

DẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



Báo Cáo Bài Tập Lớn

Lập Trình Web (CO3049)

Chủ đề: Xây dựng Website Hàng hàng không Claude Airline

Nhóm: 8

GVHD: Nguyễn Hữu Hiếu

SVTH: Phạm Dinh Bằng - 2210298
Trần Minh Quân - 2212822

TP. Hồ Chí Minh, Tháng 5 năm 2025



Phân chia công việc

STT	Họ và tên	MSSV	Nhiệm vụ	Tỉ lệ tham gia đóng góp
1			<ul style="list-style-type: none">- Nghiên cứu trình bày cơ sở lý thuyết phase 1- Vẽ diagrams- Kiểm thử và đánh giá kết quả thu được	100%
2			<ul style="list-style-type: none">- Viết báo cáo phase 1- Hiện thực chương trình qua ngôn ngữ Python- Kiểm thử và đánh giá kết quả thu được	100%



Contents

1 Giới thiệu	3
1.1 Giới thiệu về Công ty/Doanh nghiệp	3
1.2 Mục tiêu xây dựng website	3
2 Cơ sở lý thuyết	4
2.1 Các công nghệ và thư viện sử dụng	4
2.1.1 Frontend	4
2.1.2 Backend	5
2.1.3 Giao diện quản trị (Admin Panel)	6
2.2 Ưu điểm và nhược điểm của từng công nghệ	6
2.3 Các lỗ hổng bảo mật thường gặp trong ứng dụng web	7
2.4 Các yếu tố tối ưu hóa SEO trong ứng dụng	9
3 Thiết kế ứng dụng	10
3.1 Thiết kế cơ sở dữ liệu	10
3.1.1 Entity Relationship Diagram	10
3.1.2 Cấu trúc và chức năng của các bảng dữ liệu	11
3.1.3 Ánh xạ cơ sở dữ liệu thành dạng bảng	14
3.2 Mô hình ứng dụng	15
3.3 Cấu trúc mã nguồn	17
3.4 Các tính năng chính của hệ thống	18
3.4.1 Tính năng người dùng	18
3.4.2 Tính năng dành cho quản trị viên	18
3.4.3 Giao diện người dùng	19
3.5 Các sơ đồ thể hiện các quy trình chính	19
4 Hiện thực	19
4.1 Các tính năng đã hiện thực	19
4.1.1 Tính năng phía Người dùng (Client-side)	19
4.1.2 Tính năng phía Quản trị viên (Admin-side)	20
4.1.3 Tính năng Kỹ thuật và Chung	21
4.2 Demo giao diện	21
4.2.1 Trang Đăng Nhập (Login)	22
4.2.2 Trang Đăng Ký (Register)	23
4.2.3 Trang Chủ (Homepage)	24
4.2.4 Trang Giới Thiệu (About Us)	29
4.2.5 Trang Liên Hệ (Contact Us)	31
4.2.6 Trang Câu Hỏi Thường Gặp (FAQ)	33
5 Cài đặt và triển khai	35
5.1 Yêu cầu môi trường	35
5.2 Hướng dẫn cài đặt ứng dụng	35
5.2.1 Bước 1: Tải mã nguồn	35
5.2.2 Bước 2: Cài đặt và chạy frontend	35
5.2.3 Bước 3: Cài đặt và chạy backend	35



1 Giới thiệu

1.1 Giới thiệu về Công ty/Doanh nghiệp

Claude Airline là một hãng hàng không giả định được thành lập với mục tiêu cung cấp dịch vụ vận chuyển hàng không chất lượng cao, hiện đại và thân thiện với người dùng. Trong bối cảnh ngành hàng không đang ngày càng cạnh tranh khốc liệt và đòi hỏi sự thích nghi nhanh chóng với công nghệ, Claude Airline định hướng trở thành một trong những hãng hàng không tiên phong trong việc số hóa dịch vụ và nâng cao trải nghiệm khách hàng.

Với khẩu hiệu “Bay cùng tương lai”, Claude Airline đặt mục tiêu xây dựng một thương hiệu năng động, trẻ trung, an toàn và luôn đổi mới. Hãng chuyên khai thác các tuyến bay nội địa và quốc tế trong khu vực châu Á – Thái Bình Dương, đồng thời hướng tới mở rộng thị trường ra các châu lục khác trong tương lai gần. Claude Airline chú trọng đến yếu tố con người và dịch vụ, nhằm mang đến trải nghiệm bay thoải mái, tiện nghi cho mọi hành khách, từ khách phổ thông cho đến khách hàng cao cấp.

Claude Airline không chỉ là một hãng hàng không đơn thuần, mà còn mong muốn trở thành cầu nối giữa con người và thế giới, kết nối các vùng miền, quốc gia, văn hóa thông qua những chuyến bay chất lượng và an toàn. Việc đầu tư vào công nghệ và hệ thống số hóa là chiến lược dài hạn để Claude Airline phát triển bền vững trong thời đại 4.0.

1.2 Mục tiêu xây dựng website

Trong thời đại công nghệ số hiện nay, việc sở hữu một website chuyên nghiệp là điều bắt buộc đối với bất kỳ doanh nghiệp nào, đặc biệt là trong lĩnh vực dịch vụ như hàng không. Website không chỉ là kênh cung cấp thông tin mà còn đóng vai trò là nền tảng giao tiếp và giao dịch với khách hàng. Với nhận thức đó, nhóm chúng tôi xây dựng website cho Claude Airline nhằm phục vụ các mục tiêu sau:

- Đặt vé trực tuyến một cách nhanh chóng và thuận tiện:** Khách hàng có thể tìm kiếm chuyến bay, so sánh giá vé, lựa chọn chỗ ngồi và thanh toán ngay trên website. Giao diện thân thiện với người dùng giúp quá trình đặt vé diễn ra mượt mà trên cả máy tính lẫn thiết bị di động.
- Cung cấp đầy đủ thông tin chính thức từ hãng:** Bao gồm thông tin về lịch bay, chính sách hoàn hủy vé, hành lý, dịch vụ bổ sung, chương trình khuyến mãi,... đảm bảo tính minh bạch và chính xác trong quá trình cung cấp dịch vụ.
- Hỗ trợ quản lý tài khoản và chăm sóc khách hàng:** Người dùng có thể đăng ký tài khoản, xem lịch sử đặt vé, cập nhật thông tin cá nhân, gửi yêu cầu hỗ trợ hoặc đóng góp ý kiến ngay trên hệ thống.
- Hướng đến tối ưu hóa trải nghiệm người dùng và khả năng mở rộng trong tương lai:** Website được thiết kế dựa trên mô hình hiện đại, dễ bảo trì, có thể tích hợp thêm các tính năng như chương trình khách hàng thân thiết (loyalty program), tích điểm, chat hỗ trợ tự động,... nhằm nâng cao chất lượng dịch vụ và mở rộng quy mô hoạt động trong tương lai.

Tóm lại, việc xây dựng website Claude Airline là một bước đi chiến lược, không chỉ phục vụ cho việc học tập và thực hành mà còn phản ánh các tiêu chuẩn thiết kế ứng dụng web hiện đại trong ngành dịch vụ hàng không.



2 Cơ sở lý thuyết

2.1 Các công nghệ và thư viện sử dụng

2.1.1 Frontend

Phần giao diện người dùng (frontend) của website **Claude Airline** được xây dựng bằng thư viện **ReactJS**, một trong những công nghệ hiện đại và phổ biến nhất hiện nay trong lĩnh vực phát triển ứng dụng web. Ứng dụng được triển khai theo kiến trúc **Single Page Application (SPA)**, nhằm mang lại trải nghiệm mượt mà, phản hồi nhanh và giảm thiểu thời gian tải lại trang cho người dùng. Bên cạnh đó, nhóm cũng áp dụng nhiều thư viện hỗ trợ khác nhằm tối ưu hóa giao diện và hiệu năng của website.

1. Các thư viện và công nghệ sử dụng bao gồm:

- **ReactJS:** Thư viện JavaScript chính dùng để xây dựng giao diện người dùng dưới dạng các component, giúp dễ dàng tái sử dụng và quản lý mã nguồn hiệu quả.
- **React DOM:** Thư viện hỗ trợ kết nối giữa React và DOM của trình duyệt.
- **React Scripts:** Công cụ đi kèm với Create React App, hỗ trợ các tác vụ như khởi động server, đóng gói mã nguồn và chạy kiểm thử.

