hình Javascript và UI khai báo của React. Chính điều này đã giúp cho quá trình phát triển ứng dụng dành cho nhiều nền tảng trở nên nhanh hơn.

### 6.3. NodeJS và ExpressJS

#### 6.3.1. NodeJS

Node.js [6] là một platform (nền tảng) độc lập được xây dựng trên môi trường Javascript Runtime. Các ứng dụng mạng được xây dựng dựa trên Node.js có thể thực hiện nhanh chóng và dễ dàng mở rộng về sau. Nodejs được thiết kế bởi Ryan Lienhart

Dahl - một kỹ sư phần mềm người Mỹ vào năm 2009 và phát triển dưới sự bảo trợ của Joyent.

Phần Core bên dưới của Nodejs được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao. Các ứng dụng được tạo ra bởi Node.js có tốc độ xử lý nhanh, realtime thời gian thực. Node.js thích hợp cho các sản phẩm có lưu lượng truy cập lớn, cần sự thay đổi về công nghệ và cần được mở rộng hoặc tạo ra các dự án Startup một cách nhanh nhất có thể.

### 6.3.2. ExpressJS

Expressjs [4] là một framework được xây dựng trên nền tảng của Nodejs. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. Expressjs hỗ trợ các method HTTP và middleware tạo ra API vô cùng mạnh mẽ và dễ sử dụng.

Một số chức năng chính của Expressjs như sau:

- Thiết lập các lớp trung gian để trả về các HTTP request.
- Define router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.
- Cho phép trả về các trang HTML dựa vào các tham số.

### 6.4. Axios

Axios [5] là một thư viện HTTP Client dựa trên Promise dành cho nodejs và trình duyệt. Nó có tính đẳng hình (tức là cùng codebase có thể chạy trong cả trình duyệt và node.js). Ở phía server thì nó sử dụng native module http trong node.js, còn ở phía client (trình duyệt) thì nó sử dụng XMLHttpRequest.

### 6.5. Rest API



**Hình 1.1. Mô hình Rest API**

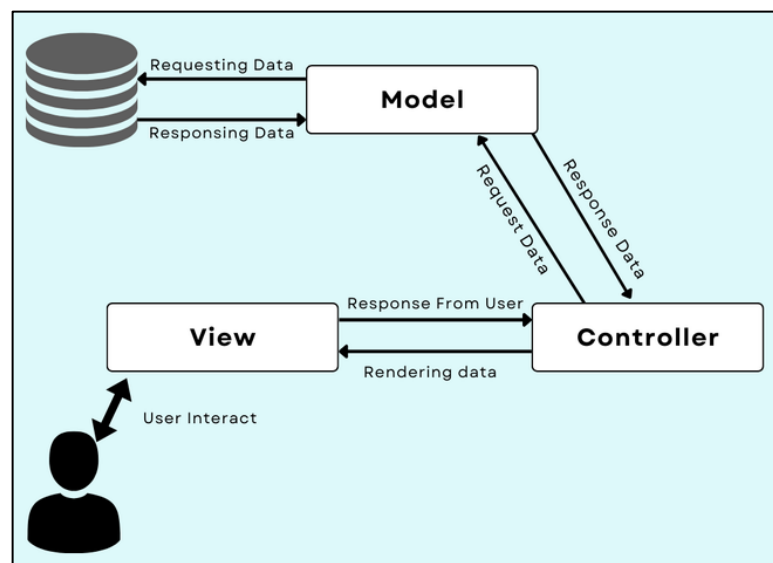
REST API [7] là một phong cách kiến trúc xác định một tập hợp các ràng buộc được sử dụng để tạo các dịch vụ web. REST API là một cách truy cập các dịch vụ web một cách đơn giản và linh hoạt mà không có bất kỳ xử lý nào.



Một yêu cầu được gửi từ máy khách đến máy chủ dưới dạng URL web dưới dạng HTTP GET hoặc POST hoặc PUT hoặc DELETE. Sau đó, một phản hồi trở lại từ máy chủ dưới dạng tài nguyên có thể là bất cứ thứ gì như HTML, XML, Image hoặc JSON.

Trong HTTP, có năm phương pháp thường được sử dụng trong kiến trúc dựa trên REST, tức là POST, GET, PUT, PATCH và DELETE. Chúng tương ứng để tạo, đọc, cập nhật và xóa tương ứng. Có những phương pháp khác ít được sử dụng như OPTIONS và HEAD.

### 6.6. Mô hình MVC



**Hình 1.2. Mô hình MVC**

Bộ điều khiển mô hình-View (MVC) [2] là một mẫu kiến trúc phân tách một ứng dụng thành ba thành phần logic chính: mô hình, chế độ xem và bộ điều khiển. Mỗi thành phần này được xây dựng để xử lý các khía cạnh phát triển cụ thể của một ứng dụng. MVC là một trong những khung phát triển web tiêu chuẩn công nghiệp thường xuyên nhất để tạo ra các dự án có thể mở rộng.

### 6.7. Hệ quản trị cơ sở dữ liệu MongoDB



**Hình 1.3. Hệ quản trị cơ sở dữ liệu MongoDB**

MongoDB [1] là một hệ quản trị cơ sở dữ liệu được thiết kế cho các ứng dụng web và hạ tầng Internet. Mô hình dữ liệu và chiến lược hoạt động được xây dựng phục vụ cho các ứng dụng có thông lượng đọc và ghi dữ liệu cao, khả năng mở rộng quy mô lưu trữ dữ liệu dễ dàng thông qua cơ chế phân mảnh linh động cùng với hỗ trợ tiện ích sao lưu dự phòng tự động.

Không phụ thuộc và số nút (node) lưu trữ dữ liệu của ứng dụng, MongoDB luôn đảm bảo một hiệu năng sử dụng khá cao cho ứng dụng. MongoDB có chiến lược phân tán dữ liệu hiệu quả và bảo trì đơn giản, mô hình dữ liệu khá trực quan – mô hình dữ liệu tài liệu (document based – data model).

MongoDB là một hệ quản trị cơ sở dữ liệu mạnh mẽ dựa trên mô hình dữ liệu hướng tài liệu biểu diễn cấu trúc dữ liệu đa dạng, lưu trữ tập dữ liệu có kích thước lớn, linh hoạt và dễ dàng mở rộng, nó là sự kết hợp khả năng mở rộng theo phương pháp scaling out linh hoạt cùng khả năng nhân bản đảm bảo tính sẵn có và bền vững của dữ liệu.

MongoDB dựa trên mô hình dữ liệu thân thiện có cấu trúc linh hoạt thuận lợi cho biểu diễn các mối quan hệ bên ngoài thực tế như hỗ trợ kiểu dữ liệu nhúng mẫu tin, kiểu mảng mẫu tin. Hỗ trợ các thao tác quản trị dễ dàng như thiết lập các cơ chế sao lưu dự phòng, thêm khả năng lưu trữ cho hệ thống. MongoDB hỗ trợ APIs linh hoạt giúp người phát triển tập trung vào phát triển chức năng của ứng dụng nhiều hơn là vấn đề lưu trữ.

Các khái niệm trong MongoDB:

- Mẫu tin – Document: là đơn vị dữ liệu nhỏ nhất của MongoDB, tương đương với khái niệm "dòng - row" hay "mẫu tin" trong cơ sở dữ liệu quan hệ: một document gồm tập hợp thứ tự các khóa đi kèm với các giá trị hay còn gọi là cặp khóa - giá trị (key/value).

- Bảng – Collection: một tập hợp các document, document tương đương với khái niệm row còn collection tương đương với khái niệm "bảng - table" trong cơ sở dữ liệu quan hệ.
- Schema-Free: Collection trong MongoDB với đặc tính nổi bật là SchemaFree, khác với bảng trong cơ sở dữ liệu quan hệ, mỗi mẫu tin trong cùng một bảng (quan hệ) phải có số trường giống nhau và kiểu dữ liệu ở các trường giữa các mẫu tin phải tương đồng. Ngược lại, các document trong cùng một collection sẽ có hình dạng khác nhau – mỗi document có thể có cấu trúc khác nhau.

```
_id: ObjectId('638b270911b2edc75a597df0')
HoTen: "Phan Hải Dương"
NgàySinh: "12/3/2000"
SDT: "0325963852"
✓ ĐịaChí: Array
  ✓ 0: Object
    TenĐịaChí: "Home"
    TỉnhTP: "Bạc Liêu"
    QuanHuyen: "Giá Rai"
    XãPhuong: "Phước Long"
    ChiTiet: "Phước Long, Giá Rai, Bạc Liêu"
    _id: ObjectId('638b270911b2edc75a597df1')
    GiớiTinh: "NAM"
    Email: "duong@gmail.com"
    CMND: "335678419"
    MatKhau: "$2b$10$0AeYooc4Af1m3zYaYmZz3uCRgrLfU1hm56SHD6iBEL/zAd39xGwtC"
  ✓ YêuThích: Array
  ✓ LịchSu: Array
    Quyen: "BAN"
    createdAt: 2022-12-03T10:38:01.590+00:00
    updatedAt: 2022-12-05T04:26:36.930+00:00
    __v: 0
```

**Hình 1.4. Phân cấp dữ liệu trong một document**

## 7. BỐ CỤC QUYỀN LUẬN VĂN

Ngoài các phần mục lục, danh mục hình, danh mục bảng, danh mục các ký hiệu và chữ viết tắt, tóm tắt, thì quyền luận văn còn được chia thành các phần như sau:

- Phần 1. Tổng quan: Trong phần này, giới thiệu về vấn đề, bối cảnh hiện nay để hình thành nên bài toán, các đối tượng và phạm vi nghiên cứu để giải quyết bài toán. Giới thiệu thông tin cơ bản về các công nghệ được sử dụng trong dự án.

- Phần 2. Nội dung và kết quả thực hiện: Trình bày các sơ đồ, mô hình của dự án. Quá trình thực hiện dự án, đồng thời chỉ ra các giải pháp, công nghệ được sử dụng. Cuối cùng phân tích, minh chứng các chức năng của hệ thống.
- Phần 3: Kết luận: Đánh giá lại quá trình thực hiện dự án, cho thấy kết quả đạt được trong suốt quá trình thực hiện dự án, từ đó, đưa ra hướng phát triển dự án trong tương lai.

### PHẦN 2: NỘI DUNG VÀ KẾT QUẢ THỰC HIỆN

#### 1. MÔ TẢ TỔNG QUAN

##### 1.1. Đặc tả hệ thống

Bài toán xây dựng ứng dụng quản lý tour du lịch với nhiều tính năng nổi bật sẽ tạo ra một không gian vô cùng tiện ích cho người du lịch và người quản lý. Các đối tác là những nhà cung cấp các dịch vụ du lịch, tham quan, nghỉ dưỡng...

