

**ĐẠI HỌC QUỐC GIA HÀ NỘI  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**



**BÁO CÁO CUỐI KỲ  
HỌC PHẦN: CƠ SỞ DỮ LIỆU WEB VÀ HTTT**

**ĐỀ TÀI:  
XÂY DỰNG WEBSITE QUẢN LÝ DU LỊCH**

Giảng viên hướng dẫn : **Vũ Tiến Dũng**

: **Phạm Duy Phương**

Nhóm sinh viên thực hiện : **Nhóm 12**

*Năm học 2023-2024*

## DANH SÁCH THÀNH VIÊN

STT	Họ tên	Mã sinh viên	Phân công công việc
1	Nguyễn Thị Huyền Sâm	21002170	Use case, Database
2	Phạm Hùng	21000143	API
3	Lê Trường Giang	21002135	Giao diện User
4	Lê Gia Huy	21002149	Giao diện Admin

# LỜI NÓI ĐẦU

Trong sự phát triển mạnh mẽ của khoa học công nghệ thế giới, có thể nói ngành công nghệ thông tin phát triển với tốc độ nhanh nhất và ngày càng thể hiện được vai trò to lớn, hết sức quan trọng đối với toàn bộ đời sống con người. Hầu hết các lĩnh vực trong xã hội đều áp dụng công nghệ thông tin, nhiều phần mềm được phát triển đã đem lại hiệu quả không thể phủ nhận. Song thực tiễn luôn đặt ra những yêu cầu mới đòi hỏi ngành công nghệ thông tin không ngừng phát triển để thỏa mãn và đáp ứng những nhu cầu của cuộc sống.

Hiện nay, nhu cầu du lịch ngày càng tăng cao, loại hình đặt vé du lịch trực tiếp từ các đại lý du lịch truyền thống trở nên bất tiện và tốn thời gian của người tiêu dùng. Bên cạnh đó, hoạt động của các công ty, hãng du lịch có quy mô ngày càng lớn khiến cho việc tổ chức quản lý các dịch vụ du lịch trở nên phức tạp và khó khăn hơn.

Dựa trên ý tưởng đó, nhóm chúng em lựa chọn đề tài "Xây dựng website quản lý du lịch" cho phép người dùng đặt các dịch vụ du lịch trực tuyến như: đặt vé máy bay, đặt phòng khách sạn và tour du lịch,... với mong muốn giải quyết một phần khó khăn của loại hình đặt tour du lịch truyền thống.

Bài báo cáo gồm các phần chính sau:

- *Chương 1: Tổng quan*
- *Chương 2: Một số Công nghệ sử dụng*
- *Chương 3: Phân tích thiết kế hệ thống*
- *Chương 4: Thiết kế cơ sở dữ liệu*
- *Chương 5: Phát triển hệ thống*

# MỤC LỤC

<b>LỜI NÓI ĐẦU</b>	<b>i</b>
<b>DANH MỤC KÝ HIỆU VÀ TỪ VIẾT TẮT</b>	<b>iv</b>
<b>DANH MỤC HÌNH ẢNH</b>	<b>vi</b>
<b>DANH MỤC BẢNG BIỂU</b>	<b>vii</b>
<b>CHƯƠNG 1: TỔNG QUAN</b>	<b>1</b>
1.1    Tổng quan về booking du lịch online . . . . .	1
1.2    Tổng quan về phát triển website . . . . .	1
<b>CHƯƠNG 2: MỘT SỐ CÔNG NGHỆ SỬ DỤNG</b>	<b>3</b>
2.1    Hệ quản trị cơ sở dữ liệu MySQL . . . . .	3
2.2    Logic doanh nghiệp và API . . . . .	3
2.2.1    Ngôn ngữ lập trình Python . . . . .	3
2.2.2    RESTfulAPI và FastAPI . . . . .	3
2.2.3    Cơ chế xác thực OAuth 2.0 . . . . .	4
2.2.4    Xác thực JWT kết hợp hệ mã hóa bất đối xứng . . . . .	4
2.2.5    Object Relational Mapping . . . . .	6
2.2.6    SQLAlchemy . . . . .	6
2.2.7    Universally Unique Identifier . . . . .	6
2.3    Frontend . . . . .	6
2.3.1    Javascript . . . . .	6
2.3.2    TypeScript . . . . .	7
2.3.3    ReactJS . . . . .	7
2.3.4    Tailwind . . . . .	8
<b>CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG</b>	<b>9</b>

3.1	Phân tích yêu cầu hệ thống . . . . .	9
3.2	Thiết kế hệ thống . . . . .	9
3.2.1	Sơ đồ use case . . . . .	9
3.2.2	Đặc tả các use case chính của hệ thống . . . . .	12

## **CHƯƠNG 4: THIẾT KẾ CƠ SỞ DỮ LIỆU 16**

4.1	Mô hình thực thể liên kết . . . . .	16
4.1.1	Xác định thực thể . . . . .	16
4.1.2	Xác định các mối quan hệ . . . . .	19
4.1.3	Xác định các thực thể liên kết . . . . .	23
4.2	Mô hình dữ liệu quan hệ . . . . .	26

## **CHƯƠNG 5: PHÁT TRIỂN HỆ THỐNG 29**

5.1	Xử lý logic . . . . .	29
5.1.1	Xác minh và phân quyền người dùng . . . . .	29
5.1.2	Thiết kế API . . . . .	30
5.1.3	Tương tác với cơ sở dữ liệu . . . . .	30
5.2	Giao diện hệ thống . . . . .	30
5.2.1	Trang web dành cho user . . . . .	30
5.2.2	Trang web dành cho admin . . . . .	36

## **KẾT LUẬN i**

## **TÀI LIỆU THAM KHẢO ii**

## DANH MỤC KÝ HIỆU VÀ TỪ VIẾT TẮT

STT	Ký hiệu	Giải thích
1	UI/UX	User Interface / User Experience
2	HTML	HyperText Markup Language
3	CSS	Cascading Style Sheets
4	PHP	Hypertext Preprocessor
5	GPL	General Public License
6	REST	REpresentational State Transfer
7	JWT	JSON Web Token
8	JWT	JSON Web Token
9	ORM	Object-Relational Mapping
10	OAuth	Open Authorization

# DANH MỤC HÌNH VẼ

Hình 2.1	Cấu trúc của JWT . . . . .	5
Hình 3.2	Sơ đồ use case tổng quát . . . . .	10
Hình 3.3	Sơ đồ chi tiết use case khách hàng . . . . .	10
Hình 3.4	Sơ đồ chi tiết use case quản lý bookings . . . . .	11
Hình 3.5	Sơ đồ chi tiết use case quản lý các dịch vụ du lịch . . . . .	11
Hình 3.6	Sơ đồ chi tiết use case quản lý nhà cung cấp . . . . .	12
Hình 3.7	Sơ đồ chi tiết use case quản lý tài khoản người dùng . . . . .	12
Hình 4.8	Quan hệ giữa user_account và user_profile . . . . .	19
Hình 4.9	Quan hệ giữa user_account và user_card . . . . .	20
Hình 4.10	Quan hệ giữa user_account và user_card . . . . .	20
Hình 4.11	Quan hệ giữa user_account và hotel_service . . . . .	20
Hình 4.12	Quan hệ giữa user_account và flight_ticket . . . . .	20
Hình 4.13	Quan hệ giữa user_account và tour . . . . .	21
Hình 4.14	Quan hệ giữa user_account và tour . . . . .	21
Hình 4.15	Quan hệ giữa user_account và flight_provider . . . . .	21
Hình 4.16	Quan hệ giữa user_account và flight_provider . . . . .	21
Hình 4.17	Quan hệ giữa hotel và hotel_service . . . . .	22
Hình 4.18	Quan hệ giữa flight và flight_ticket . . . . .	22
Hình 4.19	Quan hệ giữa flight_provider và flight . . . . .	22
Hình 4.20	Cơ sở dữ liệu của hệ thống . . . . .	26
Hình 4.21	Các bảng của User . . . . .	27
Hình 4.22	Các bảng liên quan đến Hotel . . . . .	27
Hình 4.23	Các bảng liên quan đến Flight . . . . .	28
Hình 4.24	Các bảng liên quan đến Tour . . . . .	28
Hình 5.25	Quy trình xác thực JWT . . . . .	29
Hình 5.26	Giao diện trang chủ . . . . .	31
Hình 5.27	Giao diện dark mode . . . . .	31

<i>Hình 5.28</i> Chức năng tìm kiếm khách sạn . . . . .	32
<i>Hình 5.29</i> Chức năng tìm kiếm tour . . . . .	32
<i>Hình 5.30</i> Chức năng tìm kiếm chuyến bay . . . . .	33
<i>Hình 5.31</i> Đăng nhập . . . . .	33
<i>Hình 5.32</i> Đăng ký . . . . .	34
<i>Hình 5.33</i> Xác nhận và thanh toán . . . . .	34
<i>Hình 5.34</i> Hoàn thành thanh toán . . . . .	35
<i>Hình 5.35</i> Chính sửa hồ sơ . . . . .	35
<i>Hình 5.36</i> Thay đổi mật khẩu . . . . .	36
<i>Hình 5.37</i> Giao diện quản lý khách sạn . . . . .	36
<i>Hình 5.38</i> Giao diện quản lý tour . . . . .	37
<i>Hình 5.39</i> Giao diện quản lý chuyến bay . . . . .	37
<i>Hình 5.40</i> Giao diện quản lý đặt phòng khách sạn . . . . .	38
<i>Hình 5.41</i> Giao diện quản lý đặt vé máy bay . . . . .	38
<i>Hình 5.42</i> Giao diện quản lý người dùng . . . . .	39

## DANH MỤC BẢNG BIỂU

<i>Bảng 4.1</i>	<i>Thực thể "user_account"</i>	16
<i>Bảng 4.2</i>	<i>Thực thể "user_card"</i>	16
<i>Bảng 4.3</i>	<i>Thực thể "user_profile"</i>	17
<i>Bảng 4.4</i>	<i>Thực thể "hotel"</i>	17
<i>Bảng 4.5</i>	<i>Thực thể "hotel_service"</i>	18
<i>Bảng 4.6</i>	<i>Thực thể "flight_provider"</i>	18
<i>Bảng 4.7</i>	<i>Thực thể "flight"</i>	18
<i>Bảng 4.8</i>	<i>Thực thể "flight_ticket"</i>	19
<i>Bảng 4.9</i>	<i>Thực thể "tour"</i>	19
<i>Bảng 4.10</i>	<i>Thực thể "hotel_booking"</i>	23
<i>Bảng 4.11</i>	<i>Thực thể "user_hotel_rating"</i>	23
<i>Bảng 4.12</i>	<i>Thực thể "flight_booking"</i>	24
<i>Bảng 4.13</i>	<i>Thực thể "user_flight_provider_rating"</i>	24
<i>Bảng 4.14</i>	<i>Thực thể "tour_date"</i>	25
<i>Bảng 4.15</i>	<i>Thực thể "tour_booking"</i>	25
<i>Bảng 4.16</i>	<i>Thực thể "user_tour_rating"</i>	26

# CHƯƠNG 1: TỔNG QUAN

## 1.1 Tổng quan về booking du lịch online

Online Travel Booking là một khái niệm chỉ việc đặt và quản lý các dịch vụ du lịch thông qua internet. Ngành công nghiệp du lịch trực tuyến đã phát triển mạnh mẽ trong thập kỷ qua nhờ sự tiến bộ của công nghệ và sự gia tăng nhu cầu của người tiêu dùng. Thay vì phải đến các đại lý du lịch truyền thống, người dùng có thể dễ dàng tìm kiếm, so sánh và đặt vé máy bay, đặt khách sạn, đặt tour du lịch và các dịch vụ khác chỉ trong vài cú nhấp chuột.

Online Travel Booking mang lại nhiều lợi ích cho cả người tiêu dùng và các nhà cung cấp dịch vụ du lịch. Đối với người tiêu dùng, việc sử dụng các trang web đặt vé và đặt tour giúp tiết kiệm thời gian và công sức, cho phép họ dễ dàng tìm kiếm và so sánh giá cả, đồng thời có thể đặt và thanh toán trực tuyến một cách thuận tiện. Ngoài ra, người dùng còn có thể truy cập vào đánh giá và nhận xét của người dùng khác để có được thông tin chính xác và đáng tin cậy.

Đối với các nhà cung cấp dịch vụ du lịch, trang web đặt vé và đặt tour giúp tăng khả năng tiếp cận khách hàng và tạo ra doanh thu cao hơn. Họ có thể quản lý danh sách sản phẩm, giá cả, lịch trình và khả năng đáp ứng của mình một cách hiệu quả. Ngoài ra, việc tích hợp hệ thống thanh toán an toàn và đáng tin cậy cũng giúp đảm bảo giao dịch được thực hiện một cách an toàn và bảo mật.

## 1.2 Tổng quan về phát triển website

Phát triển website là quá trình tạo ra và xây dựng một trang web hoạt động trên Internet. Quá trình này bao gồm nhiều công việc từ thiết kế giao diện, lập trình, tạo nội dung, triển khai và duy trì. Mục tiêu của việc phát triển website là tạo ra một trang web chất lượng cao, hấp dẫn và chức năng để đáp ứng nhu cầu của người dùng và đạt được mục tiêu kinh doanh của tổ chức hoặc cá nhân.

Quy trình hoạt động của một trang web bao gồm các bước cơ bản sau:

1. Thu thập yêu cầu: Bắt đầu bằng việc thu thập yêu cầu từ khách hàng hoặc tổ chức, hiểu rõ mục tiêu, đối tượng sử dụng, yêu cầu chức năng và giao diện người dùng. Điều này giúp xác định phạm vi và quyết định thiết kế tổng thể của trang web.
2. Thiết kế giao diện (UI/UX Design): Tiếp theo, thiết kế giao diện trực quan và hấp dẫn cho trang web. Bao gồm xác định cấu trúc trang, bố cục, màu sắc, hình ảnh và yếu tố thiết kế khác để tạo ra trải nghiệm người dùng tốt nhất.
3. Phát triển phần mềm: Lập trình tính năng và chức năng cần thiết cho trang

web bằng các ngôn ngữ lập trình như HTML, CSS, JavaScript, PHP, Python, và sử dụng framework phù hợp. Quá trình này tạo ra các trang web tĩnh hoặc động với tính năng tương tác.

4. Tạo nội dung: Tạo nội dung như văn bản, hình ảnh, video và các yếu tố truyền thông khác để truyền đạt thông tin, sản phẩm hoặc dịch vụ của trang web một cách hiệu quả. Nội dung cần phù hợp với mục tiêu và đối tượng sử dụng của trang web.

5. Kiểm thử và debug: Kiểm tra tính năng, tương thích trên các trình duyệt và thiết bị khác nhau, đảm bảo không có lỗi hoạt động và đánh giá tính bảo mật của trang web. Sửa lỗi và cải thiện trang web dựa trên kết quả kiểm thử.