2. Giao tiếp với backend:

- **Axios:** Thư viện chuyên dùng để gửi các HTTP request từ frontend đến backend. Nhóm sử dụng Axios để gửi và nhận dữ liệu từ các API như tìm kiếm chuyến bay, đặt vé, và xác thực người dùng.
- **Fetch API:** API tích hợp sẵn trong trình duyệt để thực hiện các request HTTP. Dùng trong các đoạn mã đơn giản hoặc khi không cần cấu hình phức tạp như Axios.

3. Điều hướng trong ứng dụng:

- **React Router DOM:** Dùng để xử lý định tuyến trong ứng dụng, giúp điều hướng giữa các trang như Trang chủ, Đặt vé, Quản lý tài khoản, Liên hệ,... mà không cần tải lại toàn bộ trang.

4. Quản lý thẻ <head> động:

- **React Helmet:** Thư viện hỗ trợ chèn và cập nhật nội dung trong thẻ <head> như <title>, <meta>, <link> từ bên trong các component React. Nhóm sử dụng React Helmet để thay đổi tiêu đề trang, mô tả SEO, favicon và các thẻ meta tương ứng với từng trang như Trang chủ, Giới thiệu. Điều này giúp tăng khả năng tối ưu SEO và cải thiện trải nghiệm người dùng khi chia sẻ đường dẫn trên mạng xã hội.

5. Thiết kế giao diện:

- **Tailwind CSS:** Framework CSS dạng utility-first cho phép thiết kế layout nhanh chóng, linh hoạt và responsive, đảm bảo khả năng hiển thị tốt trên các thiết bị như desktop, tablet và điện thoại.
- **React Slick + Slick Carousel:** Dùng để tạo hiệu ứng trình chiếu hình ảnh trong các banner quảng bá, khuyến mãi, hoặc slide giới thiệu chuyến bay.
- **React Icons:** Thư viện biểu tượng cung cấp hàng trăm icon dạng vector phục vụ việc hiển thị trên các button, thanh điều hướng và các thành phần UI.



6. Tối ưu hiệu suất và tương thích trình duyệt:

- **web-vitals:** Thư viện đo lường các chỉ số hiệu suất web như thời gian tải trang, độ ổn định giao diện, độ phản hồi,... giúp nhóm có thể đánh giá và cải thiện trải nghiệm người dùng.
- **PostCSS & Autoprefixer:** Hỗ trợ tự động thêm các tiền tố (prefix) CSS nhằm tăng khả năng tương thích với nhiều trình duyệt khác nhau.

7. Thư viện hỗ trợ khác:

- **dotenv:** Dùng để quản lý các biến môi trường trong quá trình phát triển như đường dẫn API, khóa bảo mật,...
- **animejs:** Hỗ trợ tạo các hiệu ứng chuyển động, giúp giao diện sinh động và thu hút hơn.

8. Firebase Hosting:

Firebase Hosting là một dịch vụ của Google cung cấp giải pháp hosting nhanh chóng và dễ dàng cho các ứng dụng web và mobile. Nó hỗ trợ các tính năng như tự động cập nhật khi thay đổi mã nguồn, bảo mật qua HTTPS, và khả năng phục hồi nhanh chóng từ sự cố. Firebase Hosting lý tưởng cho các ứng dụng đơn giản và yêu cầu triển khai nhanh chóng.

Những công nghệ và thư viện kể trên giúp nhóm xây dựng một giao diện hiện đại, trực quan và thân thiện với người dùng, đồng thời hỗ trợ vận hành hiệu quả giữa frontend và backend.

2.1.2 Backend

Phần backend của hệ thống website **Claude Airline** được xây dựng bằng **PHP thuần** (không sử dụng framework) kết hợp với **MySQL** để quản lý cơ sở dữ liệu, triển khai trên môi trường phát triển nội bộ sử dụng **XAMPP**.

1. Ngôn ngữ lập trình và môi trường phát triển:

- **PHP:** Là ngôn ngữ lập trình server-side phổ biến, PHP được sử dụng để xử lý các yêu cầu từ phía frontend như: xác thực người dùng, tìm kiếm chuyến bay, đặt vé, quản lý thông tin khách hàng và chuyến bay,... Do sử dụng PHP thuần nên nhóm có toàn quyền kiểm soát logic xử lý và kiến trúc mã nguồn.
- **XAMPP:** Là phần mềm giả lập server giúp nhóm triển khai PHP và MySQL trên môi trường máy tính cá nhân (localhost). XAMPP bao gồm Apache (web server), MySQL/MariaDB (hệ quản trị cơ sở dữ liệu), PHP và phpMyAdmin.

2. Hệ quản trị cơ sở dữ liệu:

- **MySQL (qua phpMyAdmin):** Được sử dụng để lưu trữ toàn bộ dữ liệu hệ thống như thông tin người dùng, chuyến bay, vé đặt, thanh toán, sân bay, lịch trình,... Các thao tác CRUD (Create - Read - Update - Delete) được thực hiện thông qua câu lệnh SQL trong PHP hoặc trực tiếp trên phpMyAdmin khi kiểm thử.

3. Giao tiếp với frontend:

- Backend cung cấp các RESTful API trả về dữ liệu dưới định dạng JSON, giúp frontend (xây dựng bằng ReactJS) dễ dàng gọi và xử lý.
- Các API được xây dựng theo cấu trúc mô-đun, bao gồm các tệp PHP riêng biệt cho từng chức năng.



4. Xử lý bảo mật:

- Các biện pháp bảo vệ cơ bản được áp dụng bao gồm:
 - Kiểm tra và lọc đầu vào (input validation) để tránh lỗi SQL Injection.
 - Sử dụng **Prepared Statements (PDO/MySQLi)** ở một số truy vấn nhạy cảm.
 - Mã hóa mật khẩu người dùng bằng **password_hash()**.
- Do sử dụng PHP thuần, các thao tác bảo mật phải được thực hiện thủ công, giúp nhóm hiểu rõ bản chất hoạt động của hệ thống.

2.1.3 Giao diện quản trị (Admin Panel)

Bên cạnh giao diện người dùng chính được xây dựng bằng ReactJS, nhóm cũng phát triển một trang quản trị (admin panel) dành cho nhân viên hệ thống quản lý dữ liệu như chuyến bay, vé, người dùng, lịch trình,... Giao diện trang admin sử dụng template có sẵn được tùy biến theo nhu cầu của hệ thống. Các công nghệ được sử dụng:

- **Template Mazer (HTML, CSS và Bootstrap):** Đây là một giao diện quản trị HTML tĩnh được thiết kế sẵn, tích hợp Bootstrap 5 và nhiều thành phần UI hiện đại như biểu đồ, bảng dữ liệu, sidebar, navbar,... Nhóm sử dụng template này để tiết kiệm thời gian thiết kế và đảm bảo giao diện nhất quán, hiện đại.
- **PHP thuần (Server-rendered):** Khác với phần frontend chính sử dụng API JSON, phần admin sử dụng cách tiếp cận truyền thống của PHP, trong đó mỗi request tới server sẽ trả về một file .php chứa HTML đã được render (chèn dữ liệu động vào view).