Nhân viên muốn đăng nhập vào hệ thống cần có tài khoản, mỗi nhân viên sẽ được cấp một mã số nhân viên duy nhất và các thông tin gồm: họ, tên, địa chỉ, số điện thoại, email, giới tính, ngày sinh và mỗi nhân viên sẽ có chức vụ khác nhau.

Khi khách hàng muốn đăng nhập vào hệ thống cần có tài khoản, mỗi khách hàng sẽ có một tài khoản và có các thông tin như: họ, tên, địa chỉ, số điện thoại, email, giới tính và ngày sinh. Sau khi khách hàng đăng nhập vào hệ thống sẽ đến trang chủ, khách hàng có thể xem tin tức, tìm kiếm tên tour, xem chi tiết, xem lịch sử đặt, đặt tour. Khi khách hàng muốn đặt tour sẽ điền thông tin cần thiết như họ tên, số lượng, ngày khởi hành.

##### 1.2. Môi trường vận hành

- Máy chủ CSDL:
  - Hệ điều hành: Window.
  - Cơ sở dữ liệu: MongoDB.
  - Server: NodeJS.
- Client:
  - Thiết bị di động có kết nối wifi hoặc 3G/4G.

### 1.3. Các yêu cầu chức năng

#### 1.3.1. Chức năng đăng ký

**Bảng 1. Bảng chức năng đăng ký**

Mã yêu cầu	01
Tên chức năng	Đăng ký tài khoản
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng khách
Mô tả tóm tắt	Chức năng cho phép người dùng đăng ký tài khoản để sử dụng các chức năng của ứng dụng
Tiền điều kiện	Ứng dụng đang hoạt động
Cách xử lý	Bước 1: Hiện thị giao diện đăng ký. Người dùng bắt đầu nhập thông tin đăng ký. Bước 2: Hệ thống kiểm tra tính hợp lệ và đầy đủ của thông tin người dùng nhập vào. Nếu thông tin không hợp lệ hiện thị thông báo đăng ký sai, ngược lại chuyển đến bước 3. Bước 3: Hệ thống chuyển đến giao diện ứng dụng cung cấp các chức năng mà người dùng có thể sử dụng khi đăng ký thành công. Bước 4: Kết thúc sự kiện.
Kết quả	Hiện thị thông báo đăng ký thành công , ngược lại báo lỗi và yêu cầu nhập lại.
Ghi chú	Đăng ký tài khoản bằng email

#### 1.3.2. Chức năng đăng nhập

**Bảng 2. Bảng chức năng đăng nhập**

Mã yêu cầu	02
Tên chức năng	Đăng nhập tài khoản
Mức ưu tiên	
Đối tượng sử dụng	Người dùng, nhân viên
Mô tả tóm tắt	Chức năng cho phép người dùng đăng nhập bằng tài khoản để sử dụng các chức năng của app
Tiền điều kiện	Phải có tài khoản đăng nhập
Các xử lý	Bước 1: Truy cập vào trang chủ, nếu chưa đăng nhập sẽ được chuyển hướng về trang đăng nhập Bước 2: Nhập thông tin đăng nhập sau đó nhấn vào nút “Đăng nhập”. Hệ thống sẽ kiểm tra thông tin đăng nhập, nếu đúng sẽ chuyển hướng về trang chủ, nếu sai sẽ hiện thị lỗi.
Kết quả	Chuyển hướng về trang chủ nếu đăng nhập thành công, hoặc hiện thị lỗi nếu thông tin đăng nhập không chính xác.

Ghi chú	Đăng nhập bằng email và mật khẩu.
---------	-----------------------------------

### 1.3.3. Chức năng xem bài đăng của tour

**Bảng 3. Bảng chức năng xem bài đăng của tour**

Mã yêu cầu	03
Tên chức năng	Xem bài đăng
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng đã có tài khoản
Mô tả tóm tắt	Chức năng cho phép người dùng đăng ký tài khoản để sử dụng các chức năng của ứng dụng.
Tiền điều kiện	Đã đăng nhập tài khoản.
Cách xử lý	Bước 1: Đăng nhập thành công vào ứng dụng.
Kết quả	Hiển thị các bài đăng tour lên giao diện.
Ghi chú	

### 1.3.4. Chức năng tìm kiếm tour

**Bảng 4. Bảng chức năng tìm kiếm tour**

Mã yêu cầu	04
Tên chức năng	Tìm kiếm tour
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng đã có tài khoản
Mô tả tóm tắt	Chức năng này cho phép người dùng tìm kiếm tour bằng tên bài đăng
Tiền điều kiện	Đã đăng nhập tài khoản
Cách xử lý	Bước 1: Hệ thống hiển thị giao diện tìm kiếm để người dùng tìm kiếm. Bước 2: Hệ thống tiến hành tìm kiếm các bài đăng có chứa từ khóa mà người dùng đã nhập, sau đó hiển thị lên giao diện. Bước 3: Người dùng chọn một mục trong danh sách kết quả để chuyển đến giao diện tour chi tiết.
Kết quả	Hệ thống hiển thị các kết quả tìm kiếm lên giao diện.
Ghi chú	

### 1.3.5. Chức năng tạo bài đăng tour

**Bảng 5. Bảng chức năng tạo bài đăng tour**

Mã yêu cầu	05
Tên chức năng	Tạo bài đăng
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng có tài khoản
Mô tả tóm tắt	Chức năng này cho phép tạo bài đăng tour lên hệ thống
Tiền điều kiện	Đã đăng nhập tài khoản
Cách xử lý	Bước 1: người dùng chọn “Tạo bài đăng” trên giao diện. Bước 2: Hệ thống hiển thị giao diện để người dùng tạo bài đăng. Bước 3: Người dùng tiến hành điền thông tin và lựa chọn hình ảnh. Bước 4: Người dùng nhấn vào nút “Đăng bài” để tạo bài đăng.
Kết quả	Bài viết mới được đăng lên ứng dụng.
Ghi chú	

### 1.3.6. Chức năng xem trang cá nhân

**Bảng 6. Bảng chức năng xem trang cá nhân**

Mã yêu cầu	06
Tên chức năng	Xem thông tin cá nhân
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng đã có tài khoản
Mô tả tóm tắt	Người dùng chọn vào “thông tin cá nhân” để xem thông tin cá nhân.
Tiền điều kiện	Đã đăng nhập tài khoản
Cách xử lý	Bước 1: Người dùng chọn “” của người dùng Bước 2: Ứng dụng hiển thị giao diện cá nhân của người dùng.
Kết quả	Hệ thống hiển thị thông tin cá nhân của người dùng lên giao diện.
Ghi chú	



### 1.3.7. Chức năng thích bài đăng

**Bảng 7. Bảng chức năng thích bài đăng**

Mã yêu cầu	07
Tên chức năng	Thích bài đăng
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng đã có tài khoản
Mô tả tóm tắt	Chức năng này cho phép người dùng nhấn thích bài đăng nếu cảm thấy quan tâm hay yêu thích bài đăng, ...
Tiền điều kiện	Đã đăng nhập tài khoản
Cách xử lý	Bước 1: Người dùng nhấn nút “Tim” trên bài đăng bất kỳ (người dùng có thể nhấn “Tim” thêm lần nữa để bỏ thích bài viết và ngược lại)
Kết quả	Icon tim sáng lên (đỏ), ngược lại icon tim tối đi (trắng) nếu người dùng unlike.
Ghi chú	

### 1.3.8. Chức năng thay đổi mật khẩu

**Bảng 8. Bảng chức năng thay đổi mật khẩu**

Mã yêu cầu	08
Tên chức năng	Thay đổi mật khẩu
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng đã có tài khoản
Mô tả tóm tắt	Người dùng có thể thay đổi mật khẩu để tăng tính bảo mật
Tiền điều kiện	Đã đăng nhập tài khoản
Cách xử lý	Bước 1: Người dùng vào giao diện thông tin cá nhân, chọn “đổi mật khẩu”. Bước 2: người dùng nhập mật khẩu hiện tại, mật khẩu mới và mật khẩu xác nhận, sau đó nhấn lưu. Bước 3: Hệ thống hiển thị kết quả.
Kết quả	Hệ thống hiển thị thông báo kết quả cập nhật mật khẩu.
Ghi chú	

**1.3.9. Chức đặt mua tour****Bảng 9. Bảng chức đặt mua tour**

Mã yêu cầu	09
Tên chức năng	Đặt mua tour
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng đã đăng nhập
Mô tả tóm tắt	Chức năng này cho phép người dùng mua tour theo nhu cầu
Tiền điều kiện	Đã đăng nhập tài khoản
Cách xử lý	Bước 1: Người dùng chọn vào tour phù hợp với nhu cầu, sau đó giao diện hiện thông tin chi tiết tour. Bước 2: Người dùng chọn “Đặt tour” để đến giao diện đặt tour. Bước 3: Người dùng kiểm tra thông tin cá nhân, chọn ngày khởi hành và số lượng, sau đó chọn “Xác nhận đặt tour”. Bước 4: Hệ thống hiển thị kết quả
Kết quả	Hệ thống hiển thị thông báo kết quả đặt tour
Ghi chú	

**1.3.10. Chức năng xem lịch sử hóa đơn****Bảng 10. Bảng chức năng xem lịch sử hóa đơn**

Mã yêu cầu	10
Tên chức năng	Xem lịch sử hóa đơn
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng đã đăng nhập
Mô tả tóm tắt	Chức năng này cho phép người dùng xem lại thông tin hóa đơn đã mua
Tiền điều kiện	Đã đăng nhập tài khoản
Cách xử lý	Bước 1: Người dùng nhấn chọn “Hóa đơn”. Bước 2: Hệ thống hiển thị giao diện lịch sử hóa đơn.
Kết quả	Hệ thống hiển thị thông tin lịch sử hóa đơn lên giao diện
Ghi chú	