6. Triển khai và duy trì: Triển khai trang web lên máy chủ và cấu hình tên miền để đưa nó lên mạng. Sau đó, duy trì và cập nhật định kỳ để đảm bảo trang web hoạt động tốt, bảo mật và cung cấp nội dung mới nhất cho người dùng.

## CHƯƠNG 2: MỘT SỐ CÔNG NGHỆ SỬ DỤNG

### 2.1 Hệ quản trị cơ sở dữ liệu MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở phổ biến, được phát triển bởi công ty Oracle và được phát hành dưới giấy phép GPL (General Public License). MySQL được sử dụng rộng rãi trên toàn thế giới và là một phần quan trọng trong cơ sở dữ liệu web. MySQL cung cấp một cách linh hoạt và mạnh mẽ các tính năng để quản lý và lưu trữ dữ liệu.. MySQL có khả năng xử lý tải cao, cho phép xử lý hàng nghìn truy vấn đồng thời và hỗ trợ các ứng dụng web có lưu lượng truy cập lớn. Đồng thời, MySQL cũng có khả năng mở rộng linh hoạt và có thể tích hợp với các công nghệ khác như PHP, Java, Python và nhiều ngôn ngữ lập trình khác. Bên cạnh đó, nó có thể chạy trên nhiều hệ điều hành như Windows, Linux và macOS, giúp phát triển ứng dụng đa nền tảng.

### 2.2 Logic doanh nghiệp và API

#### 2.2.1 Ngôn ngữ lập trình Python

Python là một ngôn ngữ lập trình phổ biến cho việc phát triển web do tính linh hoạt, dễ sử dụng và các thư viện và framework mạnh mẽ. Khi nói đến phát triển web bằng Python, có một số framework phổ biến như Django, Flask và Pyramid. Những framework này cung cấp các công cụ và cấu trúc cần thiết để xây dựng ứng dụng web, xử lý định tuyến, quản lý cơ sở dữ liệu và nhiều hơn nữa. Ngoài ra, sự hỗ trợ mạnh mẽ từ cộng đồng Python và tài liệu mở rộng làm cho nó trở thành lựa chọn tuyệt vời cho cả người mới bắt đầu và các nhà phát triển có kinh nghiệm muốn xây dựng ứng dụng web linh hoạt và có khả năng mở rộng.

#### 2.2.2 RESTful API và FastAPI

API là viết tắt của Application Programming Interface (Giao diện lập trình ứng dụng). Thuật ngữ này dùng để chỉ phương thức kết nối từ ứng dụng đến các thư viện khác nhau.

REST là từ viết tắt của REpresentational State Transfer (Truyền trạng thái đại diện). Đây là một tiêu chuẩn dùng để xây dựng API, giúp API trở nên thân thiện hơn với người dùng.

Do đó, RESTful API (hay còn được gọi là REST API) là thuật ngữ dùng để chỉ những API được xây dựng theo cấu trúc REST. Mặc dù REST có thể được sử dụng với mọi giao thức, nhưng nó thường dùng HTTP cho Web API.

FastAPI là một framework web hiện đại, nhanh (hiệu suất cao) được sử dụng để

xây dựng API với Python 3.6+ dựa trên các gợi ý kiểu dữ liệu chuẩn của Python.

FastAPI có các đặc điểm nổi bật như sau:

- Nhanh: Hiệu suất rất cao, ngang bằng với Node.js và Go (nhờ vào Starlette và Pydantic)
- Nhanh chóng khi viết code: Giảm thời gian viết code với các gợi ý kiểu tự động.
- Mạnh mẽ: Có sẵn mã sản xuất với tài liệu tương tác tự động, kiểm tra và kiểm tra kiểu tự động.
- Dựa trên tiêu chuẩn: Dựa trên và hoàn toàn tương thích với các tiêu chuẩn mở cho API (OpenAPI) và JSON Schema.
- WebSockets: Xây dựng ứng dụng thời gian thực với hỗ trợ WebSocket.
- HTTP2: Tận dụng giao thức HTTP/2 để có hiệu suất nhanh hơn
- Bảo mật: Bảo vệ API của bạn bằng xác thực, giới hạn tốc độ và các tính năng bảo mật khác.

### 2.2.3 Cơ chế xác thực OAuth 2.0

OAuth 2.0 (OAuth 2) là một tiêu chuẩn mở cho việc ủy quyền. Nó cho phép người dùng cấp quyền truy cập cho các ứng dụng của bên thứ ba vào dữ liệu của họ mà không cần chia sẻ mật khẩu của họ. OAuth 2 dựa trên khái niệm về "token" được cấp cho các ứng dụng của bên thứ ba bởi nhà cung cấp danh tính của người dùng. Các token này sau đó có thể được sử dụng bởi các ứng dụng để truy cập dữ liệu của người dùng.

### 2.2.4 Xác thực JWT kết hợp mã hóa bất đối xứng

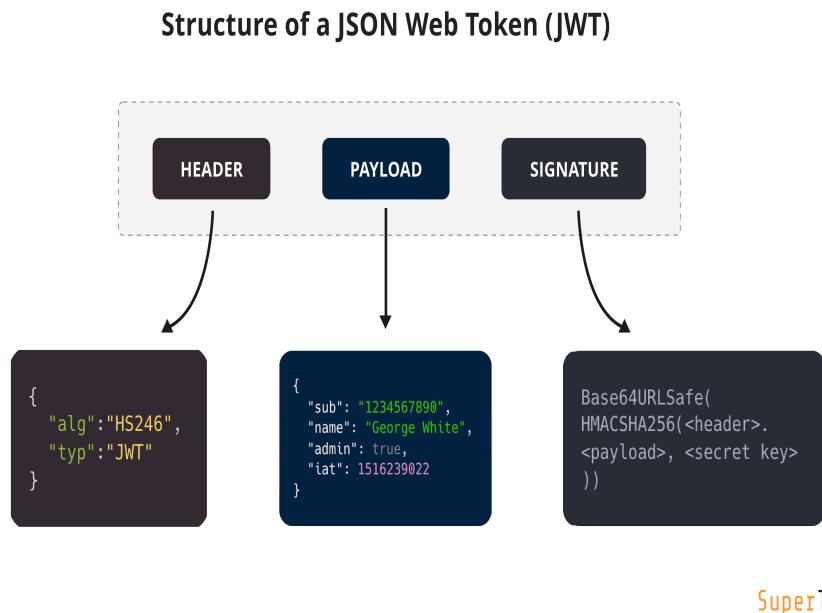
JWT (JSON Web Token) là một cách an toàn để truyền thông tin giữa hai bên dưới dạng một đối tượng JSON. Nó được sử dụng rộng rãi trong các ứng dụng web để xác thực người dùng và cho phép họ truy cập tài nguyên.

JWT thường được sử dụng cho authentication (phân quyền người dùng). Đây là trường hợp phổ biến nhất thường sử dụng JWT. Khi người dùng đã đăng nhập vào hệ thống thì những request tiếp theo từ phía người dùng sẽ chứa thêm mã JWT. Điều này cho phép người dùng được cấp quyền truy cập vào các url, service, và resource mà mã Token đó cho phép.

Cấu trúc của JWT gồm 3 phần:

- Header: chứa kiểu dữ liệu , và thuật toán sử dụng để mã hóa ra chuỗi JWT

- Payload: sẽ chứa các thông tin mà bạn muốn đặt trong chuỗi Token như username , userId , author
- Signature: được tạo ra bằng cách mã hóa phần header , payload kèm theo một chuỗi secret (khóa bí mật)



SuperTokens

Hình 2.1 Cấu trúc của JWT

Một JWT khóa đối xứng sử dụng một cặp khóa mật mã để cung cấp bảo mật tăng cường cho dữ liệu chứa trong mã thông báo. Khác với JWT khóa đối xứng, sử dụng cùng một khóa cho cả mã hóa và giải mã, JWT khóa bất đối xứng sử dụng khóa riêng tư để mã hóa và khóa công khai để xác thực.

JWT khóa bất đối xứng thường được sử dụng trong các kịch bản nơi dữ liệu nhạy cảm cần được truyền và truy cập một cách an toàn bởi nhiều bên. Ví dụ, chúng được rộng rãi áp dụng trong các hệ thống xác thực và ủy quyền, như OAuth 2.0, nơi khóa công khai được sử dụng để xác minh tính xác thực của JWT và khóa riêng tư được sử dụng để giải mã tải dữ liệu.

PyJWT là một thư viện Python được sử dụng để mã hóa và giải mã JSON Web Tokens (JWTs). Nó cung cấp một cách tiện lợi và an toàn để tạo và xác minh JWTs, mà là một cơ chế phổ biến để truyền thông an toàn thông tin giữa các bên. PyJWT giúp việc tạo và xác minh JWT trở nên dễ dàng bằng cách cung cấp một API đơn giản. Để tạo JWT, bạn chỉ cần cung cấp thông tin header và payload, và PyJWT sẽ tự động tạo ra chữ ký. Để xác minh một JWT, bạn chỉ cần cung cấp token và khóa công khai của bên đã ký

nó, và PyJWT sẽ kiểm tra chữ ký để đảm bảo rằng nó hợp lệ.

### **2.2.5 Object Relational Mapping**

Object-Relational Mapping (ORM) là một kỹ thuật lập trình cho phép các nhà phát triển làm việc với cơ sở dữ liệu quan hệ bằng cách sử dụng đối tượng. ORM cung cấp một cách để ánh xạ các đối tượng trong ngôn ngữ lập trình với các bảng và cột trong cơ sở dữ liệu quan hệ. Điều này cho phép các nhà phát triển viết mã theo hướng đối tượng hơn và dễ bảo trì hơn.

### **2.2.6 SQLAlchemy**

SQLAlchemy là một toolkit SQL mạnh mẽ và linh hoạt dành cho Python, cung cấp một bộ công cụ hoàn chỉnh cho các mẫu lưu trữ cấp doanh nghiệp nổi tiếng, được thiết kế để truy cập cơ sở dữ liệu một cách hiệu quả và có hiệu suất cao. Nó sử dụng một phương pháp declarative để định nghĩa cấu trúc và hành vi của bảng và đối tượng cơ sở dữ liệu, cho phép các nhà phát triển tập trung vào logic kinh doanh thay vì cú pháp SQL cơ bản.

Tổng quát, SQLAlchemy là một toolkit SQL Python toàn diện và đa tính năng giúp đơn giản hóa việc truy cập cơ sở dữ liệu và ánh xạ đối tượng-quan hệ, làm cho nó trở thành một lựa chọn phổ biến cho việc xây dựng ứng dụng cơ sở dữ liệu có khả năng mở rộng và hiệu suất cao.

### **2.2.7 Universally Unique Identifier**

Mã định danh duy nhất toàn cầu (UUID) là số 128 bit được sử dụng để nhận dạng duy nhất thông tin. Nó thường được biểu diễn dưới dạng chuỗi thập lục phân gồm 32 ký tự. UUID được sử dụng trong nhiều ứng dụng, bao gồm cơ sở dữ liệu, phát triển phần mềm và hệ thống phân tán, trong đó việc nhận dạng duy nhất các thực thể là rất quan trọng. Chúng được thiết kế độc đáo cả về không gian và thời gian, khiến chúng phù hợp để sử dụng trong các hệ thống phân tán và đảm bảo rằng các mã định danh được tạo ra rất khó bị trùng lặp.

## **2.3 Frontend**

### **2.3.1 Javascript**

JavaScript viết tắt là JS, là ngôn ngữ lập trình phổ biến dùng để tạo ra các trang web tương tác. Được tích hợp và nhúng vào HTML giúp website trở nên sống động hơn. JavaScript đóng vai trò như một phần của trang web, thực thi cho phép Client-Side Script từ phía người dùng cũng như phía máy chủ (Nodejs) tạo ra các trang web động.

JavaScript là một ngôn ngữ lập trình thông dịch với khả năng hướng đến đối tượng. Là một trong 3 ngôn ngữ chính trong lập trình web (HTML, CSS, JavaScript) và có mối

liên hệ lẫn nhau để xây dựng một website sống động, chuyên nghiệp.

Một số chức năng có JavaScript có thể làm như:

1. JavaScript có thể thay đổi nội dung HTML.
2. JavaScript có thể thay đổi giá trị thuộc tính HTML.
3. JavaScript có thể thay đổi kiểu HTML (CSS).
4. JavaScript có thể ẩn các phần tử HTML.
5. JavaScript có thể hiển thị các phần tử HTML.

### 2.3.2 *TypeScript*

TypeScript là một ngôn ngữ lập trình cải tiến JavaScript bằng cách thêm các kiểu loại vào cú pháp. TypeScript thêm cú pháp bổ sung vào JavaScript để các công cụ soạn thảo mã có thể sớm phát hiện những lỗi viết mã. Đồng thời, mã TypeScript chuyển đổi sang JavaScript và cung cấp tất cả các lợi ích tương tự như ở JavaScript. TypeScript cũng chạy trong các ứng dụng và với các khung và thư viện JavaScript.

### 2.3.3 *ReactJS*

ReactJS là một thư viện JavaScript mã nguồn mở phát triển bởi Facebook và cộng đồng các nhà phát triển trên toàn thế giới. Với đặc tính linh hoạt, đơn giản và dễ sử dụng, ReactJS đã trở thành một trong những công nghệ phát triển web được ưa chuộng nhất hiện nay.

ReactJS được phát triển bởi Facebook và được giới thiệu lần đầu tiên vào năm 2011. Ban đầu, ReactJS được phát triển để xây dựng giao diện người dùng trên trang web Facebook, nhằm cải thiện tốc độ và hiệu suất của ứng dụng web.

Tuy nhiên, ReactJS không được công bố cho cộng đồng phát triển cho đến năm 2013, khi Facebook công bố mã nguồn mở của nó và giới thiệu cho cộng đồng lập trình viên. Từ đó, ReactJS nhanh chóng trở thành một trong những thư viện phổ biến nhất để phát triển các ứng dụng web động.

Sau đó, vào năm 2015, Facebook giới thiệu phiên bản React Native, một framework phát triển ứng dụng di động sử dụng ReactJS. React Native cho phép các nhà phát triển xây dựng các ứng dụng di động cho cả iOS và Android sử dụng cùng một mã nguồn, tương tự như ReactJS trên web.

Hiện nay, ReactJS đã trở thành một trong những thư viện phát triển web phổ biến nhất, được sử dụng rộng rãi bởi các công ty lớn và nhỏ trên toàn thế giới. Facebook cũng tiếp tục đầu tư phát triển và nâng cấp ReactJS để đáp ứng nhu cầu của cộng đồng phát triển.

#### **2.3.4 Tailwind**

Tailwind là một utility-first CSS framework giúp chúng ta xây dựng giao diện người dùng một cách nhanh nhất. Tailwind CSS được phát hành lần đầu tiên vào tháng 10/2019 được phát triển bởi Adam Wathan. Nếu so sánh với các nền tảng lập trình CSS khác như Bootstrap, Materialize, Foundation, Material Design... thì giữa chúng có điểm chung với nhau.