2.2 Ưu điểm và nhược điểm của từng công nghệ

Frontend

Công nghệ / Thư viện	Ưu điểm	Nhược điểm
ReactJS	Tổ chức theo component, dễ mở rộng, cộng đồng lớn	Không tối ưu SEO nếu không dùng SSR, cần thêm thư viện hỗ trợ
React DOM	Cập nhật hiệu quả với Virtual DOM, tích hợp tốt với React	Chỉ dùng cho web, không hỗ trợ native
React Router DOM	Định tuyến linh hoạt, dễ quản lý navigation	Cần nắm vững cấu trúc nested routes
React Slick / Slick Carousel	Tạo slide ảnh đẹp, responsive tốt	Tùy chỉnh phức tạp nếu nâng cao
React Icons	Có nhiều icon phổ biến, dễ dùng	Không phải icon nào cũng sẵn có
Axios	API đơn giản, có interceptors	Kích thước lớn hơn Fetch
Fetch API	Có sẵn trong trình duyệt, nhẹ	Cú pháp dài dòng, không có interceptors
Tailwind CSS	Thiết kế nhanh, hiện đại, class tiện dụng	Dễ rối nếu không tổ chức tốt, khó học ban đầu
animejs	Hiệu ứng animation mượt, nhẹ	Khó dùng nếu cần hiệu ứng phức tạp
web-vitals	Do hiệu suất Core Web Vitals thực tế	Không trực quan với người mới



dotenv	Quản lý biến môi trường dễ, bảo mật cấu hình	Không bảo vệ được mã nhúng vào frontend
PostCSS + Autoprefixer	Tự động hoá CSS, hỗ trợ đa trình duyệt	Cần cấu hình thêm, dễ phát sinh lỗi nhỏ
React Helmet	Hỗ trợ thay đổi <head> động, cải thiện SEO cho SPA	Không tối ưu cho React 18, hiệu suất thấp nếu lạm dụng
Firebase Hosting	Dễ triển khai, chỉ cần vài lệnh, tốc độ nhanh nhờ CDN toàn cầu, bảo mật miễn phí qua HTTPS, tích hợp dễ dàng với các dịch vụ Firebase khác.	Giới hạn tài nguyên cho ứng dụng lớn, chi phí cao khi vượt qua gói miễn phí, tùy chỉnh cấu hình hạn chế.

Backend

Công nghệ / Công cụ	Ưu điểm	Nhược điểm
PHP (thuần)	Dễ học, phổ biến, tích hợp tốt với web server	Khó mở rộng nếu không có framework, hiệu suất không cao
XAMPP	Cài đặt nhanh, tích hợp Apache, MySQL, PHP	Không phù hợp cho môi trường production
MySQL	Mạnh, phổ biến, hỗ trợ SQL chuẩn	Không phù hợp dữ liệu phi cấu trúc, thiếu realtime
phpMyAdmin	Quản lý CSDL trực quan, dễ dùng	Không phù hợp cho thao tác lớn
HTML (Trang admin)	Cơ bản, hỗ trợ tốt mọi trình duyệt	Không động nếu không có JS hoặc PHP
Bootstrap 5	Thiết kế responsive nhanh, nhiều component đẹp	Dễ bị trùng lặp giao diện nếu không tùy chỉnh
Mazer Template	Giao diện admin hiện đại, sẵn biểu đồ, form	Cần hiểu cấu trúc file để tùy biến
PHP SSR (render phía server)	Hiển thị nội dung động nhanh, không phụ thuộc JS	Phải tải lại trang, không linh hoạt bằng React

2.3 Các lỗ hổng bảo mật thường gặp trong ứng dụng web

1. Cross-Site Scripting (XSS)

Mô tả: Là lỗ hổng cho phép kẻ tấn công chèn mã JavaScript độc hại vào nội dung trang web — mã này sẽ được trình duyệt người dùng thực thi.

Ví dụ dễ bị XSS

```
1 <?php
2   echo "Hello " . $_GET['name'];
3 ?>
```

Listing 1: Mã PHP dễ bị XSS

Cách khắc phục



```
1 <?php
2   echo "Hello " . htmlspecialchars($_GET['name']);
3 ?>
```

Listing 2: Khắc phục bằng htmlspecialchars()

2. SQL Injection (SQLi)

Mô tả: SQLi cho phép kẻ tấn công thực thi lệnh SQL tùy ý lên cơ sở dữ liệu bằng cách thao túng input từ người dùng.

Ví dụ dễ bị SQLi

```
1 <?php
2   $username = $_GET['username'];
3   $query = "SELECT * FROM users WHERE username = '$username'";
4   mysqli_query($conn, $query);
5 ?>
```

Cách khắc phục: Sử dụng Prepared Statement

```
1 <?php
2   $stmt = $conn->prepare("SELECT * FROM users WHERE username = ?");
3   $stmt->bind_param("s", $_GET['username']);
4   $stmt->execute();
5 ?>
```

3. Cross-Site Request Forgery (CSRF)

Mô tả: CSRF là lỗ hổng khiếu nại tấn công có thể gửi request thay mặt người dùng mà không được phép, thông qua session đã tồn tại.

Ví dụ Người dùng đăng nhập và truy cập trang sau: <http://example.com/delete.php?id=10>

Kẻ tấn công có thể nhúng link này vào hình ảnh hoặc form giả mạo.

Cách khắc phục

- Sử dụng token CSRF duy nhất cho mỗi phiên và mỗi form.
- Kiểm tra HTTP Referer Header.

4. Directory Traversal

Mô tả: Cho phép kẻ tấn công truy cập trái phép tới file ngoài vùng được phép thông qua kỹ thuật "../"

Ví dụ lỗi

```
1 <?php
2   $file = $_GET['file'];
3   include("documents/" . $file);
4 ?>
```

Truy cập đường dẫn: <http://example.com/index.php?file=../../../../etc/passwd>

Cách khắc phục

- Kiểm tra và giới hạn danh sách file cho phép.
- Sử dụng hàm basename() hoặc thư viện lọc đường dẫn.



5. Upload file không kiểm soát

Mô tả: Cho phép tải lên file thực thi mã độc hoặc shell.

Ví dụ lỗi

```
1 <?php
2     move_uploaded_file($_FILES['file']['tmp_name'], "uploads/" . $_FILES['
3         file']['name']);
?>
```

Cách khắc phục

- Kiểm tra định dạng, phần mở rộng, MIME type.
- Đổi tên file upload và lưu ở thư mục không thể thực thi.

6. Lộ các thông tin nhạy cảm qua JSON API

Mô tả: Nếu API trả về quá nhiều thông tin không cần thiết (ví dụ: hash mật khẩu, ID nội bộ).

Vị trí nguy hiểm: Các API PHP trả JSON như:

```
1 <?php
2     echo json_encode($user); // if $user contain password
3 ?>
```

Cách phòng tránh: Chỉ xuất trường cần thiết, lọc output trước khi trả về.

7. Authentication Bypass (Login lỗi logic)

Mô tả: Nếu hệ thống xác thực không kiểm tra session kỹ, người dùng có thể truy cập trang admin mà không login.

Vị trí nguy hiểm: Thiếu kiểm tra isset(\$_SESSION['admin']) ở đầu file admin.

Cách phòng tránh: Luôn kiểm tra session trên tất cả route cần bảo vệ.

8. CORS Misconfiguration (với ReactJS)

Mô tả: Nếu server cho phép CORS mọi nguồn (Access-Control-Allow-Origin: *), hacker có thể gọi API từ domain độc.

Cách phòng tránh: Giới hạn domain được phép truy cập API.

2.4 Các yếu tố tối ưu hóa SEO trong ứng dụng

Ứng dụng Claude Airlines đã triển khai nhiều yếu tố quan trọng để tối ưu hóa cho các công cụ tìm kiếm (SEO), nhằm cải thiện khả năng hiển thị và xếp hạng trên kết quả tìm kiếm. Các biện pháp chính bao gồm:

- **Sử dụng React Helmet:** Thư viện này được tích hợp trong frontend để quản lý và cập nhật động các thẻ meta (như title, description) cho từng trang. Điều này giúp các công cụ tìm kiếm hiểu rõ hơn về nội dung của mỗi trang riêng biệt, một yếu tố quan trọng cho các ứng dụng đơn trang (SPA).
- **Cấu hình sitemap.xml và robots.txt:** Các file này được đặt trong thư mục public của frontend. sitemap.xml giúp công cụ tìm kiếm khám phá tất cả các trang quan trọng của ứng dụng, trong khi robots.txt chỉ dẫn cho các trình thu thập dữ liệu về các phần của trang web mà chúng nên hoặc không nên truy cập.