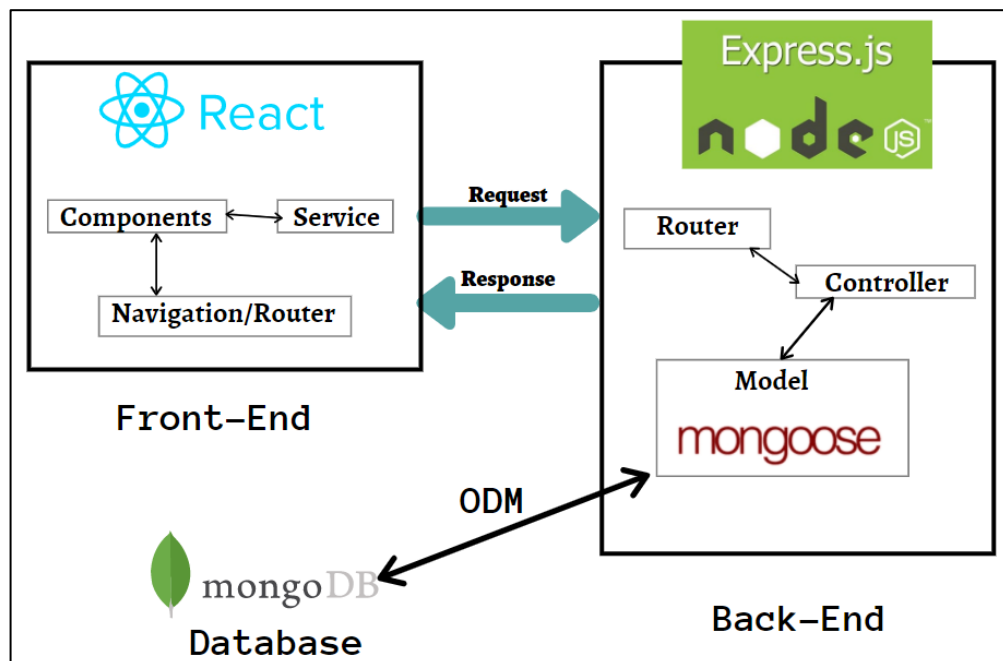
### 1.3.11. Chức năng xem chi tiết bài đăng tour

**Bảng 11. Bảng chức năng xem chi tiết bài đăng tour**

Mã yêu cầu	11
Tên chức năng	Xem chi tiết tour
Mức ưu tiên	Cao
Đối tượng sử dụng	Người dùng đã đăng nhập
Mô tả tóm tắt	Chức năng này cho phép người dùng xem thông tin chi tiết của tour
Tiền điều kiện	Đã đăng nhập tài khoản
Cách xử lý	Bước 1: Hệ thống hiển thị tất cả bài đăng. Bước 2: Người dùng chọn bài đăng theo nhu cầu của bản thân. Bước 3: Hệ thống hiển thị lên giao diện thông tin chi tiết của bài đăng tour.
Kết quả	Hệ thống hiển thị thông tin chi tiết tour lên giao diện
Ghi chú	

## 2. THIẾT KẾ VÀ CÀI ĐẶT

### 2.1. Tổng quan hệ thống

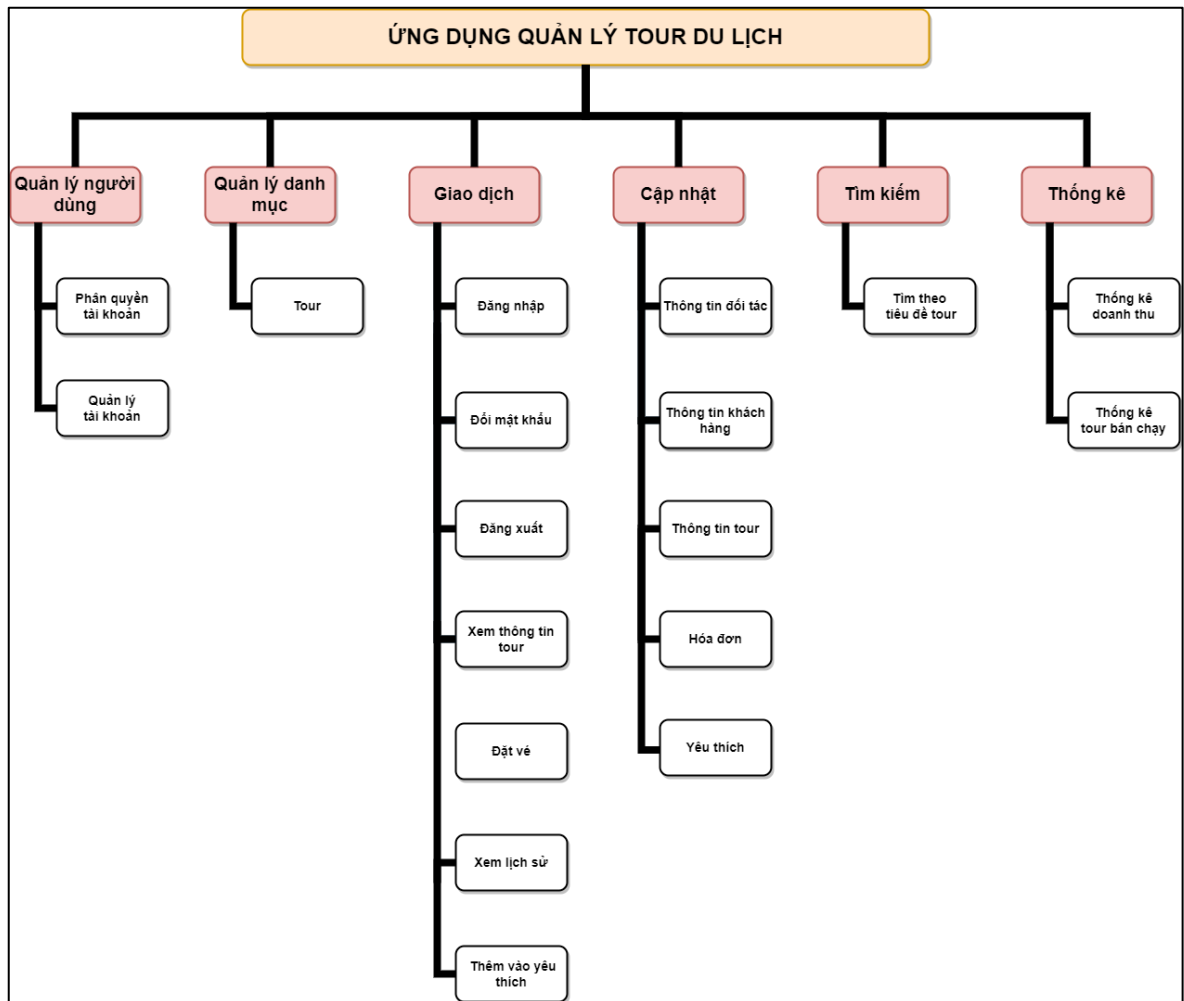


**Hình 2.1. Sơ đồ tổng quan hệ thống**

Mô hình tổng thể của phần mềm có thể chia thành bốn thành phần chính: tầng client (web app, mobile app), tầng trung gian, tầng server và tầng dữ liệu (database) được mô tả chi tiết như sau:

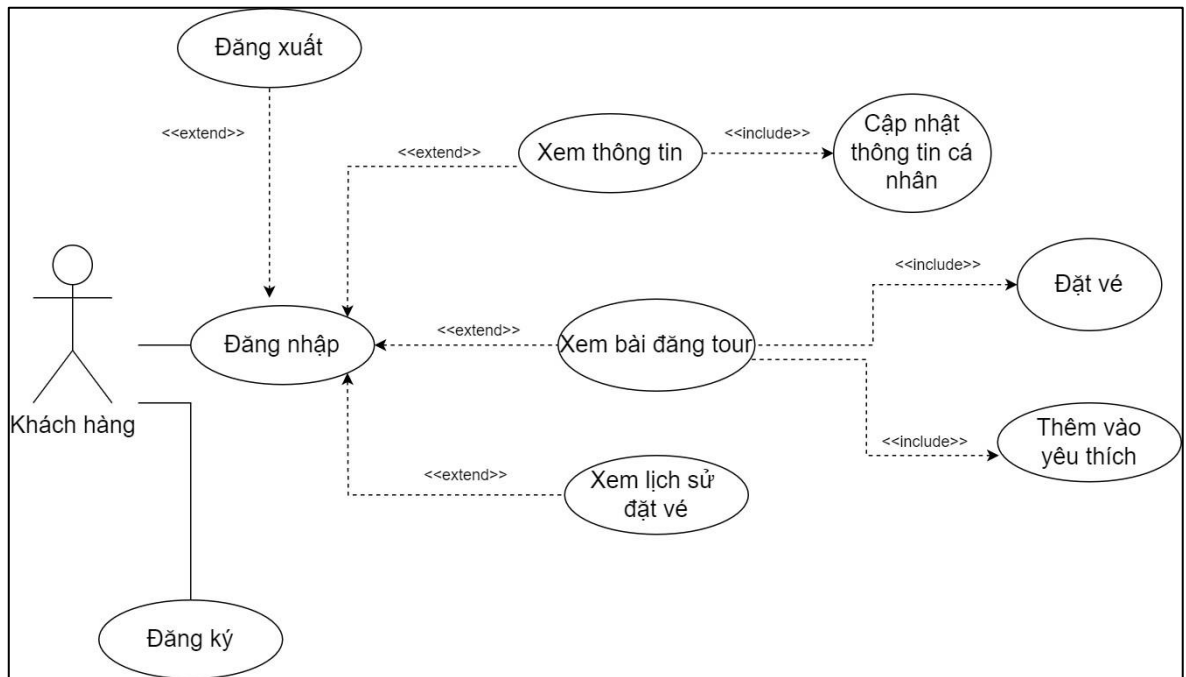
- Tầng client: là nơi người dùng tương tác với ứng dụng, chịu trách nhiệm tiếp nhận dữ liệu từ API và hiển thị cho người dùng. Đồng thời, tầng này cũng có nhiệm vụ chuyển đổi các thao tác của người dùng thành các yêu cầu gửi xuống server, tiếp nhận dữ liệu đầu vào, kiểm tra tính hợp lệ của dữ liệu và thực hiện các hành động ứng với phản hồi từ server.
- Tầng trung gian: đây là thành phần nằm giữa phần client và server, cho phép kết nối hai thành phần này lại với nhau, bao gồm mạng internet và RESTful API web service. Theo đó, mọi request được gửi từ client và mọi response trả về từ server đều sẽ đi qua tầng này, thông qua giao thức HTTP trong mạng internet.
- Tầng server: chịu trách nhiệm tiếp nhận các yêu cầu từ tầng client, xác thực và phân quyền người dùng, thực hiện xử lý logic, giao tiếp với database và trả về kết quả dưới định dạng JSON. Tầng này được cài đặt dựa trên kiến trúc RESTful API, đồng thời đảm bảo tính an toàn của hệ thống thông qua nhiều lớp bảo mật, thực hiện nối kết với CSDL và giao tiếp với các ứng dụng của bên thứ 3.
- Tầng dữ liệu: là một máy chủ được cài đặt CSDL chứa toàn bộ dữ liệu của hệ thống. Tầng này cho phép server được kết nối đến, thực hiện trích xuất, thao tác với dữ liệu, cho phép sao lưu và phục hồi dữ liệu để đảm bảo sự ổn định của hệ thống.