Tuy nhiên điểm khác biệt rõ rệt nhất là khung CSS cấp thấp (low-level CSS framework), với khung này sẽ cung cấp cho chúng ta rất nhiều chức năng phát triển thành phần tùy biến mà không nhất thiết phải viết lại CSS.

## CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 3.1 Phân tích yêu cầu hệ thống

Xây dựng trang web cho một hãng, công ty du lịch-lữ hành trung gian thay mặt các khách hàng mua vé máy bay, đặt khách sạn, thiết kế tour bằng cách liên kết với những đơn vị cung cấp các dịch vụ liên quan đến du lịch, nghỉ dưỡng chẳng hạn như công ty hàng không (đặt vé máy bay), khách sạn...

Yêu cầu cụ thể như sau:

Công ty du lịch liên kết với các bên cung cấp dịch vụ bao gồm: các hãng hàng không và khách sạn; thiết kế tour và cập nhật dịch vụ lên website với đầy đủ các thông tin và hình ảnh minh họa.

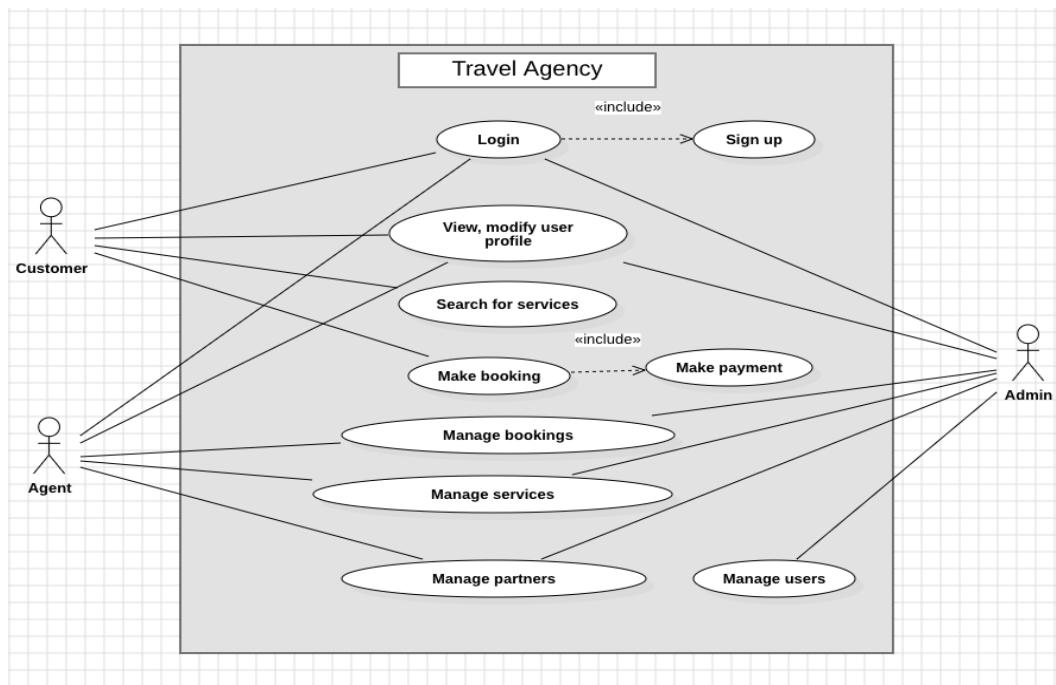
Người dùng có thể truy cập vào website để xem và tìm kiếm thông tin về các dịch vụ du lịch hoặc tìm kiếm dịch vụ theo các yêu cầu cụ thể. Sau đó khách hàng có thể đặt dịch vụ và thực hiện thanh toán.

Các thông tin về dịch vụ, khách hàng, đặt chỗ, thanh toán sẽ được lưu vào cơ sở dữ liệu, cho phép quản trị hệ thống phân quyền thực hiện các công việc thêm, sửa xóa và tìm kiếm thông tin.

### 3.2 Thiết kế hệ thống

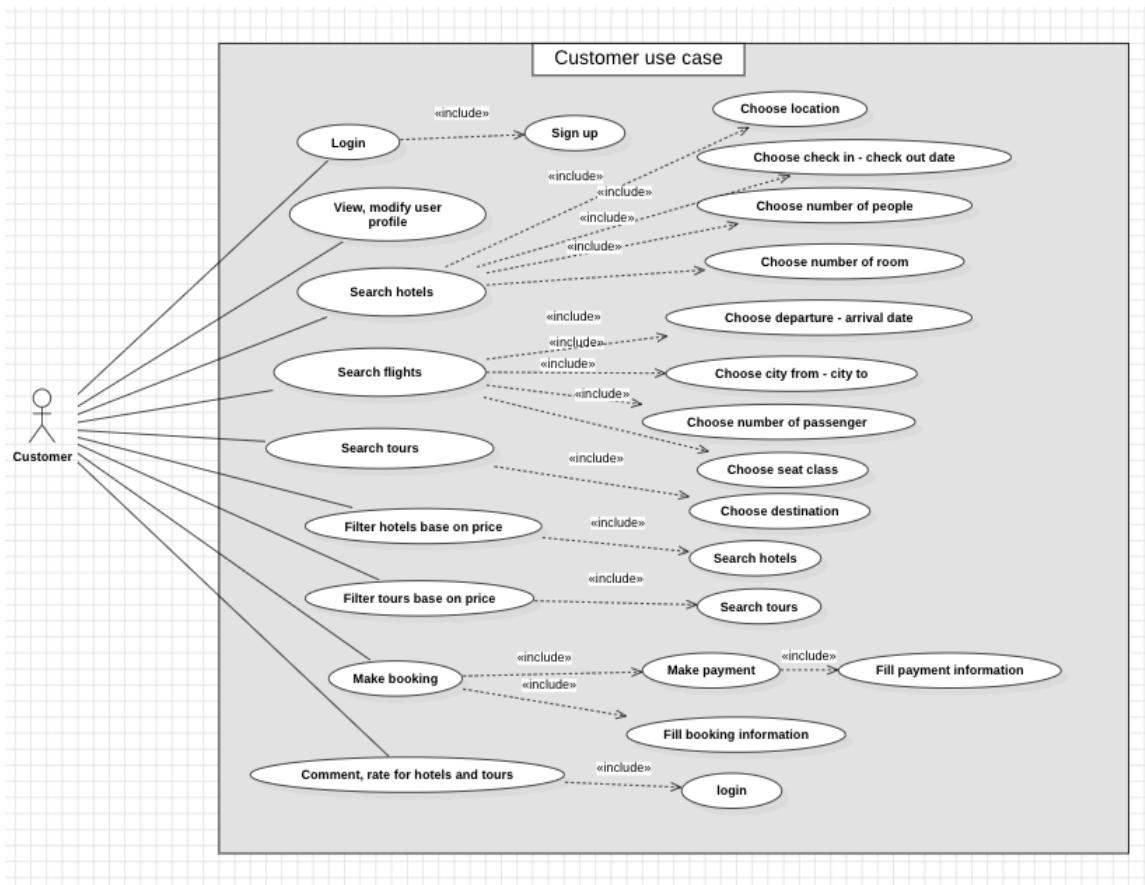
#### 3.2.1 Sơ đồ use case

Sơ đồ use case tổng quát của hệ thống:



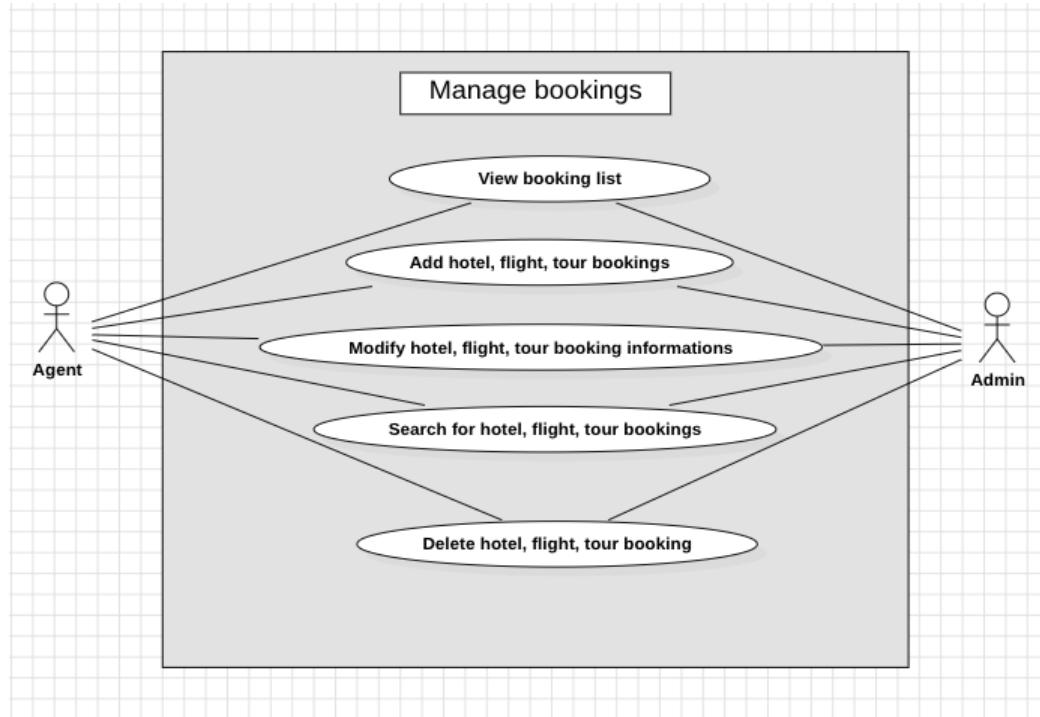
Hình 3.2 Sơ đồ use case tổng quát

Chi tiết các use case của khách hàng:



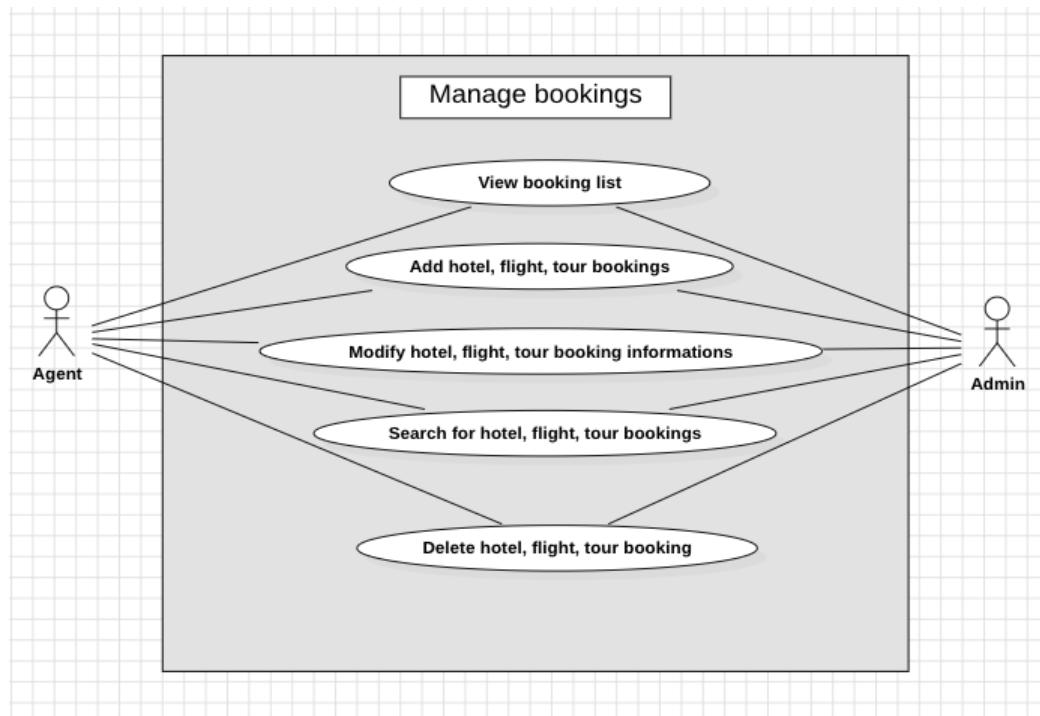
Hình 3.3 Sơ đồ chi tiết use case khách hàng

Chi tiết use case quản lý bookings:



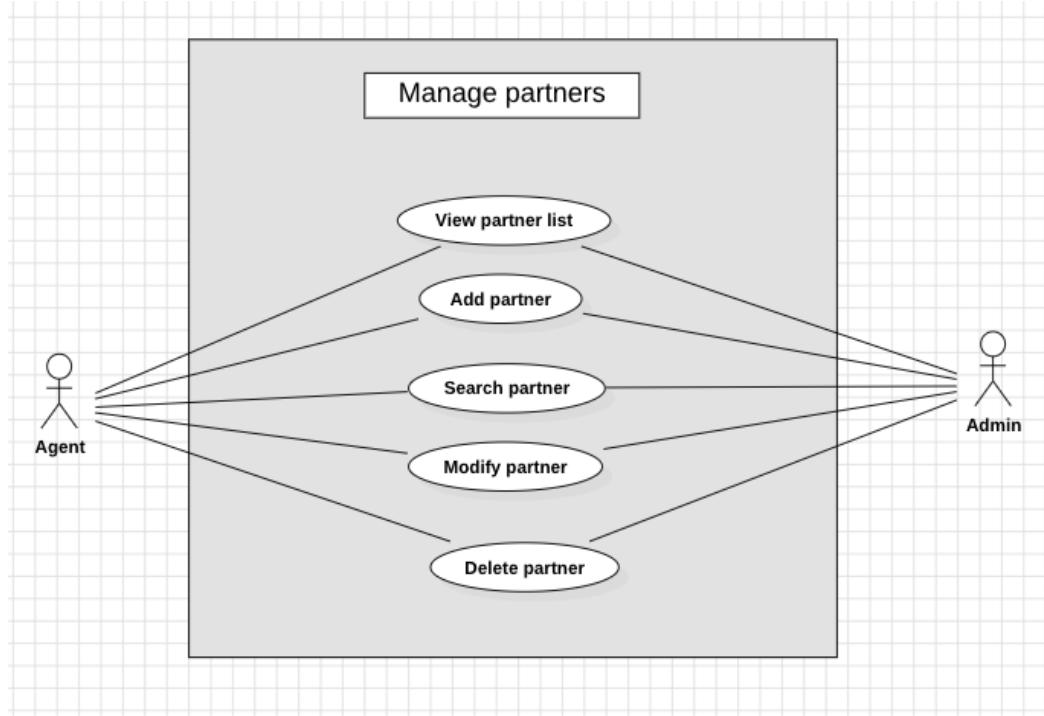
Hình 3.4 Sơ đồ chi tiết use case quản lý bookings

Chi tiết use case quản lý các dịch vụ du lịch:



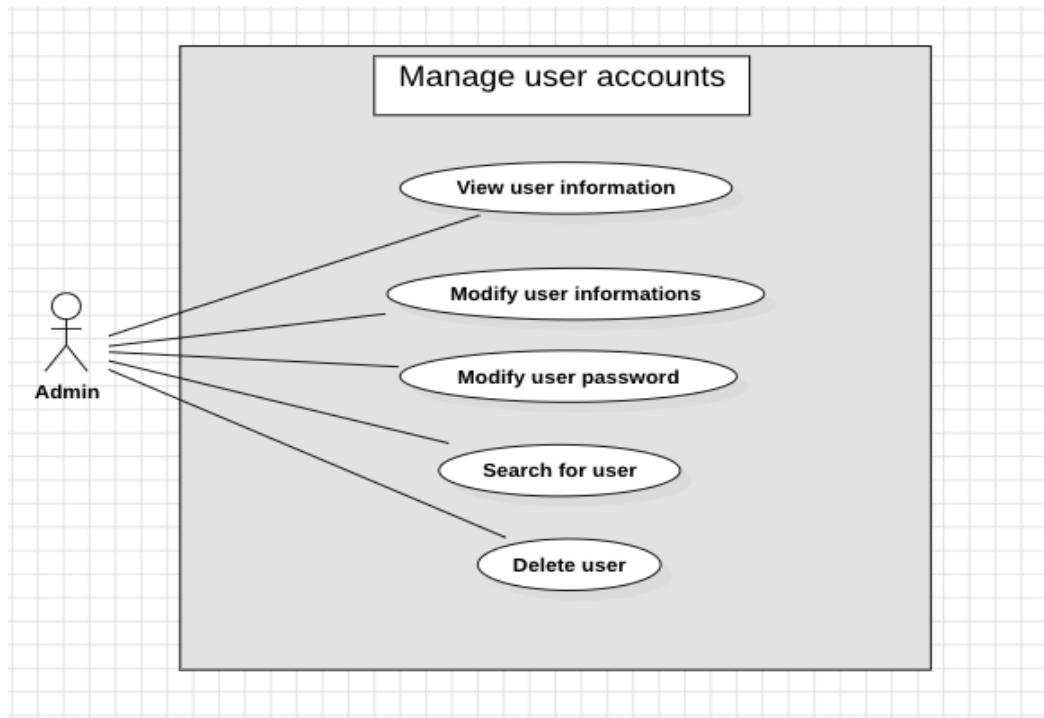
Hình 3.5 Sơ đồ hi tiết use case quản lý các dịch vụ du lịch

Chi tiết use case quản lý nhà cung cấp dịch vụ du lịch:



Hình 3.6 Sơ đồ chi tiết use case quản lý nhà cung cấp

Chi tiết use case quản lý tài khoản người dùng:



Hình 3.7 Sơ đồ chi tiết use case quản lý tài khoản người dùng

### 3.2.2 Đặc tả các use case chính của hệ thống

**Use case “Login”**

- Actor: Customer, Agent, Admin
- Mô tả: Cho phép người dùng đăng nhập vào hệ thống
- Pre-conditions:
  - Hệ thống đang ở trạng thái hoạt động
  - Người dùng đã được cấp tài khoản
- Post-conditions:
  - Người dùng đăng nhập thành công vào hệ thống
  - Hiển thị giao diện hệ thống tương ứng với quyền tài khoản
- Luồng sự kiện chính:

Hành động của Actor	Phản hồi của hệ thống
Chọn mục "Login" trên website	Hiển thị giao diện đăng nhập
Nhập tài khoản, mật khẩu được cung cấp	Kiểm tra tài khoản và mật khẩu đã đăng nhập <ul style="list-style-type: none"> <li>- Nếu không hợp lệ thì báo lỗi và yêu cầu đăng nhập lại</li> <li>- Nếu hợp lệ thì kiểm tra quyền truy cập</li> </ul>

### ***Use case “Sign up”***

- Actor: Customer, Agent
- Mô tả: Cho phép khách hàng đăng ký tài khoản
- Pre-conditions: Khách hàng truy cập vào hệ thống
- Luồng sự kiện chính:

Hành động của Actor	Phản hồi của hệ thống
Chọn mục "Sign up" trên website	Hiển thị giao diện đăng ký
Nhập thông tin: SĐT, tạo mật khẩu	Gán quyền truy cập cho người dùng và lưu SĐT, mật khẩu vào hệ thống

### ***Use case “Search for services”***

- Actor: Customer, Agent, Admin
- Mô tả: Cho phép khách hàng tìm kiếm các dịch vụ du lịch như hotels, flights, tours
- Pre-conditions: Khách hàng truy cập vào hệ thống
- Luồng sự kiện chính:

Hành động của Actor	Phản hồi của hệ thống
Điền / Chọn các thông tin tìm kiếm trên thanh tìm kiếm	Trả về thông tin dịch vụ tương ứng thỏa mãn yêu cầu
Chọn dịch vụ du lịch (hotel, flight, tour)	Hiển thị thông tin chi tiết của dịch vụ

### ***Use case “Make booking”***

- Actor: Customer
- Mô tả: Cho phép người dùng thực hiện booking
- Pre-conditions:
  - Khách hàng truy cập vào hệ thống
  - Đã bấm chọn dịch vụ du lịch (hotel, flight,tour)
- Luồng sự kiện chính:

Hành động của Actor	Phản hồi của hệ thống
Chọn “Book now” bên cạnh dịch vụ muốn book	Hiển thị giao diện booking
Điền các thông tin booking và bấm “Complete”	Hệ thống kiểm tra các thông tin booking <ul style="list-style-type: none"><li>- Nếu hợp lệ thì lưu vào database và hiện giao diện “Confirm”</li><li>- Nếu không hợp lệ thì báo lỗi và yêu cầu điền lại</li></ul>

### ***Use case "Make payment"***

- Actor: Customer
- Mô tả: Cho phép người dùng thực hiện booking
- Pre-conditions:
  - Khách hàng truy cập vào hệ thống
  - Hoàn thành bước booking
- Luồng sự kiện chính:

<b>Hành động của Actor</b>	<b>Phản hồi của hệ thống</b>
Sau khi bấm "Confirm" sau bước booking	Hiển thị giao diện payment
Điền các thông tin trong phiếu payment	<p>Hệ thống kiểm tra các thông tin payment</p> <ul style="list-style-type: none"><li>- Nếu hợp lệ thì lưu vào database và hiện giao diện "Complete"</li><li>- Nếu không hợp lệ thì báo lỗi và yêu cầu điền lại</li></ul>

## CHƯƠNG 4: THIẾT KẾ CƠ SỞ DỮ LIỆU

### 4.1 Mô hình thực thể liên kết

#### 4.1.1 Xác định thực thể

user_account	
<b>Mô tả:</b> Mỗi thực thể đặc trưng cho một tài khoản người dùng	
<b>id</b>	Thuộc tính khoá phân biệt người dùng này với người dùng khác
<b>phone_number</b>	Số điện thoại
<b>password</b>	Mật khẩu
<b>authorization</b>	Phân quyền tài khoản (customer, agent, admin)

Bảng 4.1 Thực thể “user\_account”

user_card	
<b>Mô tả:</b> Mỗi thực thể đặc trưng cho một thẻ thanh toán của người dùng	
<b>id</b>	Thuộc tính khoá phân biệt số tài khoản này với số tài khoản khác
<b>card_number</b>	Số tài khoản
<b>name_on_card</b>	Tên trên thẻ thanh toán
<b>card_cvv</b>	Mã xác minh thẻ Visa
<b>expire_date</b>	Ngày hết hạn của thẻ
<b>user_id</b>	Thuộc tính khoá ngoại cho biết thẻ thanh toán này là của người dùng nào

Bảng 4.2 Thực thể “user\_card”

user_profile	
<b>Mô tả:</b> Mỗi thực thể chứa thông tin cá nhân chi tiết của một người dùng	
<b>id</b>	Thuộc tính khoá phân biệt profile này với profile khác
<b>first_name</b>	Họ và tên đệm
<b>last_name</b>	Tên
<b>gender</b>	Giới tính
<b>dat_of_birth</b>	Ngày sinh
<b>email</b>	Email người dùng
<b>street</b>	Địa chỉ cụ thể
<b>city</b>	Thành phố người dùng đang sinh sống
<b>zip_code</b>	Mã bưu chính dùng để khai báo thông tin
<b>user_id</b>	Thuộc tính khoá ngoại cho biết profile này là của người dùng nào

Bảng 4.3 Thực thể "user\_profile"

hotel	
<b>Mô tả:</b> Mỗi thực thể đặc trưng cho một khách sạn	
<b>id</b>	Thuộc tính khoá phân biệt khách sạn này với khách sạn khác
<b>name</b>	Tên khách sạn
<b>phone_number</b>	Cho biết số điện thoại của khách sạn
<b>description</b>	Cho biết mô tả về khách sạn
<b>address</b>	Cho biết địa chỉ đầy đủ của khách sạn
<b>city</b>	Tên thành phố mà khách sạn toạ lạc
<b>zip_code</b>	Mã bưu điện của địa chỉ khách sạn

Bảng 4.4 Thực thể "hotel"

hotel_service	
<b>Mô tả:</b> Mỗi thực thể đại diện cho một loại dịch vụ khách sạn	
<b>id</b>	Thuộc tính khoá phân biệt dịch vụ này với dịch vụ khác
<b>name</b>	Cho biết tên dịch vụ
<b>room_type</b>	Cho biết loại phòng
<b>capacity</b>	Cho biết sức chứa tối đa
<b>price</b>	Cho biết giá của dịch vụ
<b>description</b>	Mô tả dịch vụ
<b>hotel_id</b>	Thuộc tính khóa ngoại cho biết dịch vụ được cung cấp bởi khách sạn nào

Bảng 4.5 Thực thể “hotel\_service”

flight_provider	
<b>Mô tả:</b> Mỗi thực thể đại diện cho một hãng cung cấp chuyến bay	
<b>id</b>	Thuộc tính khoá phân biệt hãng hàng không này với hãng hàng không khác
<b>name</b>	Tên hãng hàng không
<b>phone_number</b>	Số điện thoại của hãng hàng không
<b>email</b>	Địa chỉ email của hãng hàng không
<b>description</b>	Mô tả về hãng hàng không

Bảng 4.6 Thực thể “flight\_provider”

flight	
<b>Mô tả:</b> Mỗi thực thể đại diện cho một chuyến bay	
<b>id</b>	Thuộc tính khoá phân biệt chuyến bay này với chuyến bay khác
<b>name</b>	Tên chuyến bay
<b>departure_from</b>	Sân bay cất cánh
<b>departure_to</b>	Sân bay hạ cánh
<b>departure_time</b>	Thời điểm cất cánh
<b>arrival_time</b>	Thời điểm hạ cánh dự kiến
<b>description</b>	Thông tin mô tả chuyến bay
<b>flight_provider_id</b>	Thuộc tính khóa ngoại cho biết chuyến bay được cung cấp bởi hãng hàng không nào

Bảng 4.7 Thực thể “flight”

flight_ticket	
<b>Mô tả:</b> Mỗi thực thể đại diện cho một loại vé máy bay	
<b>id</b>	thuộc tính khoá phân biệt loại vé này với loại vé khác
<b>name</b>	Tên vé máy bay
<b>description</b>	Mô tả
<b>seat_class</b>	Hạng ghế
<b>adult_price</b>	Sân bay hạ cánh
<b>child_price</b>	Giá vé trẻ em
<b>baby_price</b>	Giá vé trẻ em dưới 1 tuổi
<b>flight_id</b>	thuộc tính khóa ngoại cho biết thông tin chuyến bay có trong vé

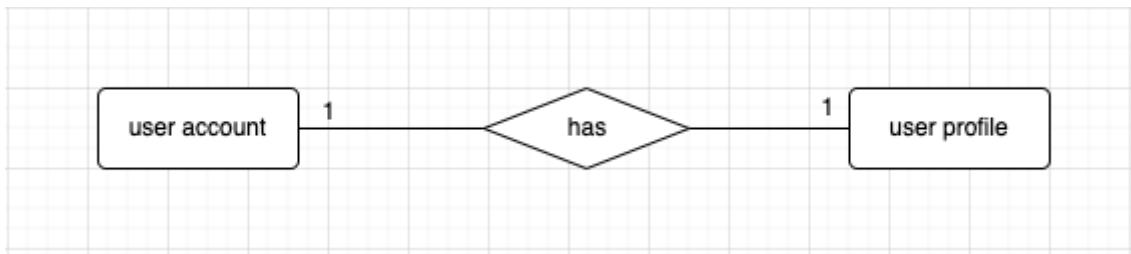
Bảng 4.8 Thực thể “flight\_ticket”

tour	
<b>Mô tả:</b> Mỗi thực thể đại diện cho một tour du lịch	
<b>id</b>	Thuộc tính khoá phân biệt tour này với tour khác
<b>name</b>	Tên tour du lịch
<b>destination</b>	Điểm đến của tour du lịch
<b>description</b>	Mô tả tour du lịch
<b>adult_price</b>	Giá vé người lớn
<b>child_price</b>	Giá vé trẻ em

Bảng 4.9 Thực thể “tour”

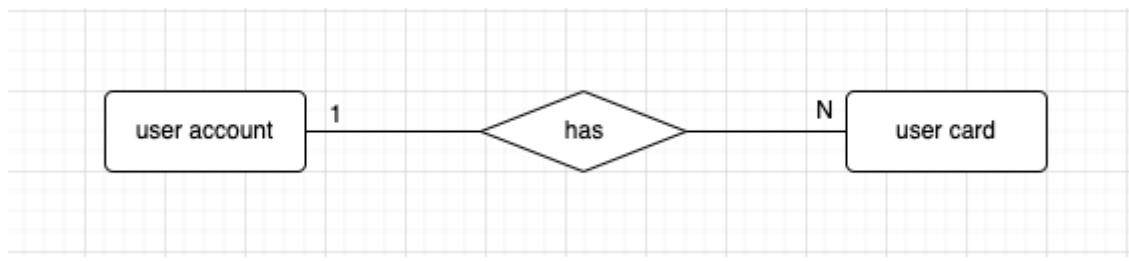
#### 4.1.2 Xác định các mối quan hệ

Quan hệ giữa user\_account và user\_profile:



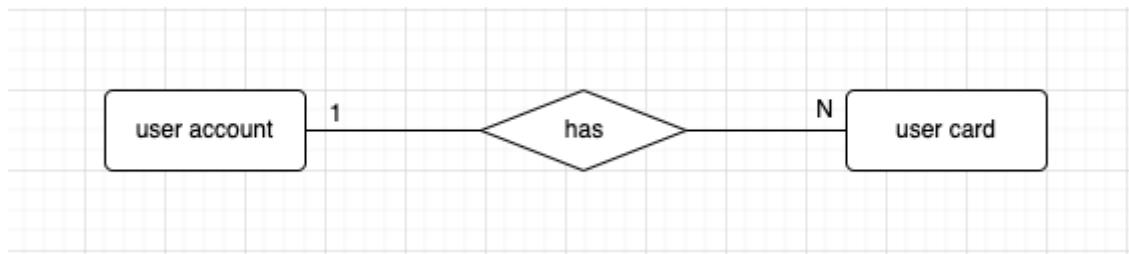
Hình 4.8 Quan hệ giữa user\_account và user\_profile