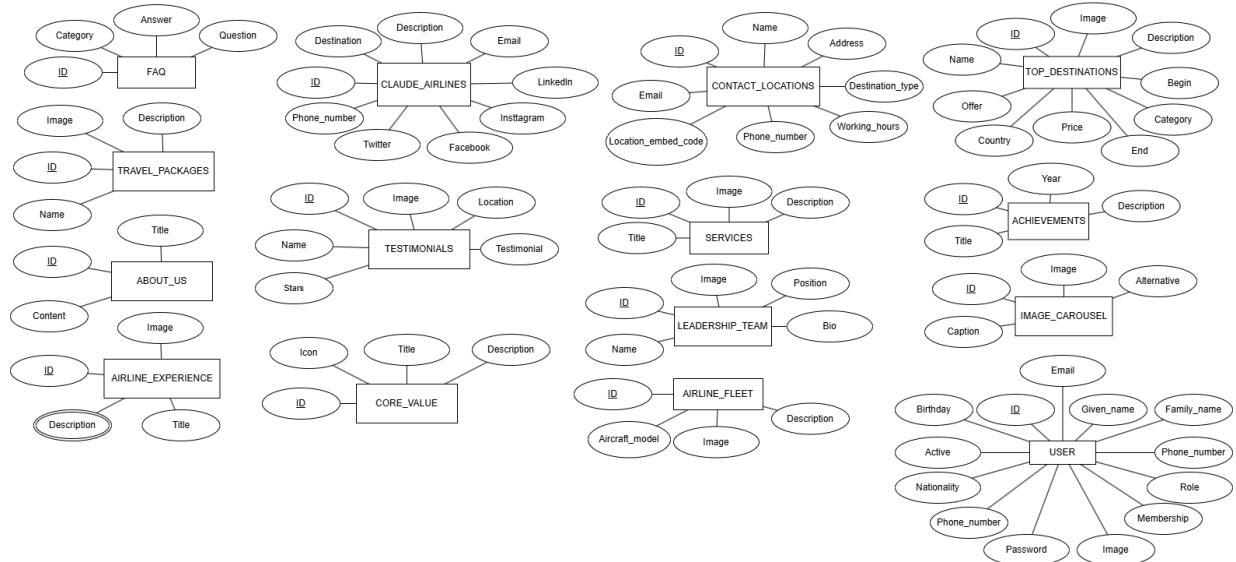
- Thuộc tính alt cho hình ảnh:** Tất cả hình ảnh trong ứng dụng đều sử dụng thuộc tính alt với mô tả rõ ràng. Điều này không chỉ cải thiện khả năng truy cập cho người dùng khiếm thị mà còn cung cấp ngữ cảnh cho công cụ tìm kiếm về nội dung của hình ảnh.
- Cấu trúc HTML ngữ nghĩa:** Ứng dụng được xây dựng với cấu trúc HTML rõ ràng và ngữ nghĩa, sử dụng các thẻ HTML phù hợp với mục đích của chúng. Điều này giúp công cụ tìm kiếm dễ dàng phân tích và hiểu cấu trúc nội dung của trang.

Những yếu tố này đóng góp vào việc cải thiện SEO tổng thể cho ứng dụng, giúp thu hút lưu lượng truy cập tự nhiên từ các công cụ tìm kiếm.

3 Thiết kế ứng dụng

3.1 Thiết kế cơ sở dữ liệu

3.1.1 Entity Relationship Diagram



Hình 1: ERD



3.1.2 Cấu trúc và chức năng của các bảng dữ liệu

Bảng USERS

1. **Mô tả:** Lưu thông tin người dùng hệ thống, bao gồm tên, email, mật khẩu, vai trò và trạng thái kích hoạt tài khoản.
2. **Đặc tả bảng:**
 - **id:** Khóa chính, tự tăng.
 - **email:** Duy nhất, định dạng hợp lệ.
 - **password:** Được mã hóa trước khi lưu.
 - **role:** Xác định quyền truy cập - ENUM ('USER', 'ADMIN'), mặc định = USER.
 - **active:** Trạng thái tài khoản - Boolean, mặc định = 1.
 - **id, email, password, family_name, given_name, birthday, nationality:** not null.

Bảng CLAUDE_AIRLINES

1. **Mô tả:** Lưu thông tin liên hệ và mô tả của hãng hàng không.
2. **Đặc tả bảng:**
 - **id:** Khóa chính, tự tăng.
 - **email, phone_number, destination, description:** Các thông tin mô tả chính.
 - **facebook, twitter, instagram, linkedin:** Liên kết mạng xã hội.

Bảng CONTACT_LOCATIONS

1. **Mô tả:** Lưu các địa điểm liên hệ của hãng hàng không tại nhiều quốc gia.
2. **Đặc tả bảng:**
 - **id:** Khóa chính.
 - **location_name, address_string, phone_number, email, location_embed_code:** Các thông tin liên hệ.
 - **working_hours, des_type:** Mô tả thêm cho từng địa điểm.

Bảng TOP_DESTINATIONS

1. **Mô tả:** Lưu thông tin về các điểm đến nổi bật được hiển thị trên trang chủ.
2. **Đặc tả bảng:**
 - **id:** Khóa chính.
 - **destination_image, destination_name, destination_country, destination_price, destination_description:**
 - **destination_begin, destination_end:** Thời gian khuyến mãi hoặc hoạt động.
 - **destination_offer, destination_category:** Các đặc điểm bổ sung.



Bảng TRAVEL_PACKAGES

1. **Mô tả:** Lưu các gói du lịch nổi bật.
2. **Đặc tả bảng:**
 - `id`: Khóa chính.
 - `package_image`, `package_name`, `package_description`.

Bảng ABOUT_US

1. **Mô tả:** Lưu nội dung giới thiệu về công ty.
2. **Đặc tả bảng:**
 - `id`, `title`, `content`: not null.

Bảng USER_TESTIMONIALS

1. **Mô tả:** Lưu phản hồi từ người dùng.
2. **Đặc tả bảng:**
 - `id`, `user_name`, `user_testimonial`, `user_image`, `user_stars`, `user_location`.

Bảng SERVICES

1. **Mô tả:** Lưu danh sách các dịch vụ được cung cấp.
2. **Đặc tả bảng:**
 - `id`, `service_title`, `service_description`, `service_image`.

Bảng IMAGE_CAROUSEL

1. **Mô tả:** Lưu hình ảnh trình chiếu trên trang chủ.
2. **Đặc tả bảng:**
 - `id`, `carousel_image`, `carousel_alt`, `carousel_caption`.

Bảng AIRLINE_EXPERIENCE

1. **Mô tả:** Lưu các trải nghiệm dịch vụ của hãng hàng không.
2. **Đặc tả bảng:**
 - `id`, `title`: not null
 - `image`.

Bảng EXPERIENCE_DESCRIPTION

1. **Mô tả:** Lưu mô tả chi tiết cho từng trải nghiệm dịch vụ.
2. **Đặc tả bảng:**
 - `id`, `airline_experience_id` (FK), `description`: not null.



Bảng CORE_VALUE

1. **Mô tả:** Lưu các giá trị cốt lõi của công ty.
2. **Đặc tả bảng:**
 - id, title, description: not null
 - icon.

Bảng LEADERSHIP_TEAM

1. **Mô tả:** Lưu thông tin ban lãnh đạo công ty.
2. **Đặc tả bảng:**
 - id, name, position, bio: not null.
 - image.

Bảng AIRLINE_FLEET

1. **Mô tả:** Lưu thông tin về các máy bay trong đội bay.
2. **Đặc tả bảng:**
 - id, aircraft_model, description: not null
 - image.

Bảng FAQ

1. **Mô tả:** Lưu các câu hỏi thường gặp và các câu trả lời tương ứng.
2. **Đặc tả bảng:**
 - id, question, answer, category: not null
 - image.

Bảng ACHIEVEMENTS

1. **Mô tả:** Lưu thông tin về thành tựu của công ty.
2. **Đặc tả bảng:**
 - id, year, title, description: not null.



3.1.3 Ánh xạ cơ sở dữ liệu thành dạng bảng



Hình 2: Mapping



3.2 Mô hình ứng dụng

Hệ thống website Claude Airline được thiết kế theo mô hình kiến trúc MVC nhằm đảm bảo khả năng mở rộng, bảo trì, tái sử dụng và phân tách rõ ràng các thành phần trong ứng dụng. Trong quá trình phát triển, nhóm áp dụng hai hình thức hiển thị View khác nhau cho hai phân hệ chính:

- Giao diện hành khách (khách hàng): được xây dựng bằng ReactJS, hoạt động theo kiểu Single Page Application (SPA), giao tiếp với backend thông qua REST API.
- Giao diện quản trị viên (admin): sử dụng server-side rendering (SSR) bằng PHP thuận, với bộ giao diện dựa trên template Mazer (Bootstrap).

Cả hai giao diện đều sử dụng chung hệ thống Model và Controller, đảm bảo tính nhất quán về mặt logic xử lý nghiệp vụ và truy cập dữ liệu.