## 2.2. Mô hình phân rã tổng quát

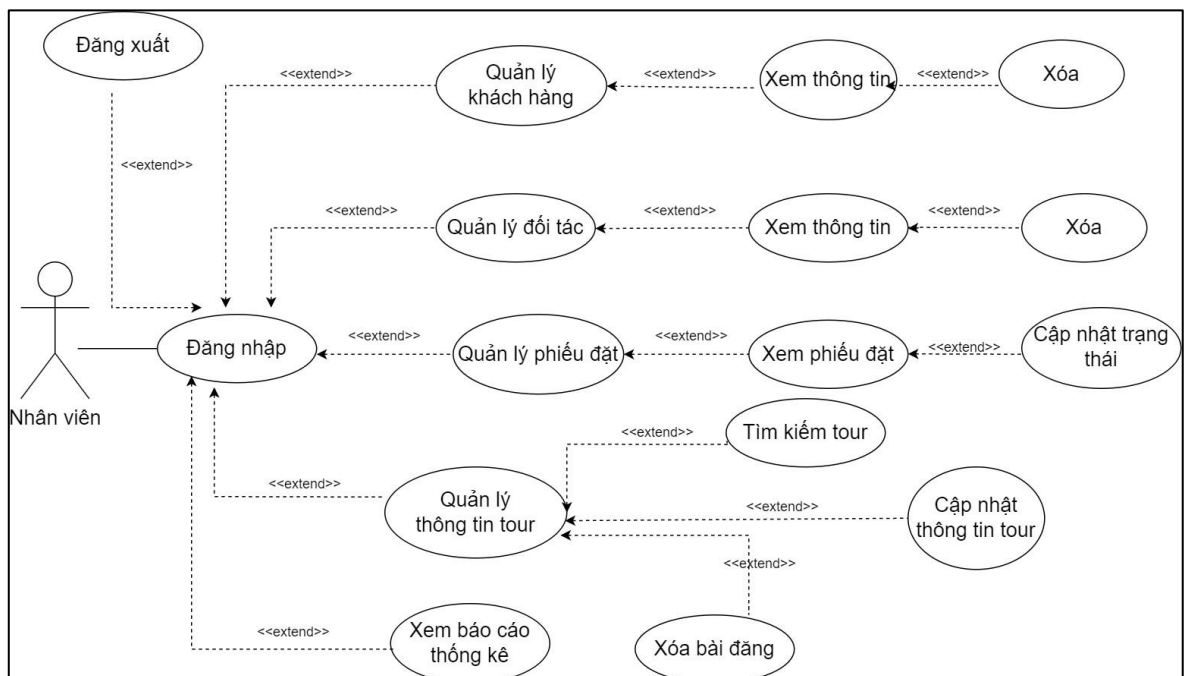


Hình 2.2. Sơ đồ phân rã chức năng.

### 2.2.1. Sơ đồ use case

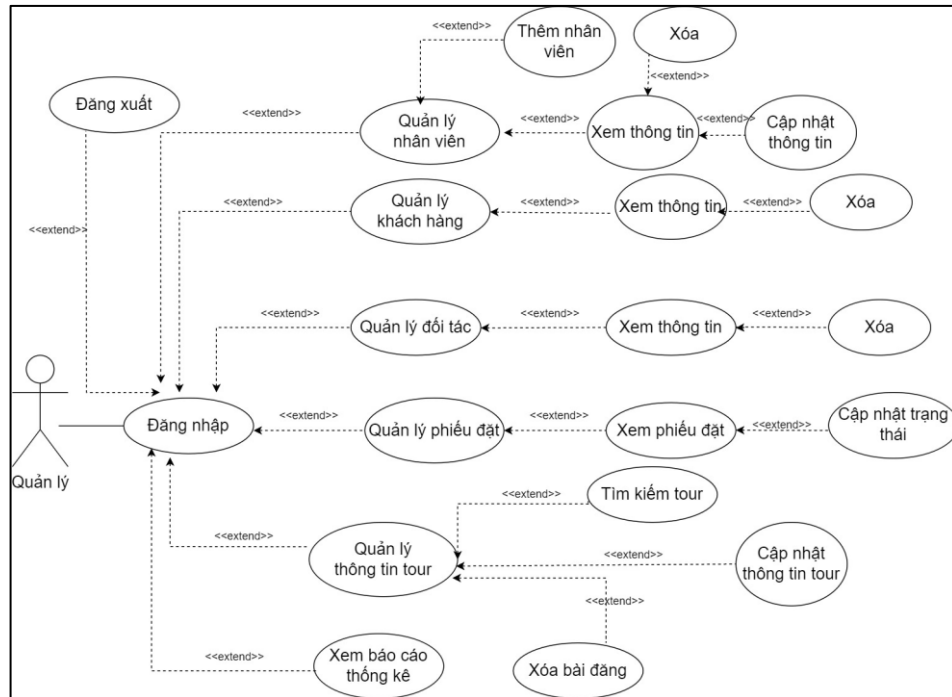


Hình 2.3. Sơ đồ use case Khách hàng



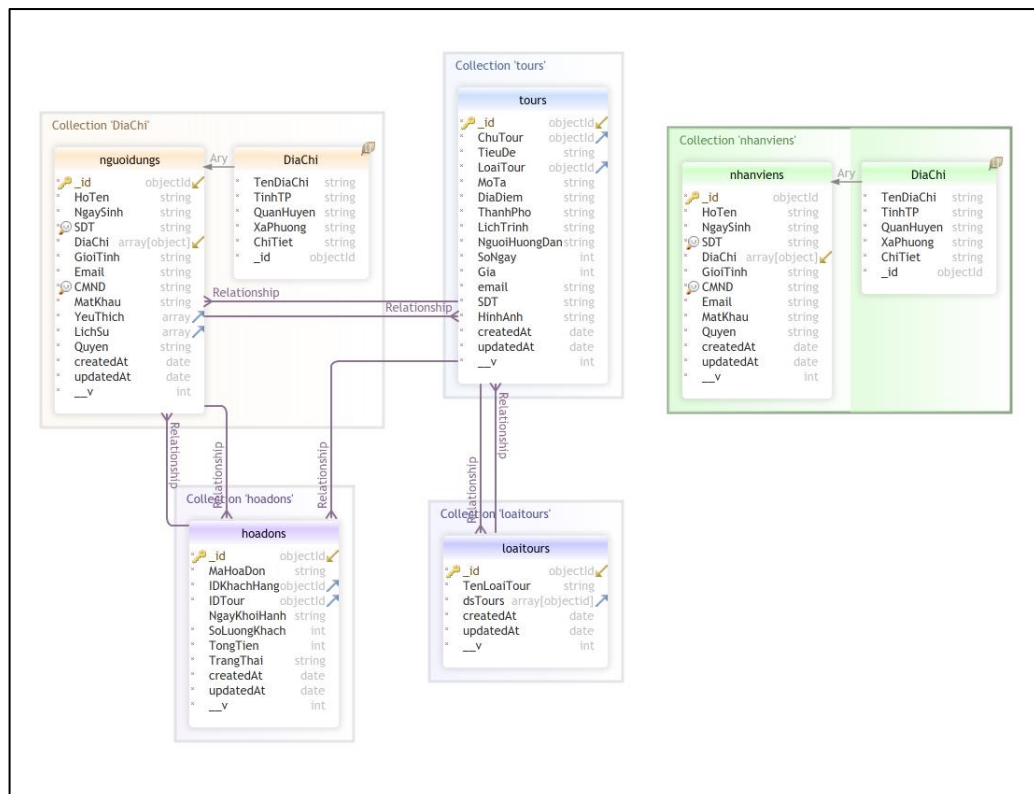
Hình 2.4. Sơ đồ use case Nhân viên

## PHẦN 2: NỘI DUNG VÀ KẾT QUẢ



Hình 2.5. Sơ đồ use case Quản lý

### 2.2.2. Sơ đồ MLD khi sử dụng MongoDB - hệ quản trị cơ sở dữ liệu NoSQL



Hình 2.6. Mô hình MLD

### 2.3. Thiết kế dữ liệu

**Bảng 12. Collection Người dùng**

Collection nguoidungs		
* Pk	_id	_ObjectId
*	HoTen	String
*	NgaySinh	String
*	SDT	String
*	DiaChi	Array[Object]
*	DiaChi._id	_ObjectId
*	DiaChi.TenDiaChi	String
*	DiaChi.TinhTP	String
*	DiaChi.QuanHuyen	String
*	DiaChi.XaPhuong	String
*	DiaChi.ChiTiet	String
*	GioiTinh	String
*	Email	String
*	CMND	String
*	MatKhau	String
*	YeuThich	Array[Object]
*	YeuThich._id	_ObjectId
*	LichSu	Array[Object]
*	LichSu.Tour	_ObjectId
*	LichSu.TrangThai	String
*	Quyen	String
*	Thoi_gian_tao	Date
*	Thoi_gian_cap_nhat	Date
Indexes		
Pk	_id	Pk



**Mô tả:**

- Thực thể người dùng bao gồm: Mã đối tượng người dùng (\_id), Tên đăng nhập người dùng (HoTen), mật khẩu (MatKhau), email (Email), số điện thoại (SDT), Ngày sinh (NgaySinh), giới tính (GioiTinh), chứng minh nhân dân/ căn cước công dân (CMND), mảng đối tượng yêu thích (YeuThich), thời gian tạo (Thoi\_gian\_tao), thời gian cập nhật (Thoi\_gian\_cap\_nhat), mảng các Object địa chỉ (DiaChi). Trong đó, mảng các Object địa chỉ thanh toán có: mã từng đối tượng địa chỉ (DiaChi.\_id), tên địa chỉ (DiaChi.TenDiaChi), xã/ phường (Dia\_chi.Xa\_Phuong), huyện/ quận (Dia\_chi.Huyen\_Quan), tỉnh/ thành phố (Dia\_chi.Tinh\_ThanhPho), mảng đối tượng sở thích gồm có: mã từng đối tượng sở thích (So\_thich.\_id).
- Mỗi khách hàng có một mã duy nhất để phân biệt với các khách hàng khác.
- Mã số hỗ trợ khách hàng lấy lại mật khẩu khi bị mất mật khẩu.