Quan hệ giữa user\_account và user\_card:



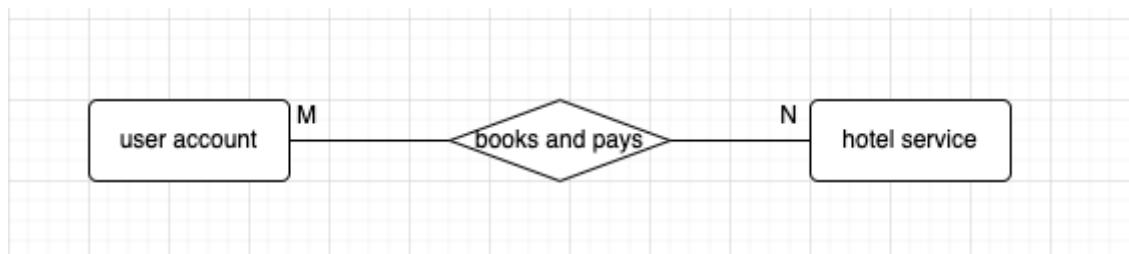
Hình 4.9 Quan hệ giữa user\_account và user\_card

Quan hệ giữa user\_account và user\_card:



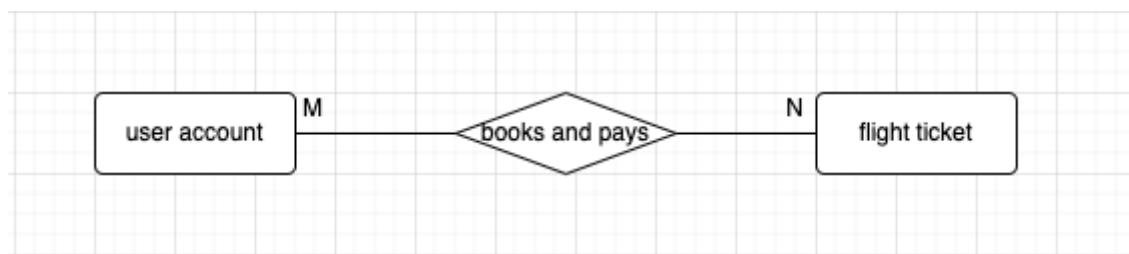
Hình 4.10 Quan hệ giữa user\_account và user\_card

Quan hệ giữa user\_account và hotel\_service:



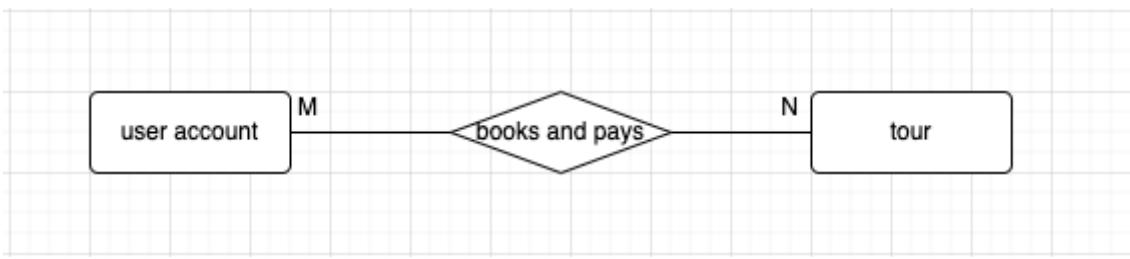
Hình 4.11 Quan hệ giữa user\_account và hotel\_service

Quan hệ giữa user\_account và flight\_ticket:



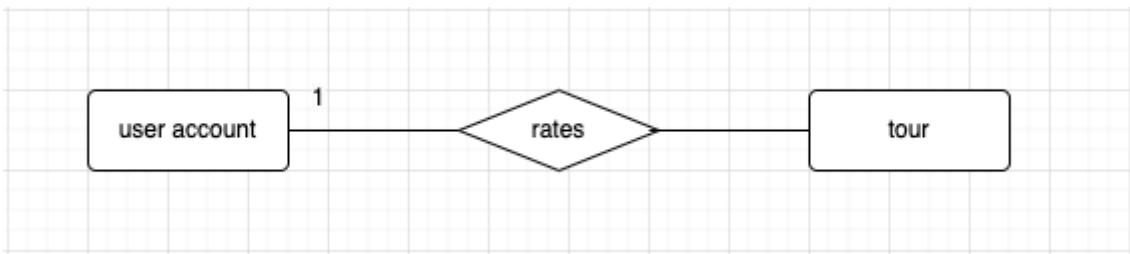
Hình 4.12 Quan hệ giữa user\_account và flight\_ticket

Quan hệ giữa user\_account và tour:



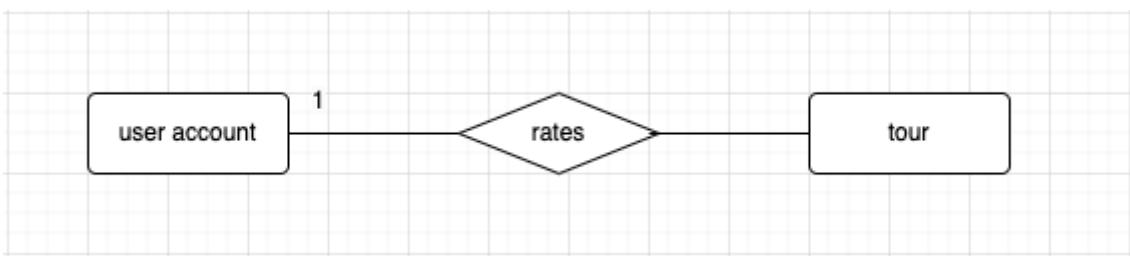
Hình 4.13 Quan hệ giữa user\_account và tour

Quan hệ giữa user\_account và tour:



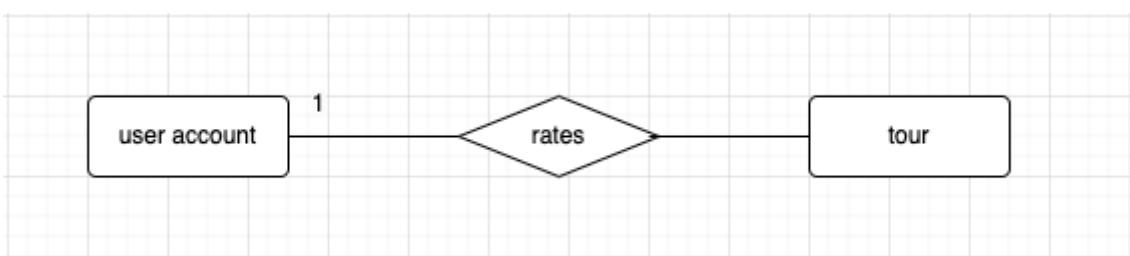
Hình 4.14 Quan hệ giữa user\_account và tour

Quan hệ giữa user\_account và flight\_provider:



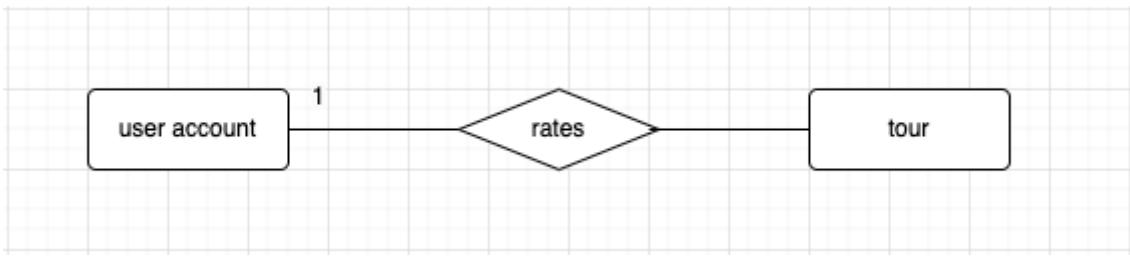
Hình 4.15 Quan hệ giữa user\_account và flight\_provider

Quan hệ giữa user\_account và flight\_provider:



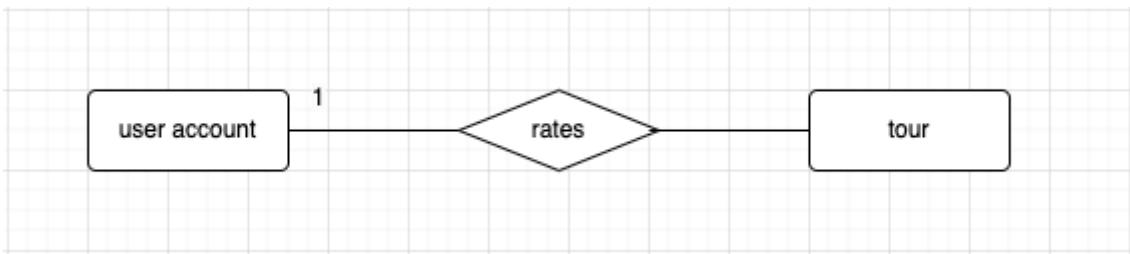
Hình 4.16 Quan hệ giữa user\_account và flight\_provider

Quan hệ giữa hotel và hotel\_service:



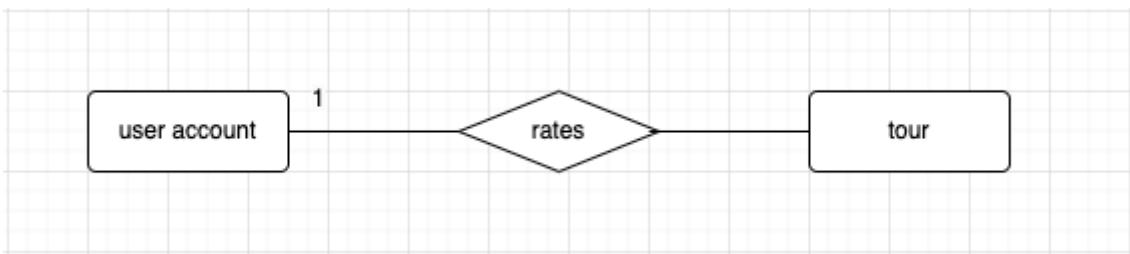
Hình 4.17 Quan hệ giữa hotel và hotel\_service

Quan hệ giữa flight và flight\_ticket:



Hình 4.18 Quan hệ giữa flight và flight\_ticket

Quan hệ giữa flight\_provider và flight:



Hình 4.19 Quan hệ giữa flight\_provider và flight

#### 4.1.3 Xác định các thực thể liên kết

hotel_booking	
<b>Mô tả:</b> Mỗi thực thể đặc trưng cho một lần đặt dịch vụ khách sạn	
<b>id</b>	Thuộc tính khoá phân biệt booking này với booking khác
<b>user_id</b>	Thuộc tính khoá ngoại cho biết người dùng nào đã thực hiện booking
<b>hotel_service_id</b>	Thuộc tính khoá ngoại cho biết dịch vụ khách sạn mà khách hàng muốn đặt
<b>check_in</b>	Thời gian check in khách sạn
<b>check_out</b>	Thời gian check out khách sạn
<b>user_card_id</b>	Số tài khoản người dùng dùng để thanh toán cho booking này
<b>payment_time</b>	Thời gian thanh toán
<b>payment_status</b>	Trạng thái thanh toán (chưa thanh toán/ đã thanh toán/ chờ thanh toán)
<b>payment_description</b>	Mô tả thanh toán

Bảng 4.10 Thực thể “hotel\_booking”

user_hotel_rating	
<b>Mô tả:</b> Mỗi thực thể đặc trưng cho một lần đánh giá khách sạn	
<b>id</b>	Thuộc tính khoá phân biệt lượt đánh giá này với lượt đánh giá khác
<b>user_id</b>	Thuộc tính khoá ngoại cho biết người dùng nào đã đánh giá lượt đánh giá này
<b>hotel_id</b>	Thuộc tính khoá ngoại khách sạn người dùng đánh giá
<b>rating</b>	Số sao đánh giá
<b>comment</b>	Bình luận của người dùng

Bảng 4.11 Thực thể “user\_hotel\_rating”

<b>flight_booking</b>	
<b>Mô tả:</b> Mỗi thực thể đặc trưng cho một lần đặt vé máy bay	
<b>id</b>	Thuộc tính khoá phân biệt hoá đơn đặt vé máy bay này với hoá đơn đặt vé máy bay khác
<b>user_id</b>	Thuộc tính khoá ngoại cho biết người dùng nào đã thực hiện booking
<b>departure_flight_ticket_id</b>	Thuộc tính khoá ngoại cho biết id của vé máy bay của chuyến bay đi mà khách hàng đã đặt
<b>return_flight_ticket_id</b>	Cho biết id của chuyến bay về nếu đặt vé khứ hồi
<b>number_of_adult</b>	Số lượng người lớn
<b>number_of_child</b>	Số lượng trẻ em
<b>number_of_baby</b>	Số lượng trẻ em dưới 1 tuổi
<b>user_card</b>	Số tài khoản người dùng dùng để thanh toán cho booking này
<b>payment_time</b>	Thời gian thanh toán
<b>payment_status</b>	Trạng thái thanh toán (chưa thanh toán/ đã thanh toán/ chờ thanh toán)
<b>payment_description</b>	Mô tả thanh toán

Bảng 4.12 Thực thể “flight\_booking”

<b>user_flight_provider_rating</b>	
<b>Mô tả:</b> Mỗi thực thể đặc trưng cho một lần đánh giá hàng hàng không	
<b>id</b>	Thuộc tính khoá phân biệt lượt đánh giá này với lượt đánh giá khác
<b>user_id</b>	Thuộc tính khoá ngoại cho biết người dùng nào đã đánh giá lượt đánh giá này
<b>hotel_id</b>	Thuộc tính khoá ngoại cho biết hàng hàng không người dùng đánh giá
<b>rating</b>	Số sao đánh giá
<b>comment</b>	Bình luận của người dùng

Bảng 4.13 Thực thể "user\_flight\_provider\_rating"

tour_date	
<b>Mô tả:</b> Mỗi thực thể đại diện ngày đi, ngày về của một tour mà người dùng có thể đặt	
<b>id</b>	Thuộc tính khoá phân biệt ngày đi - ngày về khác nhau mà người dùng có thể đặt
<b>tour_id</b>	Thuộc tính khoá ngoại cho biết tour mà người dùng muốn đặt
<b>departure_date</b>	Ngày bắt đầu tour
<b>return_date</b>	Ngày trở về