Các thành phần trong mô hình MVC của hệ thống

1. Model (M) – Xử lý dữ liệu

Model đảm nhận việc tương tác với cơ sở dữ liệu MySQL. Tất cả các thao tác như truy xuất, thêm, sửa, xoá đều được viết trong các lớp PHP riêng biệt. Mỗi đối tượng nghiệp vụ có một model.

Các Model sử dụng câu lệnh SQL trực tiếp để tương tác với cơ sở dữ liệu thông qua thư viện mysqli.

2. Controller (C) – Điều phối xử lý logic

Controller đóng vai trò trung gian giữa người dùng (qua View) và dữ liệu (qua Model). Nó xử lý luồng dữ liệu, gọi các hàm trong Model để xử lý logic nghiệp vụ và trả kết quả cho View.

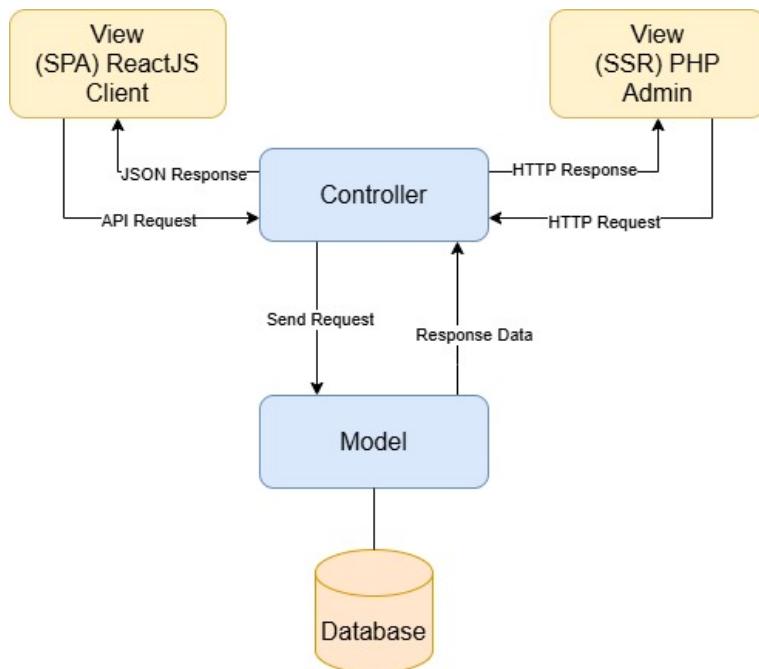
3. View (V) – Giao diện hiển thị dữ liệu

(a) View động – ReactJS (SPA)

- Dùng cho người dùng cuối (khách hàng).
- Giao diện viết bằng ReactJS, gọi API PHP qua axios để lấy dữ liệu JSON.
- Hiển thị thông tin vé máy bay, tìm kiếm chuyến bay, đặt chỗ, quản lý hồ sơ,...
- Tất cả các thao tác đều xử lý bất đồng bộ, không cần tải lại trang.

(b) View SSR - Template HTML/PHP (Admin)

- Dùng cho quản trị viên nội bộ.
- Viết bằng PHP kết hợp với HTML, CSS, Bootstrap trong template Mazer.
- View được render trên server bằng PHP: Controller lấy dữ liệu từ Model và gắn trực tiếp vào View.
- Quản lý danh sách chuyến bay, sân bay, người dùng, đơn đặt vé,...



Hình 3: MVC Design Pattern

Luồng xử lý dữ liệu

1. Với ReactJS (SPA - API):

- Người dùng thao tác trên giao diện React → gọi API bằng axios.
- API được định tuyến về các Controller tương ứng (ví dụ: /users/login).
- Controller xử lý → gọi Model → lấy dữ liệu từ MySQL.
- Dữ liệu trả về dưới dạng JSON.
- React hiển thị nội dung theo thời gian thực.

2. Với Admin (SSR - Template):

- Người dùng nhập URL → Router định tuyến đến Controller.
- Controller gọi Model để lấy dữ liệu từ cơ sở dữ liệu.
- Dữ liệu được nhúng trực tiếp vào HTML template bằng PHP.
- Giao diện hoàn chỉnh được trả về trình duyệt.

Ưu điểm khi sử dụng kết hợp MVC + SPA + SSR:

- Tách biệt giao diện và xử lý logic: Dễ bảo trì, tái sử dụng mã nguồn.
- React SPA: Tải nhanh, mượt, trải nghiệm người dùng tốt hơn.
- SSR, PHP (Admin): Dễ phát triển, triển khai nhanh, phù hợp chức năng nội bộ.
- Dùng chung Controller & Model: Đồng bộ dữ liệu, tránh dư thừa xử lý logic.
- Khả năng mở rộng: Dễ tích hợp mobile app (qua API), thêm frontend framework.



3.3 Cấu trúc mã nguồn

```
└── backend/
    └── app/
        ├── config/ // Chứa các tệp cấu hình, như config.php để lưu thông tin kết nối cơ sở dữ liệu.
        ├── controllers/ // Chứa các controller xử lý nghiệp vụ.
        ├── core/ // Chứa các tệp cốt lõi như Database.php, cung cấp kết nối và xử lý truy vấn cơ sở dữ liệu.
        ├── helpers/ // Các hàm tiện ích dùng chung toàn hệ thống.
        ├── middlewares/ // Kiểm soát request trước khi tới controller.
        ├── models/ // Truy vấn, xử lý dữ liệu với database.
        ├── views/ // Chứa các tệp HTML/PHP hiển thị giao diện.
        └── .env // Biến môi trường (database, secret keys)

    └── public/
        ├── assets/ // Ảnh, JS, CSS tĩnh, Bootstrap framework cho trang admin.
        ├── uploads/ // Nơi lưu trữ file, hình ảnh được tải lên.
        └── index.php // entry point chính, dùng để boot hệ thống (Front Controller).

└── frontend
    └── public/ // Các file tĩnh như favicon, manifest, index.html (điểm khởi tạo React).
    └── src
        ├── api/ // Định nghĩa các hàm gọi API tới backend
        ├── assets/ // Hình ảnh, icon, font được sử dụng trong giao diện.
        ├── components/ // Các thành phần tái sử dụng được (Header, Footer, Button...).
        ├── helpers/ // Các hàm xử lý dữ liệu client như format, validation.
        └── pages/ // Các trang logic ứng với từng route (Home, FAQ, About...).

    └── .env // Biến môi trường API_URL
```



3.4 Các tính năng chính của hệ thống

Hệ thống website được xây dựng nhằm cung cấp giao diện thân thiện cho người dùng và công cụ quản trị hiệu quả cho quản trị viên. Các tính năng chính bao gồm:

3.4.1 Tính năng người dùng

- Đăng ký / Đăng nhập / Đăng xuất:** Người dùng có thể tạo tài khoản, đăng nhập, và quản lý phiên làm việc.
- Phân quyền người dùng:** Hệ thống phân biệt giữa người dùng thường và quản trị viên để giới hạn quyền truy cập.
- Cập nhật thông tin cá nhân:** Cho phép người dùng thay đổi tên hiển thị, mật khẩu, avatar, v.v.
- Tìm kiếm và đặt chuyến bay:** Giao diện đơn giản, dễ sử dụng, cho phép người dùng tìm và đặt vé máy bay nhanh chóng.
- Quản lý đơn đặt vé:** Theo dõi các chuyến bay đã đặt, tình trạng vé, và thông tin thanh toán.
- Giao diện responsive:** Thiết kế hiện đại, tương thích với các thiết bị di động, máy tính bảng và desktop.
- Đăng ký nhận bản tin (Newsletter):** Người dùng có thể đăng ký nhận thông báo qua email về tin tức hoặc khuyến mãi.

3.4.2 Tính năng dành cho quản trị viên

- Quản lý người dùng:** Xem thông tin người dùng, khóa/mở tài khoản, reset mật khẩu.
- Quản lý nội dung trang tĩnh:**
 - Trang chủ
 - Trang Giới thiệu
 - Trang Liên hệ
 - Trang Hỏi/Dáp

Cho phép cập nhật nội dung văn bản, hình ảnh, logo, địa chỉ, số điện thoại, v.v.

- Quản lý liên hệ khách hàng:**

- Xem danh sách liên hệ gửi đến.
- Dánh dấu liên hệ đã đọc/chưa đọc/đã phản hồi.
- Xóa liên hệ không cần thiết.