**Bảng 13. Collection tour**

Collection tours		
* Pk	_id	_ObjectId
*	TieuDe	String
*	LoaiTour	_ObjectId
*	Email	String
*	SDT	String
*	MoTa	String
*	DiaDiem	String
*	ThanhPho	String
*	LichTrinh	String
*	NguyenHuongDan	String
*	SoNgay	String
*	Gia	Number
*	HinhAnh	String
*	Thoi_gian_tao	Date

## PHẦN 2: NỘI DUNG VÀ KẾT QUẢ

---

*	Thoi_gian_cap_nhat	Date
Indexes		
Pk	_id	Pk

### Mô tả:

- Thực thể tour bao gồm: Mã tour (\_id), tiêu đề (TieuDe), email (Email), số điện thoại (SDT), loại tour (LoaiTour), mô tả thông tin của tour (MoTa), địa điểm (DiaDiem), thành phố (ThanhPho), lịch trình (LichTrinh), người hướng dẫn (NguoiHuongDan), số ngày đi (SoNgay), giá (Gia), hình ảnh (HinhAnh), thời gian tạo (Thoi\_gian\_tao), thời gian cập nhật (Thoi\_gian\_cap\_nhat).
- Mỗi tour có một mã duy nhất để phân biệt với các tour khác.

**Bảng 14. Collection Loại Tour**

Collection loaitours		
*	_id	_ObjectId
Pk		
*	TenLoaiTour	String
*	Thoi_gian_tao	Date
*	Thoi_gian_cap_nhat	Date
Indexes		
Pk	_id	Pk

### Mô tả:

- Thực thể loại tour bao gồm: Mã loại tour (\_id), tên loại tour (TenLoaiTour),
- thời gian tạo (Thoi\_gian\_tao), thời gian cập nhật (Thoi\_gian\_cap\_nhat).
- Mỗi loại tour có một mã duy nhất để phân biệt với các loại tour khác.
- Giá sẽ phụ thuộc vào số lượng khách tham gia.

**Bảng 15. Collection Hóa đơn**

Collection hoadons		
* Pk	_id	_ObjectId
*	MaHoaDon	String
*	IDKhachHang	_ObjectId
*	IDTour	_ObjectId
*	NgayKhoiHanh	String
*	SoLuongKhach	Number
*	TongTien	Number
*	Thoi_gian_tao	Date
*	Thoi_gian_cap_nhat	Date
Indexes		
Pk	_id	Pk

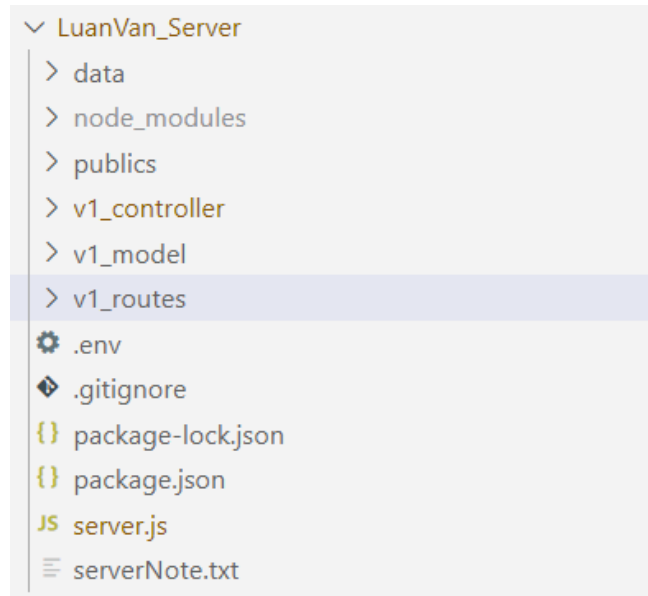
**Mô tả:**

Thực thể loại hóa đơn bao gồm: Mã id hóa đơn (\_id), mã hóa đơn hiển thị trên ứng dụng (MaHoaDon), id của khách hàng (IDKhachHang), id của tour (IDTour), ngày khởi hành (NgayKhoiHanh), số lượng khách (SoLuongKhach), thời gian tạo (Thoi\_gian\_tao), thời gian cập nhật (Thoi\_gian\_cap\_nhat).

- Mỗi loại hóa đơn có một mã duy nhất để phân biệt với các loại hóa đơn khác.
- Mỗi hóa đơn sẽ có thông tin người dùng chọn ngày đi và số lượng khách khác nhau.

### 2.4. Cấu trúc dự án

#### 2.4.1. Cấu trúc thư mục



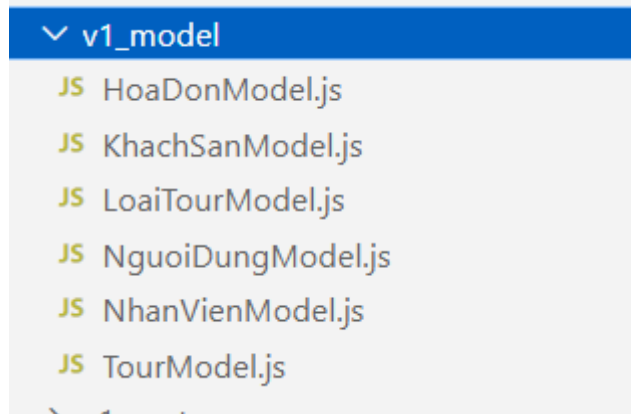
**Hình 2.7. Cấu trúc thư mục server**

- Thư mục `publics`: là nơi chứa các file ảnh cần dùng trong ứng dụng.
- Thư mục `v1_controller`: chứa các file script để thêm sửa dữ liệu vào các collection khi thực hiện thao tác thay đổi trên cơ sở dữ liệu.
- Thư mục `v1_model`: chứa các file script tạo model kết nối đến cơ sở dữ liệu MongoDB.
- Thư mục `v1_routes`: chứa các file script giúp định tuyến luồng dữ liệu trong ứng dụng khi có yêu cầu từ client một cách rõ ràng.

#### 2.4.2. Thể hiện mô hình MVC trong ứng dụng

##### - Khởi tạo Model:

Trong thư mục `v1_model` chứa các script khởi tạo các schema, ở mỗi schema thể hiện các thuộc tính của đối tượng. Trong đó schema `NhanVien` và schema `NguoaiDung` chứa các đặc tính của người dùng, schema `Tour` và schema `LoaiTour` chứa thông tin đặc tính của tour du lịch, schema `HoaDon` chứa thông tin hóa đơn của người dùng khi mua hàng.



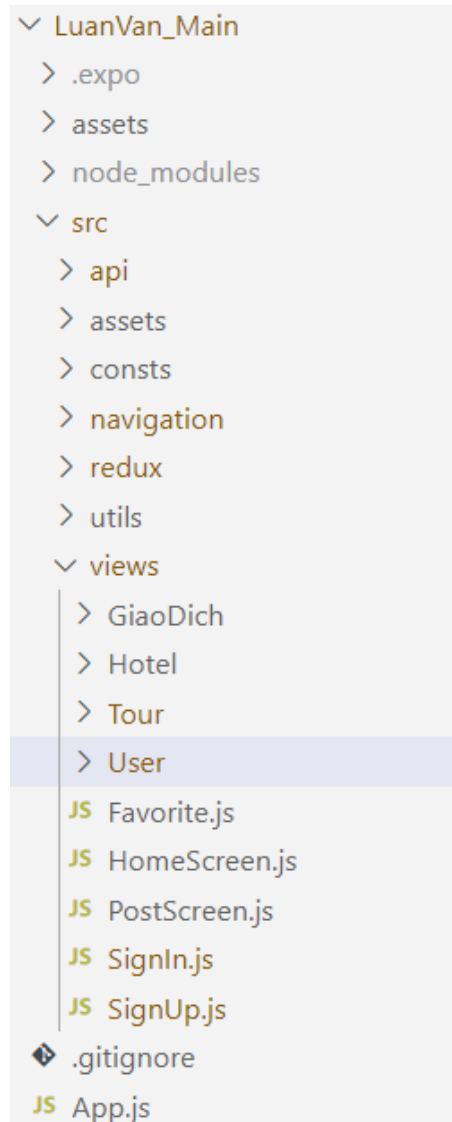
**Hình 2.8. Cấu trúc thư mục khởi tạo model**

```
const TourSchema = new mongoose.Schema(  
  {  
    ChuTour: {  
      type: mongoose.Schema.Types.ObjectId,  
      ref: "NguoiDung",  
    },  
    TieuDe: {  
      // Hiển thị tiêu đề  
      type: String,  
    },  
    LoaiTour: { ... },  
    MoTa: { ... },  
    DiaDiem: { ... },  
    ThanhPho: { ... },  
    LichTrinh: { ... },  
    NguoiHuongDan: { ... },  
    SoNgay: { ... },  
    Gia: { ... },  
    email: { ... },  
    SDT: { ... },  
    HinhAnh: { ... },  
  },  
  { timestamps: true }  
)
```

**Hình 2.9. Schema Tour**

### - Khởi tạo View:

Trong package View chứa các thể hiện GUI và xử lý sự kiện trên GUI của ứng dụng.



**Hình 2.10. Cấu trúc thư mục View**

### - Khởi tạo Controller:

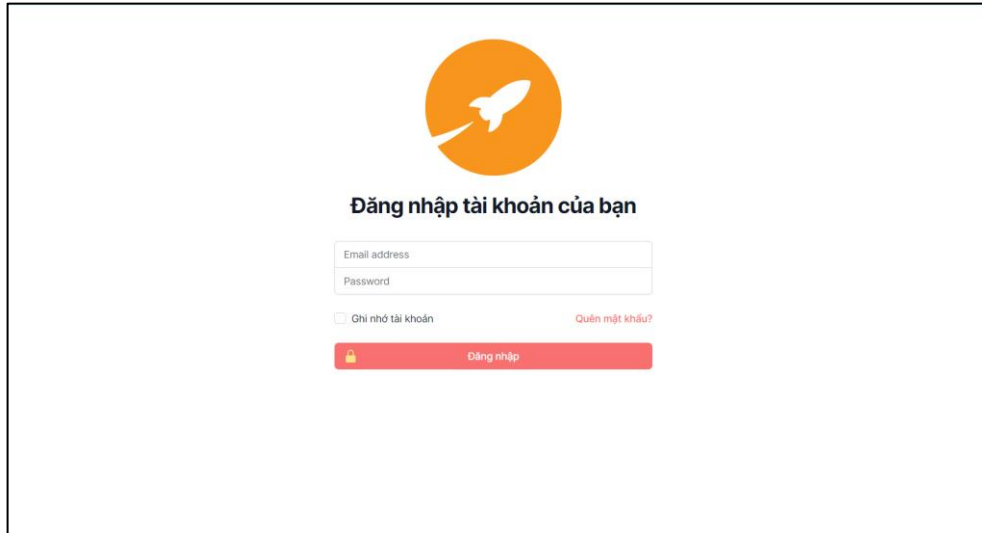
Controller bao gồm 2 thư mục `v1_controller` và `v1_routes` và file `server.js`. Trong file `server.js` dùng để khởi tạo kết nối với cơ sở dữ liệu, tạo đường dẫn cho HTTP Request. Thư mục `v1_routes` chứa các file định tuyến khi nhận được yêu cầu từ client đến controller. Thư mục `v1_controller` chứa các file script thực hiện xử lý data từ client, xác thực đăng nhập, thêm xóa dữ liệu trong cơ sở dữ liệu.