Bảng 4.14 Thực thể “tour\_date”

tour_booking	
<b>Mô tả:</b> Mỗi thực đặc trưng cho một lần đặt dịch vụ khách sạn	
<b>id</b>	Thuộc tính khoá phân biệt booking này với booking khác
<b>user_id</b>	Thuộc tính khoá ngoại cho biết người dùng nào đã thực hiện booking
<b>tour_date_id</b>	Thuộc tính khoá ngoại cho biết ngày khởi hành và ngày trở về của tour
<b>number_of_adult</b>	Số lượng người lớn
<b>number_of_child</b>	Số lượng trẻ em
<b>user_card_id</b>	Số tài khoản người dùng dùng để thanh toán cho booking này
<b>payment_time</b>	Thời gian thanh toán
<b>payment_status</b>	Trạng thái thanh toán (chưa thanh toán/ đã thanh toán/ chờ thanh toán)
<b>payment_description</b>	Mô tả thanh toán

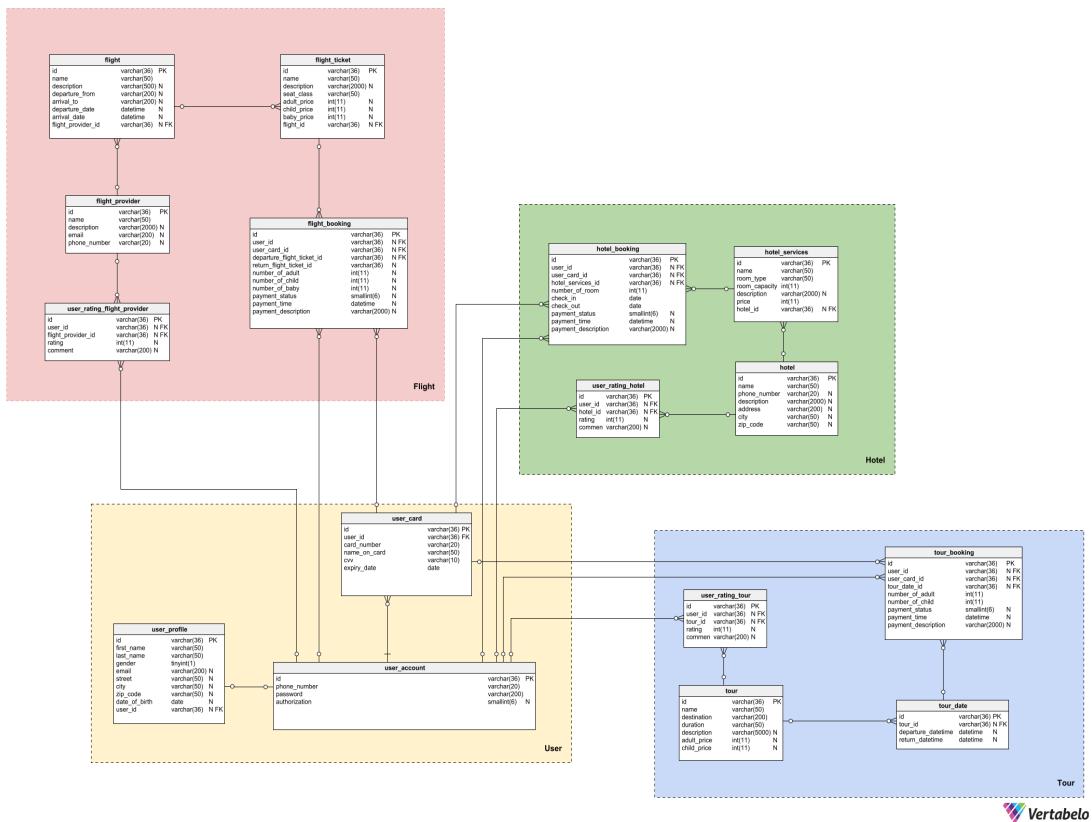
Bảng 4.15 Thực thể “tour\_booking”

user_tour_rating	
<b>Mô tả:</b> Mỗi thực thể đặc trưng cho một lần đánh giá tour không	
<b>id</b>	Thuộc tính khóa phân biệt lượt đánh giá này với lượt đánh giá khác
<b>user_id</b>	Thuộc tính khóa ngoại cho biết người dùng nào đã đánh giá lượt đánh giá này
<b>hotel_id</b>	Thuộc tính khoá ngoại cho biết tour du lịch không người dùng đánh giá
<b>rating</b>	Số sao đánh giá
<b>comment</b>	Bình luận của người dùng

Bảng 4.16 Thực thể "user\_tour\_rating"

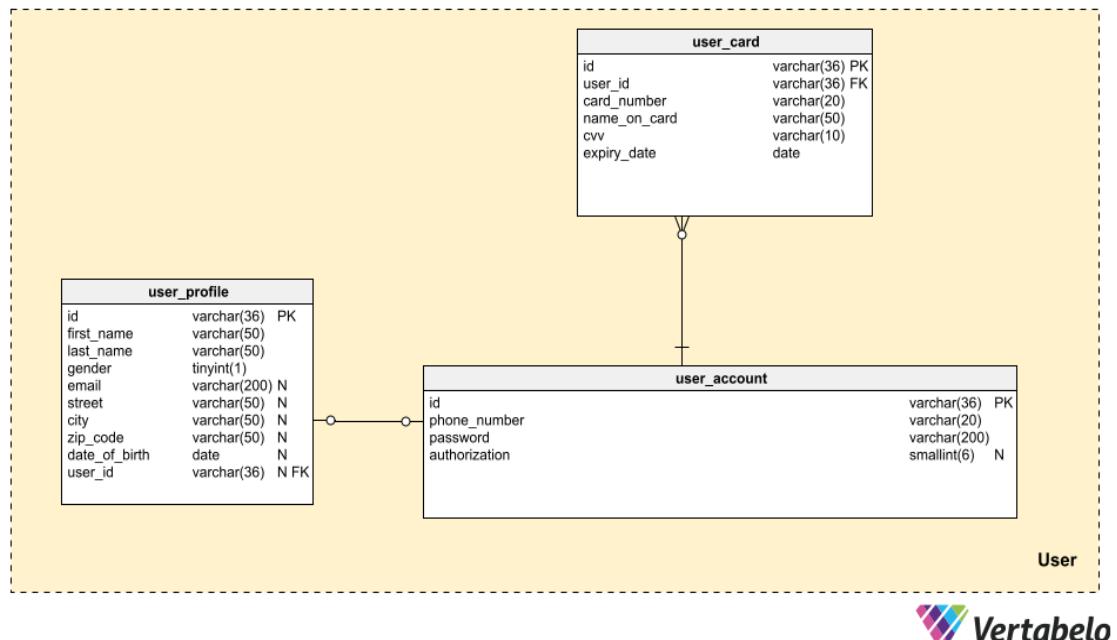
## 4.2 Mô hình dữ liệu quan hệ

Sơ đồ RDD của hệ thống gồm 4 section: User, Hotel, Flight và Tour.



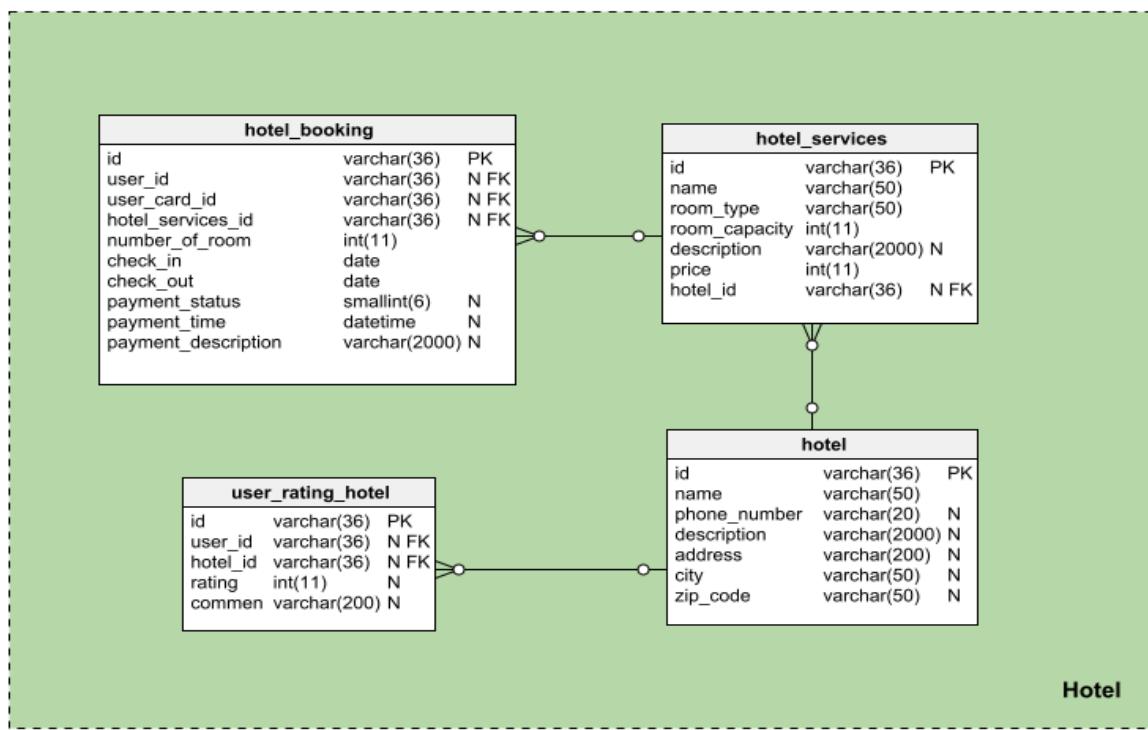
Hình 4.20 Cơ sở dữ liệu của hệ thống

Section 1 bao gồm các bảng liên quan đến thông tin User



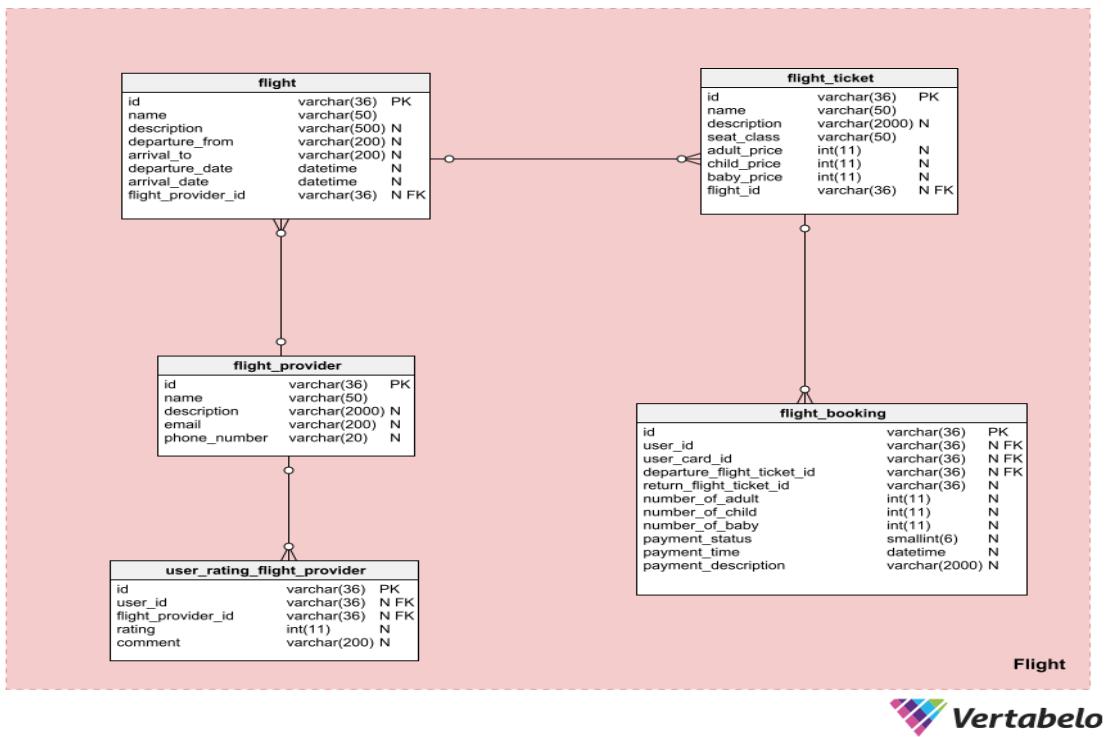
Hình 4.21 Các bảng của user

Section 2 bao gồm các bảng liên quan đến Hotel



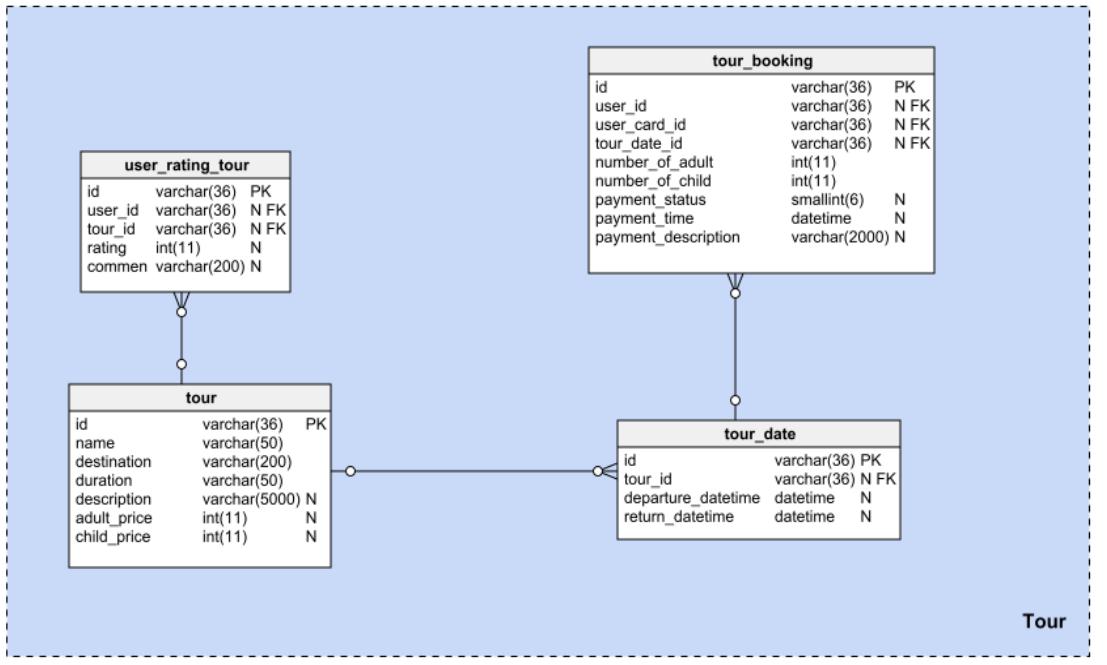
Hình 4.22 Các bảng liên quan đến Hotel

### Section 3 bao gồm các bảng liên quan đến Flight



Hình 4.23 Các bảng liên quan đến Flight

### Section 4 bao gồm các bảng liên quan đến Tour

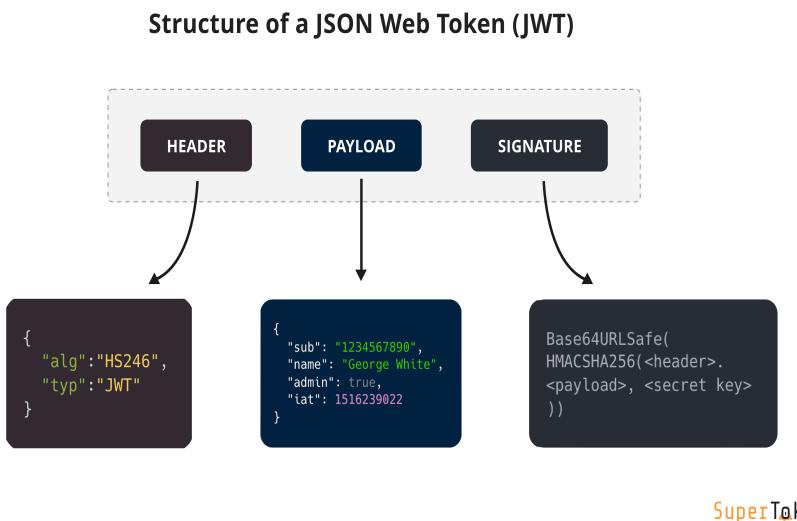


Hình 4.24 Các bảng liên quan đến Tour

# CHƯƠNG 5: PHÁT TRIỂN HỆ THỐNG

## 5.1 Xử lý logic

Website này sử dụng giao thức OAuth2 để xác thực người dùng và cấp quyền truy cập đến tài nguyên mà người dùng hiện đang nắm giữ. Để xác minh người dùng, trang web sử dụng xác minh JWT kết hợp hệ mã hóa bất đối xứng.