- Quản lý hỏi/dáp (FAQ):**

- Xem, thêm, chỉnh sửa và xóa các câu hỏi và câu trả lời thường gặp.



3.4.3 Giao diện người dùng

- **Trang chủ:** Hiển thị tổng quan nội dung website.
- **Trang Giới thiệu:** Thông tin về tổ chức hoặc doanh nghiệp.
- **Trang Liên hệ:** Thông tin liên hệ và biểu mẫu gửi phản hồi.
- **Trang Hỏi/Đáp:** Danh sách các câu hỏi và trả lời phổ biến.

3.5 Các sơ đồ thể hiện các quy trình chính

4 Hiện thực

4.1 Các tính năng đã hiện thực

Dựa trên kế hoạch và thiết kế, hệ thống website Claude Airline đã hoàn thiện các nhóm tính năng chính, phục vụ cả người dùng đầu cuối và quản trị viên hệ thống. Dưới đây là chi tiết các chức năng đã được hiện thực:

4.1.1 Tính năng phía Người dùng (Client-side)

Các tính năng này được xây dựng bằng ReactJS, tập trung vào trải nghiệm người dùng mượt mà và tương tác trực quan. Toàn bộ các trang được tối ưu hóa cho công cụ tìm kiếm (SEO) bằng cách sử dụng React Helmet để quản lý các thẻ meta.

- **Trang Chủ (Homepage):** Đây là giao diện chính của website, được thiết kế để thu hút sự chú ý của người dùng. Trang bao gồm một khu vực trình chiếu hình ảnh động (hero slider) ở đầu trang, cùng với một biểu mẫu tìm kiếm chuyến bay rõ ràng và một khung tìm kiếm nhanh dạng lõp phủ tiện lợi. Nội dung trang được làm phong phú với các mục như:
 - Cung cấp đa dạng lựa chọn khám phá và lên kế hoạch chuyến đi, bao gồm các điểm đến hấp dẫn (với chức năng lọc), các gói du lịch đặc biệt, và những gợi ý cho hành trình tiếp theo.
 - Thông tin về các chương trình ưu đãi được cập nhật liên tục qua khu vực đếm ngược khuyến mãi, cùng với biểu mẫu tiện lợi để người dùng đăng ký nhận tin tức và các ưu đãi mới nhất.
 - Hiển thị các nội dung làm tăng sự tin cậy và hiểu biết về hãng, như những đánh giá chân thực từ hành khách, giới thiệu chi tiết về các dịch vụ cao cấp và một bộ sưu tập hình ảnh trực quan.
- **Trang Giới Thiệu (About Us):** Cung cấp thông tin chi tiết về Claude Airlines, được lấy động từ backend qua API, bao gồm:
 - Thông tin chung về công ty.
 - Giá trị cốt lõi của Claude Airlines.
 - Các thành tựu nổi bật đã đạt được.
 - Thông tin về đội ngũ lãnh đạo.
 - Mô tả về trải nghiệm bay cùng hãng.



- Trang Đội Bay: Hiển thị thông tin chi tiết về các loại máy bay trong đội bay của Claude Airlines, dữ liệu được tải từ AirlineFleetModel thông qua API.
- Xác thực người dùng:
 - Chức năng Đăng nhập cho người dùng đã có tài khoản.
 - Chức năng Đăng ký tài khoản mới.
- Trang Liên Hệ:
 - Hiển thị thông tin liên hệ chung (địa chỉ, số điện thoại, giờ làm việc).
 - Liệt kê các địa điểm văn phòng/chi nhánh, với dữ liệu được lấy từ ContactLocationModel qua API, có khả năng hiển thị dưới dạng danh sách hoặc tích hợp bản đồ.
 - Cung cấp biểu mẫu liên hệ cho phép người dùng gửi tin nhắn hoặc yêu cầu hỗ trợ trực tiếp.
- Trang Câu Hỏi Thường Gặp (FAQ): Hiển thị danh sách các câu hỏi và câu trả lời thường gặp, được phân loại theo danh mục để người dùng dễ dàng tra cứu. Dữ liệu được lấy từ FaqModel qua API và trang được tối ưu SEO.

4.1.2 Tính năng phía Quản trị viên (Admin-side)

Giao diện quản trị được xây dựng bằng PHP thuần kết hợp template Mazer, cho phép quản lý các nội dung và dữ liệu của website.

- Quản lý Nội dung Trang "Giới Thiệu": Cho phép quản trị viên xem, thêm, sửa, xóa các mục thông tin chung hiển thị trên trang "Giới Thiệu" (tiêu đề, nội dung). Cung cấp API (ví dụ: getAboutUs) để frontend lấy dữ liệu.
- Quản lý Thành Tựu: Cho phép quản trị viên quản lý danh sách các thành tựu của công ty (năm, tiêu đề, mô tả) thông qua các thao tác CRUD. Cung cấp API (ví dụ: getAchievements) cho frontend.
- Quản lý Trải Nghiệm Bay: Cho phép quản trị viên thêm, sửa, xóa các mục mô tả trải nghiệm bay (tiêu đề, mô tả chi tiết, hình ảnh minh họa). Cung cấp API cho frontend.
- Quản lý Đội Bay: Cho phép quản trị viên quản lý thông tin các loại máy bay trong đội bay (model, mô tả, hình ảnh) thông qua các thao tác CRUD. Cung cấp API cho frontend.
- Quản lý Địa Điểm Liên Hệ: Cho phép quản trị viên quản lý danh sách các địa điểm liên hệ (tên, loại, địa chỉ, SDT, giờ làm việc, email, mã nhúng bản đồ) với các chức năng CRUD và xác thực dữ liệu đầu vào. Cung cấp API (ví dụ: index, getById) cho frontend.
- Quản lý Giá Trị Cốt Lõi: Cho phép quản trị viên thêm, sửa, xóa các giá trị cốt lõi của công ty (tiêu đề, mô tả, icon). Hệ thống hỗ trợ xử lý xóa file icon liên quan khi một giá trị bị xóa. Cung cấp API cho frontend.
- Quản lý Câu Hỏi Thường Gặp (FAQ): Cho phép quản trị viên quản lý danh sách các câu hỏi FAQ (câu hỏi, câu trả lời, danh mục) với đầy đủ các thao tác CRUD. Cung cấp API cho frontend.



- Quản lý Đội Ngũ Lãnh Đạo: Cho phép quản trị viên quản lý thông tin các thành viên trong đội ngũ lãnh đạo (tên, chức vụ, tiểu sử, hình ảnh) thông qua các thao tác CRUD. Cung cấp API cho frontend.
- Quản lý Người Dùng: Cho phép quản trị viên xem danh sách, thêm, sửa, kích hoạt/vô hiệu hóa tài khoản người dùng (bao gồm thông tin cá nhân, vai trò). Hỗ trợ hiển thị hình ảnh đại diện và sử dụng session cho các thông báo lỗi/thành công.
- Quản lý Trình Chiếu Hình Ảnh (Image Carousel): Cho phép quản trị viên quản lý các hình ảnh hiển thị trong slider/carousel trên trang chủ, thông qua controller và model tương ứng ở backend.

4.1.3 Tính năng Kỹ thuật và Chung

Các tính năng nền tảng và kỹ thuật hỗ trợ cho toàn bộ hệ thống.

- Backend API theo kiến trúc RESTful: Hệ thống backend cung cấp các API được thiết kế theo chuẩn RESTful, giúp việc giao tiếp giữa frontend và backend trở nên rõ ràng và hiệu quả.
- Kiến trúc MVC cho Backend: Phần backend được xây dựng dựa trên mô hình MVC (Model-View-Controller) sử dụng PHP thuần, giúp tách biệt logic, dữ liệu và giao diện, dễ dàng bảo trì và mở rộng.
- Controller và Database Core: Controller.php trong thư mục core xử lý các vấn đề chung như CORS, tải model và view. Database.php quản lý kết nối và thực thi các truy vấn đến cơ sở dữ liệu MySQL.
- Middleware Xác thực (authMiddleware.php): Implement một middleware để bảo vệ các route hoặc chức năng quan trọng yêu cầu người dùng phải đăng nhập và có quyền truy cập.
- Tối ưu hóa SEO với React Helmet ở Frontend: Tất cả các trang quan trọng phía người dùng đều được tích hợp React Helmet để quản lý động các thẻ meta, cải thiện khả năng được lập chỉ mục và xếp hạng bởi các công cụ tìm kiếm.
- Giao diện đáp ứng (Responsive UI): Giao diện người dùng được thiết kế để tương thích và hiển thị tốt trên nhiều loại thiết bị và kích thước màn hình khác nhau (desktop, tablet, mobile).
- Triển khai Frontend bằng Firebase Hosting: Hệ thống đã được cấu hình để có thể triển khai phần frontend lên Firebase Hosting một cách dễ dàng.
- Helper jsonResponse.php: Một hàm tiện ích được sử dụng ở backend để chuẩn hóa cấu trúc và định dạng của các phản hồi JSON từ API.