Thư mục v1\_controller:

- File `NguoiDungController.js`
  - `addNguoiDung`: Thêm một người dùng
  - `getOne`: Xử lý đăng nhập
  - `updateNguoiDung`: Xử lý update data người dùng
  - `getUser`: Lấy thông tin người dùng theo id
  - `getAll`: Lấy danh sách tất cả thông tin người dùng
  - `getAllMua`: Lấy danh sách người dùng có quyền là MUA
  - `getAllBan`: Lấy danh sách người dùng có quyền là BAN
  - `getLike`: Lấy danh sách id tour yêu thích của người dùng
- File `TourController.js`:
  - `addTour`: Thêm dữ liệu tour
  - `deleteTour`: Xóa tour theo id
  - `updateTour`: Update thông tin tour
  - `searchTour`: Tìm kiếm tour theo tiêu đề
  - `getAll`: Lấy danh sách tất cả tour
  - `getOnebyID`: Lấy thông tin tour theo id
  - `getByList`: Lấy danh sách tour theo danh sách id có sẵn
- File `HoaDonController.js`:
  - `addHoaDon`: Thêm dữ liệu hóa đơn
  - `updateHoaDon`: Xử lý cập nhật dữ liệu hóa đơn
  - `deleteHoaDon`: Xóa hóa đơn theo id
  - `getAll`: Lấy danh sách tất cả hóa đơn
  - `getByIDHoaDon`: Lấy thông tin hóa đơn theo id
  - `getByMaHoaDon`: Lấy thông tin hóa đơn theo mã hóa đơn
  - `getByIDNguoiDung`: Lấy danh sách hóa đơn của người dùng

### 3. KẾT QUẢ THỰC HIỆN

#### 3.1. Chức năng đăng nhập hệ thống



The screenshot displays a login form with the following elements:

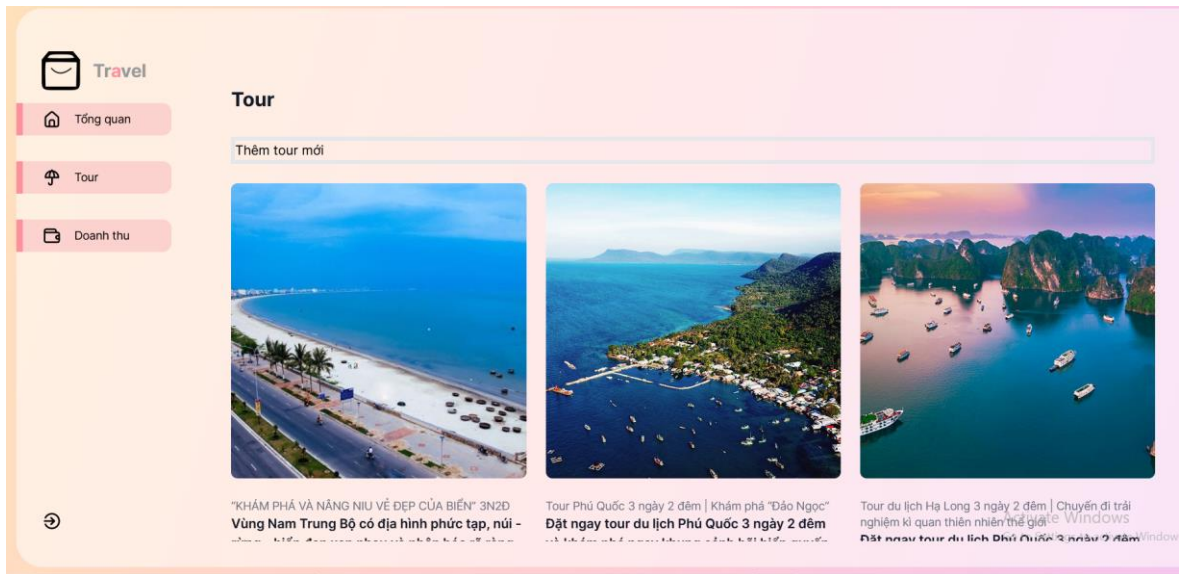
- A logo featuring a white rocket inside an orange circle.
- The title "Đăng nhập tài khoản của bạn" (Login your account).
- Input fields for "Email address" and "Password".
- A checkbox labeled "Ghi nhớ tài khoản" (Remember account).
- A link labeled "Quên mật khẩu?" (Forgot password?).
- A red button labeled "Đăng nhập" (Login).

**Hình 2.11. Giao diện đăng nhập**

Tại đây người dùng cần đăng nhập vào hệ thống bằng email và password đã đăng ký. Sau khi nhập đầy đủ thông tin, người dùng nhấn đăng nhập, hệ thống sẽ sử dụng “fineOne” trong thư viện Mongoose tìm kiếm người dùng và kiểm tra dữ liệu mà người dùng đã nhập có khớp với email và password người dùng đã đăng ký bằng phương thức “compare” trong thư viện Bcrypt. Khi kiểm tra thông tin hoàn thành, nếu thành công hệ thống sẽ trả về thông tin của người dùng, ngược lại sẽ trả về không tìm thấy tài khoản (NoUser) hoặc sai mật khẩu (NoPassword).



### 3.2. Chức năng xem danh sách bài đăng của người dùng là khách hàng

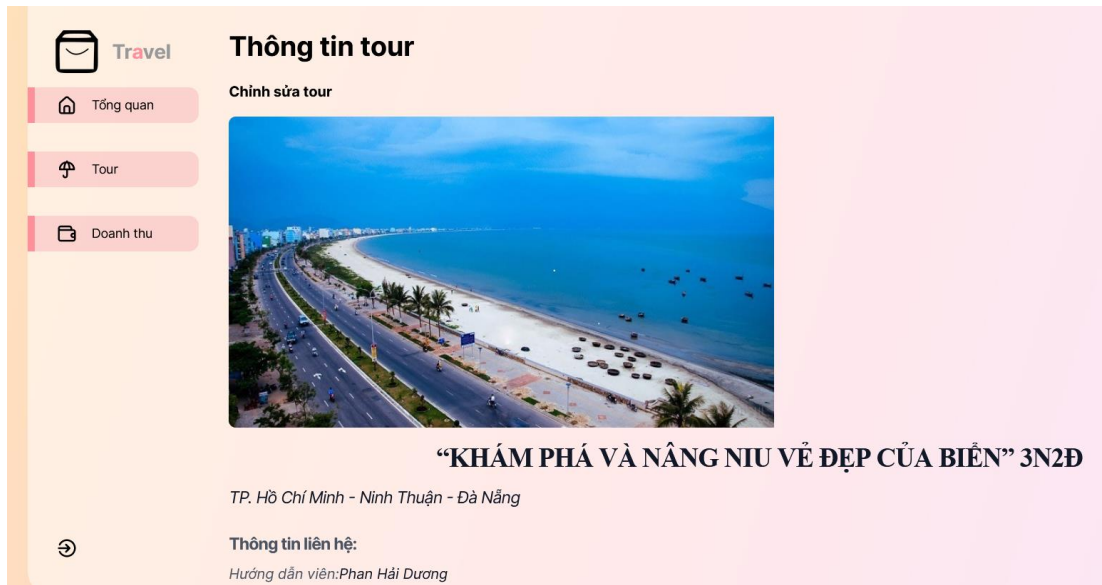


**Hình 2.12. Giao diện xem danh sách bài đăng**

Vì người dùng là người đã tạo bài đăng tour. Nên tại giao diện này, hệ thống sẽ tìm kiếm danh sách bài đăng tour bằng cách tìm kiếm tour theo id của người dùng sử dụng phương thức “findById”, mà id người dùng đã được hệ thống trả kết quả từ lúc đăng nhập thành công.

Khi hệ thống tìm kiếm bài đăng tour không chỉ tìm kiếm theo id người dùng mà hệ thống còn có thể tìm kiếm bài đăng theo id của bài đăng, theo tiêu đề hay một danh sách id cho trước.

### 3.3. Chức năng chi tiết bài đăng của người dùng là khách hàng



Hình 2.13. Giao diện chi tiết bài đăng

Như đã đề cập, hệ thống có thể tìm kiếm thông tin bài đăng tour với đường dẫn “Link” từ “/tour/:id” theo id tour để có thể giúp người dùng xem thông tin bài đăng tour một cách chi tiết hơn.

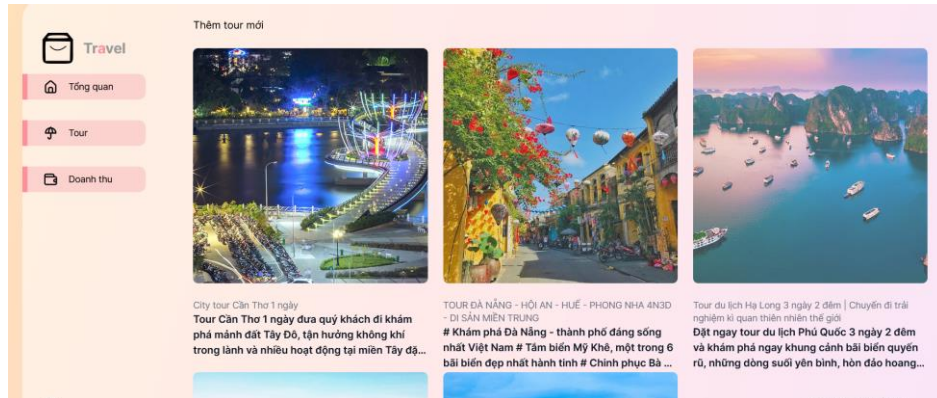
### 3.4. Chức năng thêm Tour mới của người dùng là khách hàng

Hình 2.14. Giao diện thêm bài đăng

Chức năng thêm bài đăng tour mới cho phép người dùng tạo một bài đăng mới trên hệ thống.