Hình 5.25 Quy trình xác thực JWT

Đầu tiên, hệ thống khởi tạo một cặp khóa RSA bao gồm private key và public key. Hai key này sẽ được lưu trữ an toàn trên server để bảo mật. Khi người dùng đăng nhập, hệ thống nhận được mật khẩu và tên tài khoản của người dùng. Lúc này hệ thống sẽ lấy mật khẩu đã băm trong cơ sở dữ liệu và tiến hành băm mật khẩu nhận từ người dùng để so sánh với mật khẩu đã băm. Cuối cùng, hệ thống sẽ sử dụng private key làm signing key rồi tiến hành mã hóa bằng thuật toán RS256 để tạo jwt. Tiếp theo, khi người dùng yêu cầu truy cập tài nguyên yêu cầu xác thực, hệ thống sẽ giải mã jwt mà người dùng gửi đến và sử dụng public key làm verifying key để xác thực jwt. Khi đó, hệ thống sẽ biết được phạm vi truy cập tài nguyên của người dùng và xác minh được người dùng đó.

### 5.1.1 Xác minh và phân quyền người dùng

Việc thiết kế API dựa theo tiêu chuẩn RESTful API xoay quanh việc kiểm soát tài nguyên của người dùng. Thư viện pedantic sẽ giúp kiểm tra dữ liệu người dùng gửi đến khi gọi API đảm bảo dữ liệu không bị thiếu, không bị sai kiểu dữ liệu và có thể dễ dàng chuyển đổi sang kiểu dữ liệu khác. Các đối tượng được trả về bởi phương thức GET luôn

được chuyển đổi về kiểu dữ liệu json. Việc sử dụng kiểu dữ liệu json để trao đổi giữa người dùng và hệ thống giúp để dàng đọc và xử lý thông tin.

### **5.1.2 Thiết kế API**

Việc thiết kế API dựa theo tiêu chuẩn RESTful API xoay quanh việc kiểm soát tài nguyên của người dùng. Thư viện pedantic sẽ giúp kiểm tra dữ liệu người dùng gửi đến khi gọi API đảm bảo dữ liệu không bị thiếu, không bị sai kiểu dữ liệu và có thể dễ dàng chuyển đổi sang kiểu dữ liệu khác. Các đối tượng được trả về bởi phương thức GET luôn được chuyển đổi về kiểu dữ liệu json. Việc sử dụng kiểu dữ liệu json để trao đổi giữa người dùng và hệ thống giúp để dàng đọc và xử lý thông tin.

Để kiểm soát tổng số đối tượng được trả về trong response, API sử dụng hai tham số là page\_num và page\_size. Đối với API cần yêu cầu xác thực, authentication token luôn được truyền trong header để đảm bảo tính bảo mật.

### **5.1.3 Tương tác với cơ sở dữ liệu**

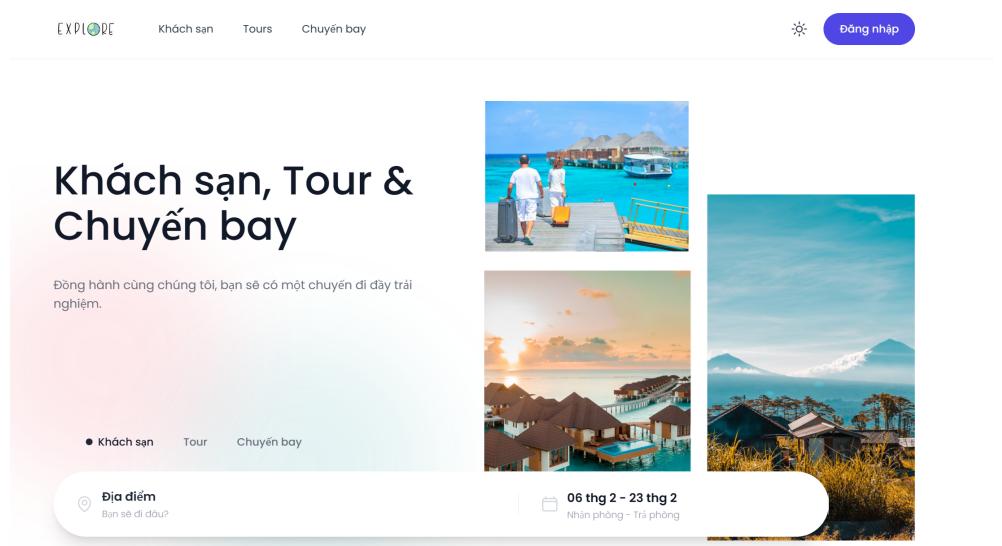
Mỗi khi một API được gọi, một kết nối đến cơ sở dữ liệu sẽ được tạo ra để thao tác với chúng. Thư viện context manager sẽ giúp tạo ra kết nối này và đảm bảo chúng luôn đóng trong mọi trường hợp để tránh hiện tượng quá tải các kết nối đến cơ sở dữ liệu. Thư viện sqlalchemy giúp xây dựng ánh xạ đến cơ sở dữ liệu giúp giảm độ phức tạp của mã và giúp mã dễ dàng bảo trì, nâng cấp. Đồng thời, việc sử dụng sqlalchemy giúp đảm bảo tính vẹn toàn dữ liệu.

Để định danh các đối tượng trong cơ sở dữ liệu, một mã uuid có độ dài 128 bit, 32 ký tự sẽ được sinh tự động. Tiếp theo là về tính bảo mật, khi người dùng tạo tài khoản, mật khẩu sẽ được băm trước sau đó mới lưu trữ an toàn trong cơ sở dữ liệu nhằm giảm thiểu rủi ro lộ mật khẩu.

## **5.2 Giao diện hệ thống**

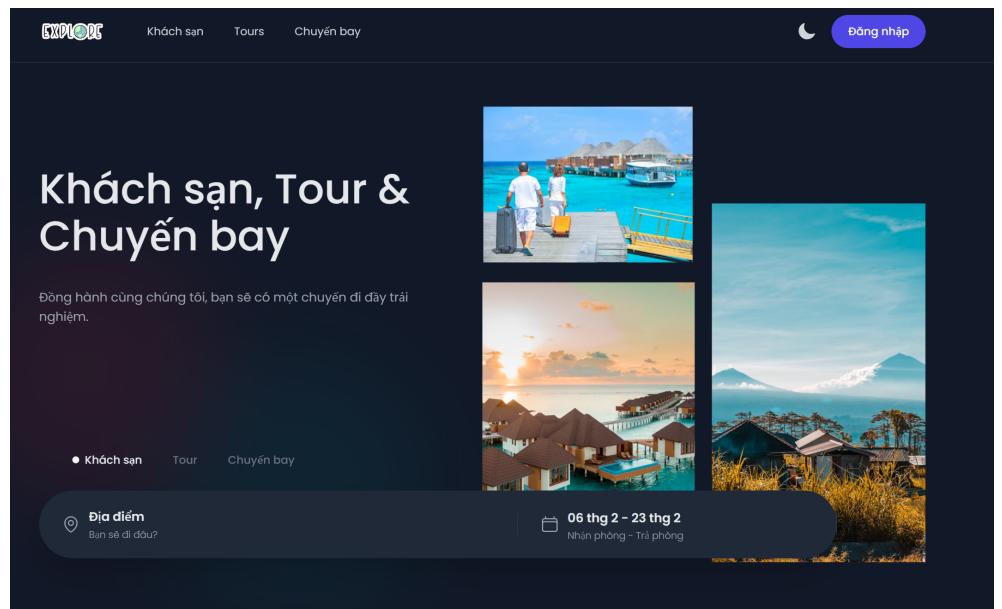
### **5.2.1 Trang web dành cho user**

Trang chủ có thanh điều hướng tới các trang khách sạn, tours và chuyến bay. Ngoài ra còn có thanh tìm kiếm cho từng mục.



Hình 5.26 Giao diện trang chủ

Giao diện darkmode cho trang web.



Hình 5.27 Giao diện darkmode

Chức năng tìm kiếm khách sạn. Có 2 mục đầu vào là điểm đến và ngày nhận phòng, trả phòng. Khi người dùng nhập xong có thể nhấp vào ô tìm kiếm và trả về kết quả mong muốn.

# Thành phố Hồ Chí Minh

Việt Nam 1 khách sạn

● Khách sạn Tour Chuyến bay

Địa điểm  
Bạn sẽ đi đâu?



Hình 5.28 Chức năng tìm kiếm khách sạn

Chức năng tìm kiếm tour. Có 3 mục đầu vào, đó là địa điểm tour, ngày tham quan, và số người tham quan. Sau khi người dùng nhập xong có thể nhấp vào ô tìm kiếm và hệ thống sẽ trả về kết quả mong muốn.

EXPLORE Tour

# Hà Nội

Việt Nam 2 Tours

Khách sạn ● Tour Chuyến bay

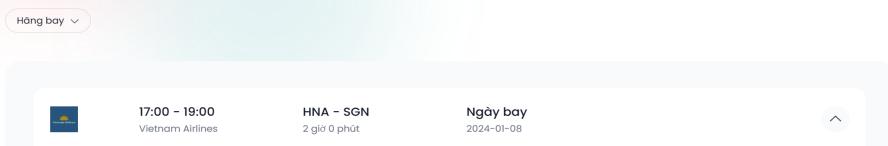
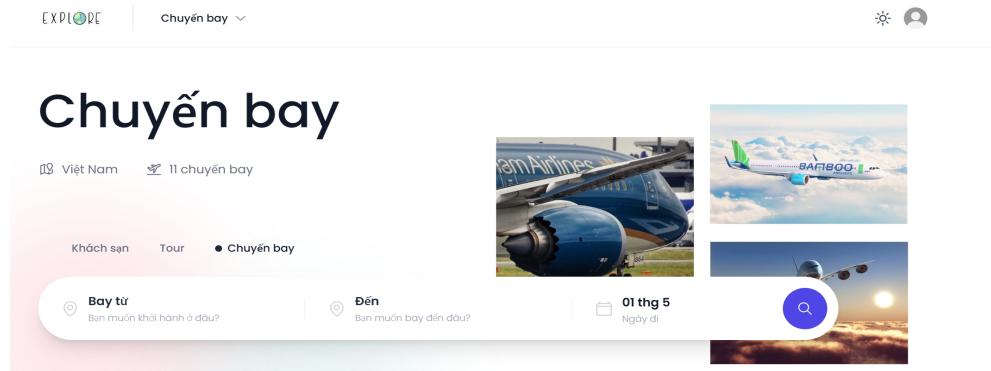
Địa điểm  
Bạn sẽ đi đâu?

01 tháng 3 - 16 tháng 3 Date

4 Người  
Người

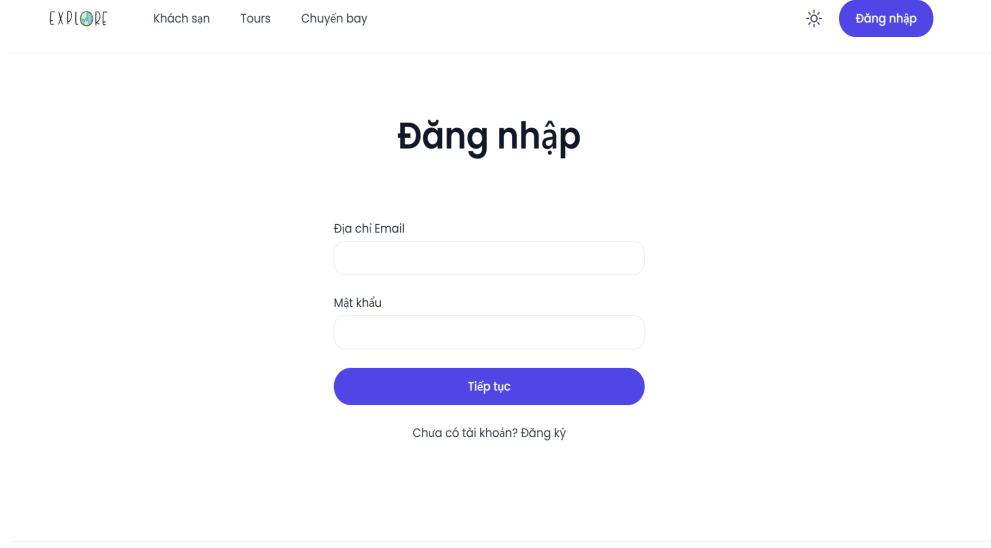
Hình 5.29 Chức năng tìm kiếm tour

Chức năng tìm kiếm chuyến bay. Có 3 mục đầu vào, đó là điểm xuất phát, điểm đến và ngày bay. Sau khi người dùng nhập xong có thể nhấp vào ô tìm kiếm và hệ thống sẽ trả về kết quả mong muốn.



Hình 5.30 Chức năng tìm kiếm chuyến bay

Chức năng đăng nhập. Yêu cầu người dùng nhập địa chỉ email và mật khẩu, sau đó có thể nhấn tiếp tục. Nếu người dùng chưa có tài khoản có thể nhấp vào "đăng ký". Sau khi người dùng đăng nhập xong, hệ thống sẽ tự động quay trở về trang chủ.



Hình 5.31 Đăng nhập

Chức năng đăng ký. Yêu cầu người dùng nhập địa chỉ email, mật khẩu và yêu cầu xác nhận mật khẩu. Sau khi người dùng nhập đúng thông tin, có thể nhấn "tiếp tục". Sau khi đăng ký xong, hệ thống sẽ tự động quay trở về trang chủ.