4.2 Demo giao diện

Phần này sẽ trình bày giao diện của các trang chính trong hệ thống website Claude Airline.



4.2.1 Trang Đăng Nhập (Login)

Hình 4: Giao diện Trang Đăng Nhập

4.2.1.1 Mô tả chung Trang Đăng Nhập cho phép người dùng hiện tại truy cập hệ thống Claude Airline với giao diện tập trung, đơn giản.

4.2.1.2 Các thành phần giao diện

- Khu vực chính:** Logo Claude Airline, tiêu đề "Welcome Back", và biểu mẫu đăng nhập gồm trường "Email" (có biểu tượng *FaEnvelope*) và "Mật khẩu" (có biểu tượng *FaLock*). Cả hai đều bắt buộc.
- Tùy chọn và điều hướng:** Bao gồm tùy chọn "Remember me", liên kết "Forgot Password?", nút "Login to Account" (có trạng thái xử lý), và các liên kết quay lại trang chủ hoặc chuyển đến trang đăng ký.
- Thông báo:** Hiển thị kết quả đăng nhập (thành công/lỗi) và tự động chuyển hướng khi thành công.

4.2.1.3 Chức năng chính

- Xác thực client-side:** Kiểm tra tính hợp lệ và đầy đủ của email, mật khẩu.
- Gọi API backend (`Authen.login()`):** Gửi thông tin để xác thực người dùng.
- Phân quyền và điều hướng:** Chuyển hướng ADMIN đến trang quản trị, người dùng thường đến trang cá nhân/trang chủ và lưu 'accessToken' vào 'localStorage'.
- Xử lý lỗi:** Thông báo lỗi đăng nhập cụ thể (sai thông tin, tài khoản không hoạt động).



4.2.2 Trang Đăng Ký (Register)

Hình 5: Giao diện Trang Đăng Ký

4.2.2.1 Mô tả chung Trang Đăng Ký cho phép người dùng mới tạo tài khoản Claude Airline, với biểu mẫu chi tiết hơn trang Đăng Nhập.

4.2.2.2 Các thành phần giao diện

- Khu vực chính:** Logo Claude Airline, tiêu đề "Create Your Account", và biểu mẫu đăng ký được chia thành các mục: "Thông tin cá nhân" (Họ, Tên, Ngày sinh, Quốc tịch), "Thông tin liên hệ" (Email, SDT, Thẻ thành viên - tùy chọn), và "Bảo mật tài khoản" (Mật khẩu, Xác nhận mật khẩu, thanh đo độ mạnh mật khẩu).
- Yêu cầu và điều hướng:** Yêu cầu đồng ý với Điều khoản và Chính sách, nút "Create Account" (có trạng thái xử lý), và các liên kết quay lại trang chủ hoặc chuyển đến trang đăng nhập.
- Thông báo:** Hiển thị kết quả đăng ký và tự động chuyển hướng đến trang đăng nhập khi thành công.

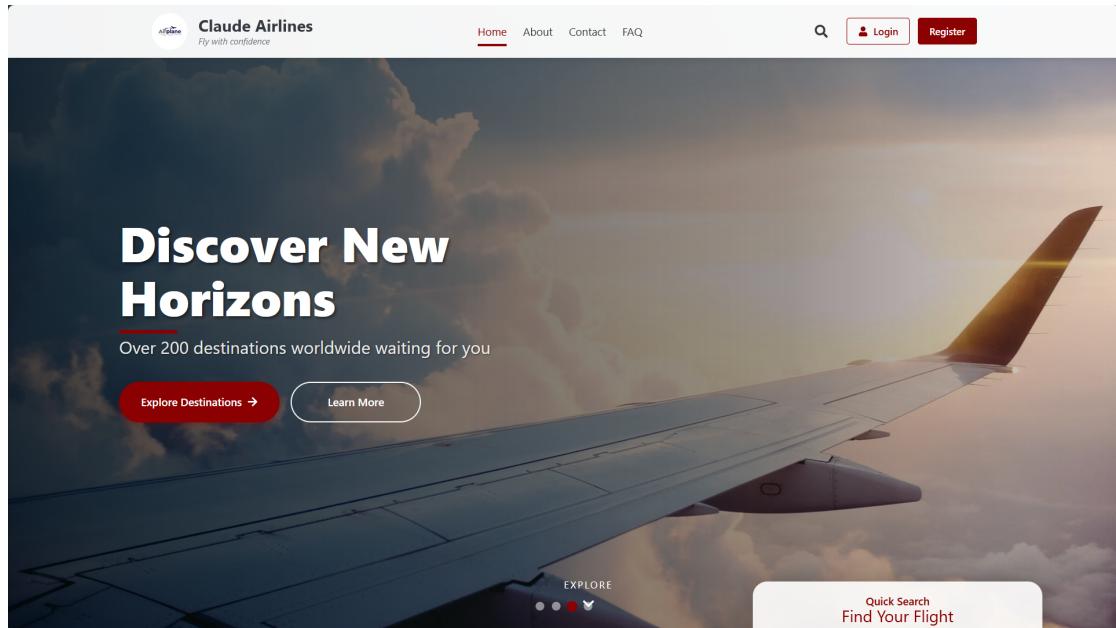
4.2.2.3 Chức năng chính

- Xác thực client-side:** Kiểm tra toàn diện các trường bắt buộc (rỗng, định dạng email/SDT, độ dài và khớp mật khẩu).
- Gọi API backend (Authen.register()):** Gửi thông tin để tạo tài khoản mới.
- Xử lý phản hồi API:** Thông báo lỗi (email đã tồn tại) hoặc thông báo thành công và hướng dẫn bước tiếp theo.



4.2.3 Trang Chủ (Homepage)

4.2.3.1 Mô tả chung Trang Chủ (Homepage.js) là giao diện chính của Claude Airlines, được thiết kế để cung cấp tổng quan về dịch vụ, điểm đến và ưu đãi, thu hút người dùng ngay từ cái nhìn đầu tiên.



Hình 6: Giao diện Trang Chủ

4.2.3.2 Các thành phần chính và chức năng

- Phần Đầu Trang (HeaderHomepage.js):** Bao gồm *Hero Slider* với các slide động, thông điệp và nút CTA; *Khung Tìm Kiếm Nhanh* cho phép truy cập nhanh biểu mẫu tìm chuyến bay chi tiết; và *Hiệu ứng cuộn* thay đổi giao diện header.
- Điểm Đến Nổi Bật (FeaturedDestinations.js):** Tải động và hiển thị các điểm đến hàng đầu dưới dạng thẻ trực quan (hình ảnh, tên, giá, ưu đãi). Cho phép lọc theo danh mục và xem thêm/thu gọn.



The screenshot shows a travel booking interface. At the top, there's a navigation bar with the Claude Airlines logo, a search bar, and user account options (Login, Register). Below the header, a filter section allows users to 'Filter by category' with options like 'All', 'North America', 'Asia', '312', and 'Europe'. Three travel deals are displayed in a grid:

- New York2 2** (USA) - \$899.99 round trip. Valid from 2025-09-09 to 2026-02-15. Book Now.
- Tokyo** (Japan) - \$699.99 round trip. Valid from 2024-01-15 to 2024-01-25. Book Now.
- New York** (USA) - \$899.99 round trip. Valid from 2024-02-05 to 2024-02-15. Book Now.

A 'View All Destinations' button is located at the bottom of the grid.

Hình 7: Giao diện Các điểm đến nổi bật

- **Gói Du Lịch (TravelPackage.js):** Tải và trình bày thông tin các gói du lịch đặc biệt từ API.