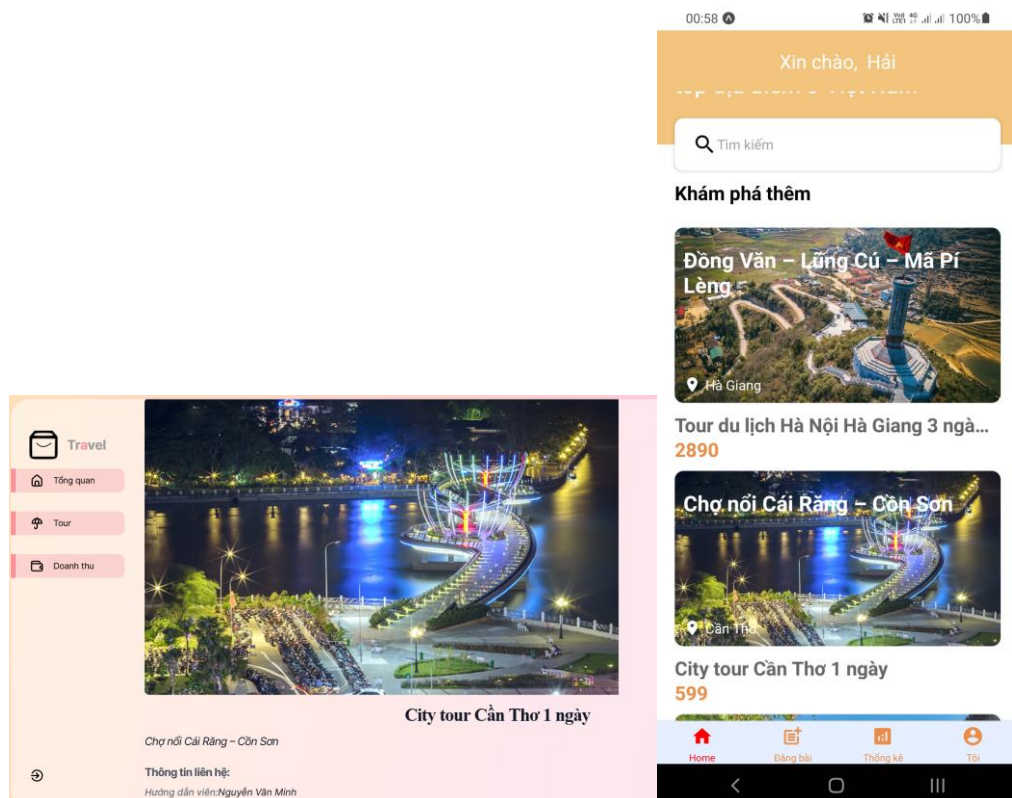
## PHẦN 2: NỘI DUNG VÀ KẾT QUẢ

Người dùng tiến hành nhập các thông tin cần thiết của bài đăng tour, sự kiện onChange sẽ được kích hoạt ngay khi giá trị của thẻ thay đổi và gán dữ liệu vào form sau đó nhấn nút tạo bài đăng thì thư viện axios sử dụng phương thức “post” truyền dữ liệu lên server. Ngay khi hệ thống nhận được yêu cầu sẽ kiểm tra dữ liệu, nếu hợp lệ sẽ tiến hành thêm bài đăng vào cơ sở dữ liệu ngược lại sẽ thông báo lỗi.



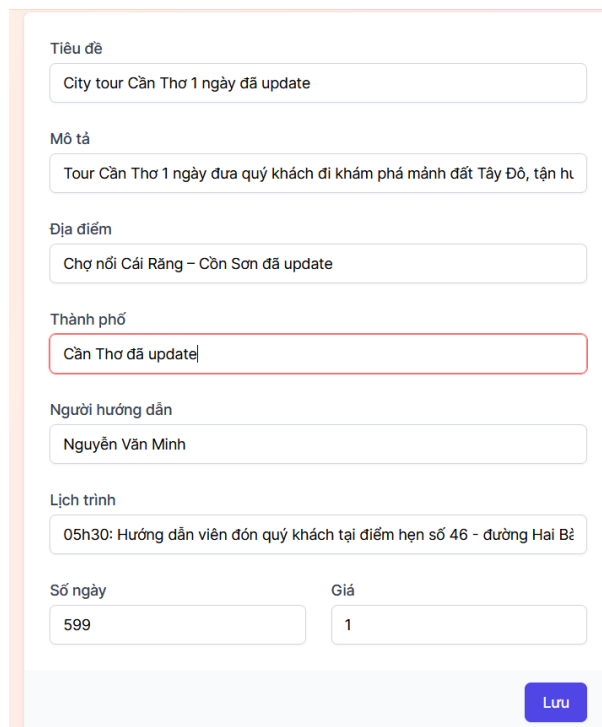
Hình 2.15. Giao diện danh sách tour sau khi thêm

Sau khi thêm thành công, chúng ta chuyển sang trang danh sách tour sẽ có bài đăng tour vừa mới thêm. Ngay lập tức ứng dụng di động cũng người dùng cũng sẽ lấy được thông tin bài đăng mới này và xem được thông tin chi tiết bài đăng này.



Hình 2.16. Giao diện chi tiết bài đăng sau khi thêm mới

### 3.5. Chức năng chỉnh sửa bài đăng của người dùng là khách hàng



The screenshot shows a form for editing a post. The fields are as follows:

- Tiêu đề**: City tour Cần Thơ 1 ngày đã update
- Mô tả**: Tour Cần Thơ 1 ngày đưa quý khách đi khám phá mảnh đất Tây Đô, tận h
- Địa điểm**: Chợ nổi Cái Răng – Cồn Sơn đã update
- Thành phố**: Cần Thơ đã update
- Người hướng dẫn**: Nguyễn Văn Minh
- Lịch trình**: 05h30: Hướng dẫn viên đón quý khách tại điểm hẹn số 46 - đường Hai Bè
- Số ngày**: 599
- Giá**: 1

A blue button labeled "Lưu" (Save) is located at the bottom right of the form.

**Hình 2.17. Giao diện chỉnh sửa bài đăng**

Chức năng chỉnh sửa bài đăng giúp người dùng thay đổi bài đăng khi thông tin trên bài đăng đã không còn phù hợp.

Người dùng tiến hành thay đổi các thông tin không còn phù hợp sau đó nhấn lưu. Hệ thống sẽ tiến hành rest api, khi server nhận được yêu cầu sẽ tiến hành tìm kiếm bài đăng tour theo id tour mà người dùng muốn cập nhật bằng hàm “findById”, sau đó tiến hành cập nhật các thông tin mà người dùng đã thay đổi và lưu vào cơ sở dữ liệu, trả thông báo về cho người dùng cập nhật thành công nếu không có lỗi xảy ra.



**Hình 2.18. Giao diện chi tiết bài đăng sau khi chỉnh sửa**

## PHẦN 2: NỘI DUNG VÀ KẾT QUẢ

Sau khi chỉnh sửa bài đăng tour thành công khi ta vào lại trang thông tin chi tiết tour sẽ thấy thông tin đã được chỉnh sửa. Về ứng dụng phía người dùng cũng sẽ thấy được sự thay đổi này.



Hình 2.19. Giao diện phía di động sau khi chỉnh sửa

### 3.6. Chức năng thống kê của người dùng là khách hàng

Giao dịch gần đây					
Mã đơn hàng	Tour du lịch	Khách hàng	Số lượng	Ngày	
20221205HD14519	"KHÁM PHÁ VÀ NÂNG NIU VẺ ĐẸP CỦA BIỂN" 3N2Đ	Nguyễn Văn Long	1	12	
20221206HD32356	Tour Phú Quốc 3 ngày 2 đêm   Khám phá "Đảo Ngọc"	Nguyễn Văn Long	1	12	
20221206HD39494	"KHÁM PHÁ VÀ NÂNG NIU VẺ ĐẸP CỦA BIỂN" 3N2Đ	Nguyễn Văn Long	1	7-12-2022	
20221206HD45322	"KHÁM PHÁ VÀ NÂNG NIU VẺ ĐẸP CỦA BIỂN" 3N2Đ	Nguyễn Văn Long	5		
20221206HD54732	"KHÁM PHÁ VÀ NÂNG NIU VẺ ĐẸP CỦA BIỂN" 3N2Đ	Nguyễn Văn Long	10		
20221206HD64275	"KHÁM PHÁ VÀ NÂNG NIU VẺ ĐẸP CỦA BIỂN" 3N2Đ	Nguyễn Văn Long	10	28-12-2022	
20221207HD72097	"KHÁM PHÁ VÀ NÂNG NIU VẺ ĐẸP CỦA BIỂN" 3N2Đ	Nguyễn Văn Long	3	14-12-2022	
20221209HD89545	"KHÁM PHÁ VÀ NÂNG NIU VẺ ĐẸP CỦA BIỂN" 3N2Đ	Phan Hải Dương	1		
20221209HD98137	Tour du lịch Hạ Long 3 ngày 2 đêm   Chuyến đi trải nghiệm kì quan thiên nhiên thế giới	Nguyễn Văn Long	2		

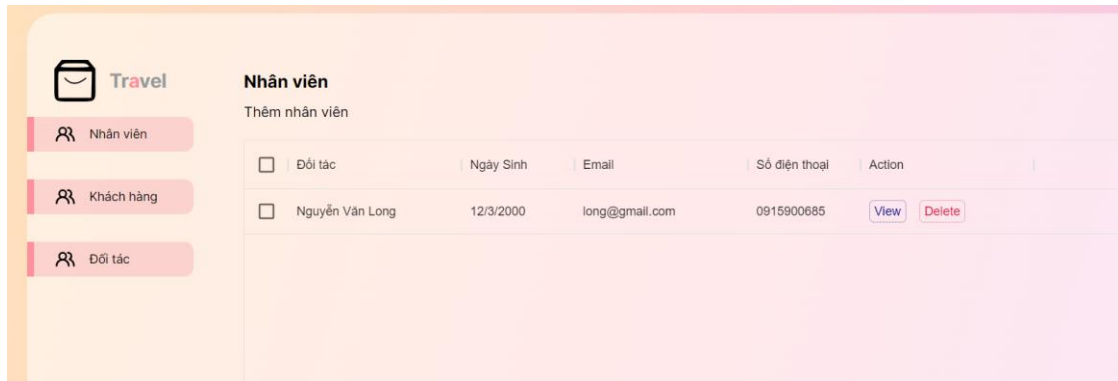
Hình 2.20. Giao diện thống kê



## PHẦN 2: NỘI DUNG VÀ KẾT QUẢ

Giao diện thống kê hiển thị danh sách các hóa đơn do thư viện axios dùng phương thức “get” để kéo dữ liệu về client khi khách hàng mua tour thành công.