## Đăng ký

Tiếp tục

Đã có tài khoản? Đăng nhập

*Hình 5.32 Đăng ký*

Chức năng thanh toán. Người dùng có thể chọn 2 phương thức thanh toán là thẻ tín dụng hoặc Paypal và yêu cầu nhập thông tin cần thiết. Phía bên cạnh hiển thị thông tin đặt trước của người dùng. Sau khi kiểm tra kỹ, người dùng có thể nhấn xác nhận và thanh toán.

**Xác nhận và thanh toán**

**Thanh toán bằng**

Credit card 
Paypal

Ngày hết hạn  
mm/dd/yyyy
CVC

Xác nhận và thanh toán

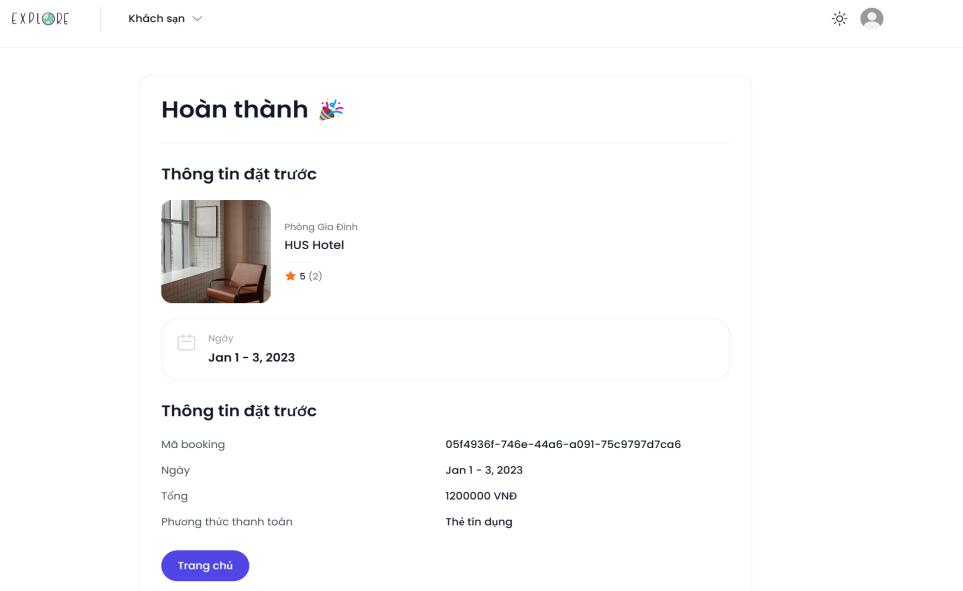


Phòng Gia Định  
**HUS Hotel**  
★ 4.6 (112)

<b>Giá chi tiết</b>	
400000 VNĐ x 3 ngày	1200000 VNĐ
<b>Tổng</b>	<b>1200000 VNĐ</b>

*Hình 5.33 Xác nhận và thanh toán*

Hoàn thành thanh toán. Hệ thống thông báo cho người dùng rằng thanh toán thành công và hiển thị thông tin thanh toán, người dùng có thể quay về trang chủ nếu muốn.



Hình 5.34 Hoàn thành thanh toán

Chỉnh sửa hồ sơ. Người dùng có thể xem và cập nhật hồ sơ của mình nếu muốn.

The screenshot shows a user profile editing page. At the top, there's a header with 'EXPLORE' and 'Khách sạn'. Below the header, there are links for 'Thông tin tài khoản' and 'Thay đổi mật khẩu'. The main section is titled 'Thông tin tài khoản' (Account information) and contains fields for: Tên (Name: Lê Trương Giang), Giới tính (Gender: Nam), Email (Email: letruonggiang7gpl@gmail.com), Ngày sinh (Date of birth: 11/15/2003), Địa chỉ (Address: HN), and Số điện thoại (Phone number: 0123456). A blue 'Cập nhật' (Update) button is located at the bottom of the form.

Hình 5.35 Chỉnh sửa hồ sơ

Thay đổi mật khẩu. Người dùng có thể thay đổi mật khẩu nếu muốn.

Thay đổi mật khẩu

Mật khẩu hiện tại

Mật khẩu mới

Xác nhận mật khẩu

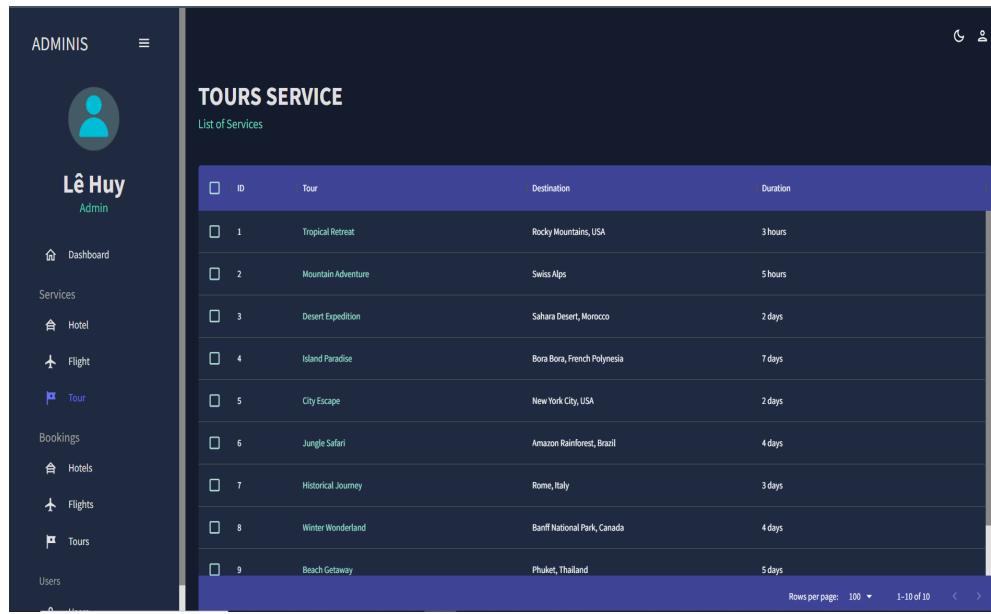
Xác nhận

Hình 5.36 Thay đổi mật khẩu

### 5.2.2 Trang web dành cho admin

ID	Hotel	Address	Phone
1	Khach San A	Ha Noi	(232)545-2345
2	Khach San B	Hai Phong	(232)545-7654
3	Khach San C	Bac Giang	(232)545-7233
4	Khach San D	Hai Duong	(232)545-6483
5	Khach San E	Thanh Hoa	(232)545-1234
6	Khach San F	Quang Ninh	(232)545-9876
7	Khach San G	Da Nang	(232)545-4567
8	Khach San H	Nha Trang	(232)545-8765
9	Khach San I	Can Tho	(232)545-3456

Hình 5.37 Giao diện quản lý khách sạn

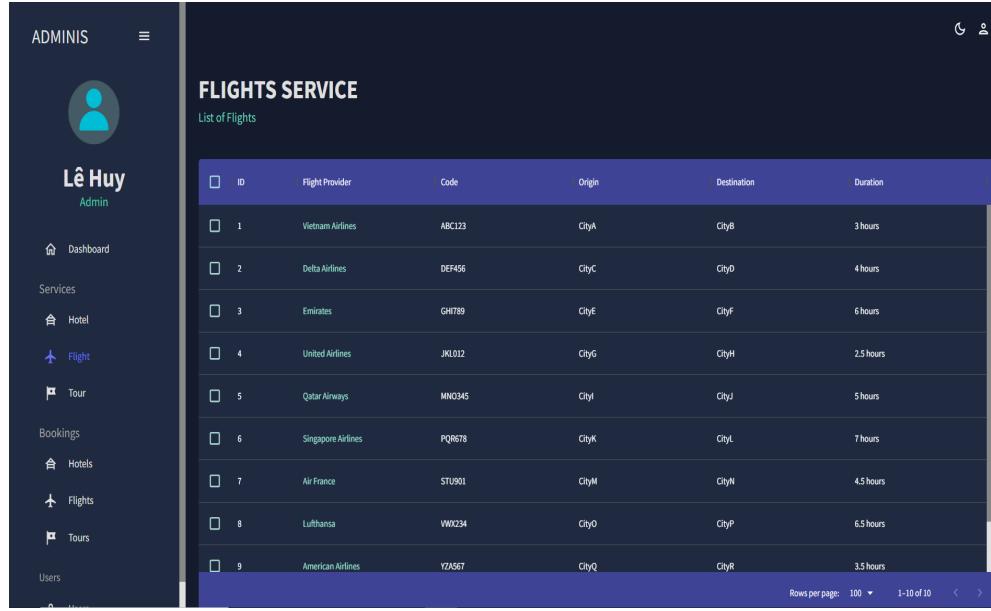


The screenshot shows the Tours Service section of a web application. The left sidebar, titled "ADMINIS", includes a user profile for "Lê Huy Admin" and links for Dashboard, Services (Hotel, Flight, Tour), Bookings (Hotels, Flights, Tours), and Users. The main content area is titled "TOURS SERVICE" and "List of Services". It displays a table with columns: ID, Tour, Destination, and Duration. The data is as follows:

ID	Tour	Destination	Duration
1	Tropical Retreat	Rocky Mountains, USA	3 hours
2	Mountain Adventure	Swiss Alps	5 hours
3	Desert Expedition	Sahara Desert, Morocco	2 days
4	Island Paradise	Bora Bora, French Polynesia	7 days
5	City Escape	New York City, USA	2 days
6	Jungle Safari	Amazon Rainforest, Brazil	4 days
7	Historical Journey	Rome, Italy	3 days
8	Winter Wonderland	Banff National Park, Canada	4 days
9	Beach Getaway	Phuket, Thailand	5 days

At the bottom right of the table, there are buttons for "Rows per page: 100" and "1-10 of 10".

Hình 5.38 Giao diện quản lý tour



The screenshot shows the Flights Service section of a web application. The left sidebar, titled "ADMINIS", includes a user profile for "Lê Huy Admin" and links for Dashboard, Services (Hotel, Flight, Tour), Bookings (Hotels, Flights, Tours), and Users. The main content area is titled "FLIGHTS SERVICE" and "List of Flights". It displays a table with columns: ID, Flight Provider, Code, Origin, Destination, and Duration. The data is as follows:

ID	Flight Provider	Code	Origin	Destination	Duration
1	Vietnam Airlines	ABC123	CityA	CityB	3 hours
2	Delta Airlines	DEF456	CityC	CityD	4 hours
3	Emirates	GHI789	CityE	CityF	6 hours
4	United Airlines	JKL012	CityG	CityH	2.5 hours
5	Qatar Airways	MNO345	CityI	CityJ	5 hours
6	Singapore Airlines	PQR678	CityK	CityL	7 hours
7	Air France	STU901	CityM	CityN	4.5 hours
8	Lufthansa	VWX234	CityO	CityP	6.5 hours
9	American Airlines	YZA567	CityQ	CityR	3.5 hours

At the bottom right of the table, there are buttons for "Rows per page: 100" and "1-10 of 10".

Hình 5.39 Giao diện quản lý chuyến bay

**HOTELS BOOKING**

List of Hotels

ID	Hotel	Hotel Service ID	User Booked	Check In	Check Out	Payment Time
1	Khach San A	101	tuannguyen213@gmail.com	04-01-2023	10-01-2023	03-01-2023 15:30:00
2	Khach San E	101	0923910371	23-04-2023	30-04-2023	20-04-2023 18:32:41
3	Khach San B	102	john.doe@example.com	15-05-2023	20-05-2023	10-05-2023 10:15:00
4	Khach San C	103	alice.smith@example.com	01-07-2023	05-07-2023	25-06-2023 14:45:30
5	Khach San D	104	0987654321	12-08-2023	18-08-2023	08-08-2023 09:20:15
6	Khach San F	105	jane.doe@example.com	09-09-2023	15-09-2023	05-09-2023 17:12:00
7	Khach San G	106	0912345678	02-11-2023	08-11-2023	28-10-2023 20:00:45
8	Khach San H	107	bob.smith@example.com	14-12-2023	20-12-2023	10-12-2023 12:30:20
9	Khach San I	108	0765432109	17-01-2024	23-01-2024	12-01-2024 09:45:10

Hình 5.40 Giao diện quản lý đặt phòng khách sạn

**FLIGHTS BOOKING**

List of Flights

ID	Flight Provider	Departure	Return	User Booked	Payment Time
1	Vietnam Airlines	15-01-2024		tuannguyen213@gmail.com	10-12-2023 12:30:20
2	Air France	25-12-2023	08-01-2024	0512798561	24-12-2023 14:32:11
3	Lufthansa	18-02-2024	25-02-2024	john.doe@example.com	15-01-2024 09:45:55
4	Emirates	02-03-2024		mary.smith@example.com	28-02-2024 16:18:42
5	Delta Airlines	10-04-2024	18-04-2024	1234567890	05-04-2024 11:20:30
6	Qatar Airways	22-05-2024	30-05-2024	jane.doe@example.com	18-05-2024 08:37:15
7	Singapore Airlines	15-06-2024		9876543210	10-06-2024 15:12:40
8	American Airlines	05-07-2024	12-07-2024	alice.johnson@example.com	30-06-2024 10:05:22
9	British Airways	18-08-2024	25-08-2024	0987654321	12-08-2024 13:28:18

Hình 5.41 Giao diện quản lý đặt vé máy bay

ID	Name	Phone Number	Mail	Authorize
1	Nguyen Van A	0831991023	ngvank@gmail.com	1
2	Tran Thi B	0908765432	tranB@example.com	2
3	Le Van C	0987654321	levanC@example.com	3
4	Pham Thi D	0854321098	phamD@example.com	1
5	Hoang Van E	0932108765	hoangE@gmail.com	2
6	Do Thi F	0876543210	doF@example.com	3
7	Truong Van G	0812345678	truongG@gmail.com	1
8	Nguyen Thi H	0945678901	nguyenH@example.com	2
9	Vu Van I	0890123456	vui@gmail.com	3

Hình 5.42 Giao diện quản lý người dùng

## KẾT LUẬN

### Kết luận chung

Trong báo cáo này, nhóm em đã triển khai xây dựng một website phục vụ cho việc du lịch cơ bản bao gồm khách sạn, tour du lịch khám phá và cung cấp chuyến bay bằng cách sử dụng các công nghệ như đã nêu ở chương 2 cũng như các kiến thức đã được dạy trên lớp.

Kết quả cho thấy trang web có thể thực hiện được các tác vụ cơ bản như tìm kiếm khách sạn, tour, chuyến bay, đăng nhập, đăng ký, đặt phòng khách sạn, tour, chuyến bay, quản lý người dùng, quản lý khách sạn, quản lý chuyến bay, cũng như các dịch vụ.

### Hướng phát triển

Nhìn chung website nhóm chúng em triển khai chỉ ở mức độ rất cơ bản so với một trang web hoàn chỉnh, còn nhiều phần có thể cai

## **TÀI LIỆU THAM KHẢO**

[1] FastAPI Documentations

React Documentation

MySQL Documentation