### 3.7. Chức năng quản lý nhân viên



Hình 2.21. Chức năng quản lý nhân viên

Chức năng quản lý nhân viên cho phép người dùng xem thông tin chi tiết của nhân viên đó hoặc có thể xóa thông tin nhân viên nếu người dùng có quyền quản lý.

### 3.8. Chức năng thêm một nhân viên mới

Họ và tên

Ngày sinh

Giới tính

Email

CMND

Số điện thoại

Mật khẩu

Thành phố

Quận/Huyện

Xã/Phường

Địa chỉ

Lưu

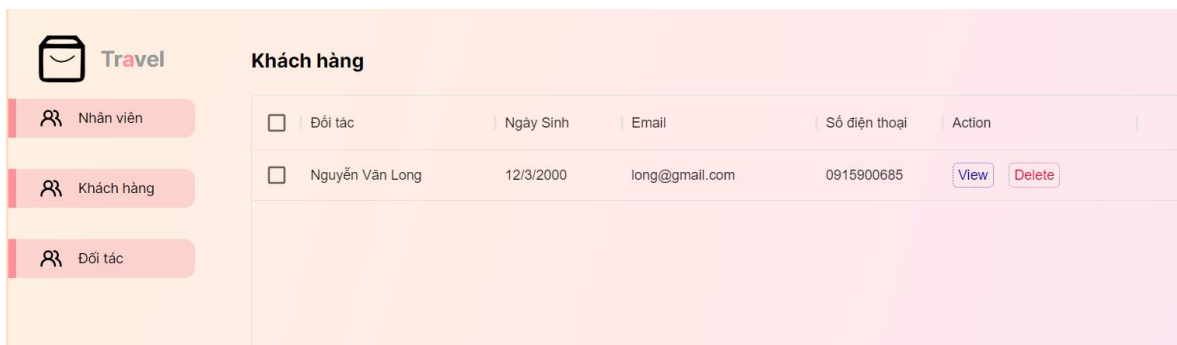
Hình 2.22. Chức năng thêm một nhân viên mới

## PHẦN 2: NỘI DUNG VÀ KẾT QUẢ

Chức năng thêm nhân viên mới cho phép người dùng tạo một nhân viên mới trên hệ thống.

Người dùng tiến hành nhập các thông tin cần thiết của nhân viên, sự kiện onChange sẽ được kích hoạt ngay khi giá trị của thẻ thay đổi và gán dữ liệu vào form sau đó nhấn nút lưu nhân viên. Ngay khi hệ thống nhận được yêu cầu sẽ kiểm tra dữ liệu, nếu hợp lệ sẽ tiến hành thêm nhân viên vào cơ sở dữ liệu ngược lại sẽ thông báo lỗi.

### 3.9. Chức năng quản lý khách hàng

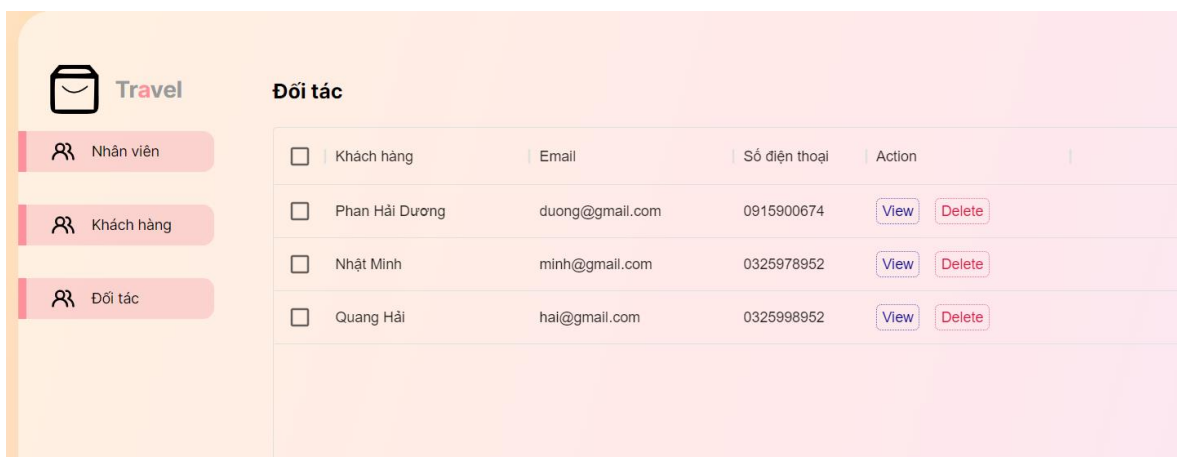


<input type="checkbox"/>	Đôi tác	Ngày Sinh	Email	Số điện thoại	Action
<input type="checkbox"/>	Nguyễn Văn Long	12/3/2000	long@gmail.com	0915900685	<a href="#">View</a> <a href="#">Delete</a>

Hình 2.23. Chức năng quản lý khách hàng

Chức năng quản lý khách hàng cho phép người dùng xem thông tin chi tiết của khách hàng và có thể xóa khách hàng ra khỏi hệ thống nếu khách hàng có những biểu hiện lỗi nghiêm trọng.

### 3.10. Chức năng quản lý đối tác



<input type="checkbox"/>	Khách hàng	Email	Số điện thoại	Action
<input type="checkbox"/>	Phan Hải Dương	duong@gmail.com	0915900674	<a href="#">View</a> <a href="#">Delete</a>
<input type="checkbox"/>	Nhật Minh	minh@gmail.com	0325978952	<a href="#">View</a> <a href="#">Delete</a>
<input type="checkbox"/>	Quang Hải	hai@gmail.com	0325998952	<a href="#">View</a> <a href="#">Delete</a>

Hình 2.24. Chức năng quản lý đối tác

Chức năng quản lý đối tác cho phép người dùng xem thông tin chi tiết của đối tác và có thể xóa đối tác ra khỏi hệ thống nếu đối tác có những biểu hiện lỗi nghiêm trọng.

### PHẦN 3: KẾT LUẬN

#### 1. KẾT QUẢ ĐẠT ĐƯỢC

- củng cố và áp dụng được kiến thức đã được học ở nhà trường về các ngôn ngữ HTML, CSS, JavaScript, phương pháp phân tích và thiết kế CSDL, v.v.
- Tìm hiểu và áp dụng thành công các công nghệ chưa được giảng dạy ở trường bao gồm ReactJs, React Native, NodeJs, RESTful API và CSDL phi quan hệ MongoDB.
- Xây dựng được RESTful API server sử dụng HTTP requests để giao tiếp, thao tác với dữ liệu và trả về cho người dùng.
- Thực hiện bảo mật xác thực người dùng và phân quyền cho người dùng thực hiện các tính năng được cho phép.
- Tìm hiểu cách sử dụng một số framework, thư viện phổ biến như ExpressJs, React, Mongoose, Axios, ...
- Xử lý hình ảnh được upload bởi người dùng.
- Tìm hiểu cách cài đặt một hệ thống client-server, phân chia thành phần và gắn kết các thành phần với nhau theo mô hình MVC.
- Kết nối và thao tác thành công với hệ thống máy chủ CSDL từ xa và dữ liệu dùng chung được lưu trữ trên đám mây.
- Biết được cách triển khai một ứng dụng lên internet.
- Học được cách làm việc và lưu trữ source code thông qua Github.

#### 2. CÁC ĐIỂM HẠN CHẾ

- Hệ thống có thể hoạt động khá chậm trên một số thiết bị có cấu hình yếu, thời gian chờ còn phụ thuộc vào tốc độ mạng.
- Thời gian tải lần đầu còn chậm vì dữ liệu phải lưu vào bộ nhớ.
- Hệ thống còn thực hiện ở quy mô nhỏ.
- Hệ thống chưa thật sự đáp ứng hết nhu cầu cho người dùng.

#### 3. HƯỚNG PHÁT TRIỂN

- Tối ưu hóa tốc độ hệ thống và trên các thiết bị có cấu hình yếu.



- Cải tiến giao diện theo xu hướng hiện nay.
- Nâng cao bảo mật người dùng.
- Cần tìm hiểu thêm về mong muốn của người dùng trong quá trình tương tác với ứng dụng để tiếp tục hoàn thiện chức năng đáp ứng được nhu cầu của người dùng.

## TÀI LIỆU THAM KHẢO

### Tiếng Việt

- [1] Bùi Thị Hồng Phúc (2012), Nghiên cứu CSDL NoSQL MongoDB ứng dụng vào tính toán lưu trữ phân tán. Luận văn Thạc sĩ Công nghệ thông tin, Trường Đại học Cần Thơ
- [2] Cơ bản về mô hình MVC (2022). Địa chỉ <Cơ bản về mô hình MVC (viblo.asia)> [Truy cập ngày 6/12/2022].

### Tiếng Anh

- [3] Bonnie Eisenman (2017), Learning React Native: Building Native Mobile Apps with JavaScript, O'Reilly Media, pp 8-10.
- [4] Express/Node introduction (2022). Địa chỉ <Express/Node introduction - Learn web development | MDN (mozilla.org)> [Truy cập ngày 7/12/2022].
- [5] Getting Started (2022). Địa chỉ <Getting Started | Axios Docs (axios-http.com)> [Truy cập ngày 6/12/2022].
- [6] Introduction to Node.js (2022). Địa chỉ <Introduction to Node.js (nodejs.dev)> [truy cập ngày 6/12/2022].
- [7] REST API (Introduction) (2022). Địa chỉ <REST API (Introduction) - GeeksforGeeks> [Truy cập ngày 6/12/2022].
- [8] What is JavaScript? (2021). Địa chỉ <Introduction - JavaScript | MDN (mozilla.org)> [Truy cập năm 2021].
- [9] Document Guides Mongoose (2022). Địa chỉ <Mongoose v6.8.0: Schemas (mongoosejs.com)> [Truy cập năm 2022]