The screenshot shows a travel package section. At the top, there's a navigation bar with the Claude Airlines logo, a search bar, and user account options (Login, Register). A main heading reads 'Looking for a travel package?' followed by a subtext: 'Check out our travel packages to make your trip planning easier.' Two travel packages are highlighted with large images:

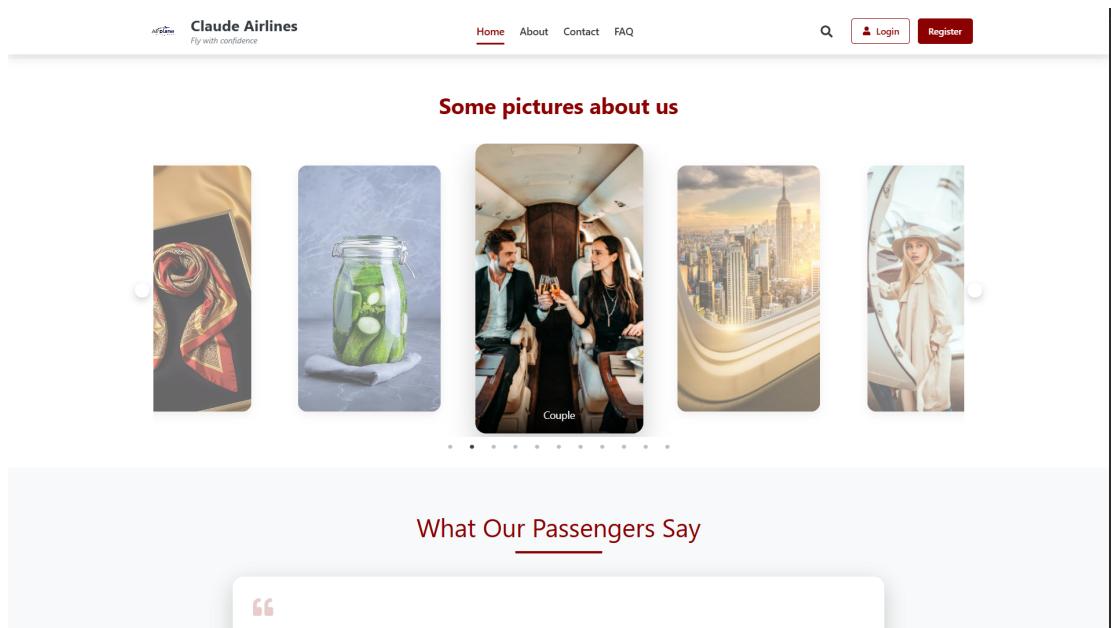
- F1 Grand Prix**: An image of a Formula 1 racing car. Text below it says: 'Experience the thrill of Formula 1 racing with our exclusive travel package.'
- UEFA Champions League**: An image of five football players in various team jerseys. Text below it says: 'Watch the UEFA Champions League final live with our travel package.'

Hình 8: Giao diện Các gói du lịch (travel packages)

- **Trình Chiếu Hình Ảnh (ImageCarousel.js):** Sử dụng 'react-slick' để tạo carousel hiển



thì các hình ảnh giới thiệu về hãng, dịch vụ, hoặc điểm đến, được tải động từ API.



Hình 9: Giao diện Thư viện ảnh ở trang chủ

- **Đánh Giá Từ Hành Khách (Testimonials.js):** Hiển thị các nhận xét, đánh giá của hành khách dưới dạng slider, tải từ API (có dữ liệu mẫu dự phòng).
- **Khu Vực Dịch Vụ (ServicesSection.js):** Giới thiệu các dịch vụ cao cấp (giải trí, ghế ngồi) sử dụng component 'ServiceUnit.js', dữ liệu tải từ API.



The screenshot shows the Claude Airlines website's "Our Premium Services" section. At the top, there is a navigation bar with links for Home, About, My Vlog, Contact, and FAQ. There are also search, login, and register buttons. The main heading is "Our Premium Services" with a subtext: "Experience the comfort and luxury with our exclusive in-flight services designed to make your journey memorable." Below this, there are three service cards: "Inflight Entertainment" (two passengers watching a screen), "Comfortable Seating" (a passenger sleeping comfortably), and "Gourmet Dining" (a meal tray with various dishes). Each card has a red circular icon at the bottom right.

Hình 10: Giao diện Danh mục dịch vụ

- **Thông Điệp Khuyến Mãi Phụ (Chom.js):** Khối thông tin tinh nhẩn mạnh ưu đãi giá vé.

The screenshot shows the Claude Airlines website's promotional offer section. The heading is "Save more with us". On the left, there is a block of text: "The ticket price was already affordable, now it's even more attractive. Book your flight with us and enjoy a 10% discount on all routes. Save 10% on all routes. Grab the offer to reach your desired destination." Below this is a "Book Now" button. To the right is a large image of the Tokyo skyline at sunset, featuring Mount Fuji and the Tokyo Tower.

- **Đăng Ký Nhận Ưu Đãi (Offer.js):** Cung cấp biểu mẫu đăng ký email nhận bản tin và khuyến mãi.



- **Bản Đồ Vị Trí và Chân Trang:** Nhúng bản đồ Google Maps và hiển thị thông tin bản quyền.

Claude Airlines
Fly with confidence

Home About Contact FAQ

Search

Login Register

Our Location

The map displays the Claude Airlines office at 123 Aviation Blvd, Los Angeles, CA 90045. It also shows the University of Technology, Ho Chi Minh City, and various landmarks in the surrounding area, including the Bitexco Financial Tower, the Saigon River, and the Ben Thanh Market. The map includes a legend for keyboard shortcuts and copyright information from Google.

© 2025 Our Airline. All rights reserved.

Claude Airlines

Claude Airlines is dedicated to providing world-class air travel experiences with modern aircraft, exceptional service, and a commitment to safety and sustainability.

[Home](#) [About Us](#) [Our Fleet](#) [Services](#) [FAQ](#) [Contact Us](#)

Quick Links

[Home](#) [About Us](#) [Our Fleet](#) [Services](#) [FAQ](#) [Contact Us](#)

Our Services

[Flight Booking](#) [Cargo Services](#) [Charter Flights](#) [In-flight Entertainment](#) [Loyalty Program](#)

Contact Us

123 Aviation Blvd, Los Angeles, CA 90045
+1 (800) 123-4567
support@skylineairlines.com

Subscribe to Our Newsletter

Enter your email [Subscribe](#)

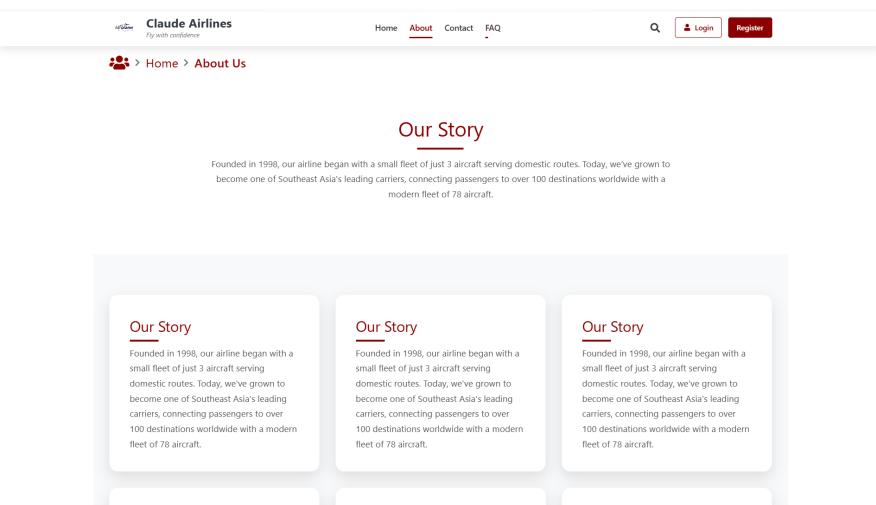


4.2.4 Trang Giới Thiệu (About Us)

4.2.4.1 Mô tả chung Trang Giới Thiệu ('AboutUs.js') cung cấp thông tin chi tiết về Claude Airlines, bao gồm lịch sử, sứ mệnh, giá trị cốt lõi, đội ngũ và các thành tựu. Trang được thiết kế với nhiều thành phần con để truyền tải thông tin một cách trực quan và hấp dẫn.

4.2.4.2 Các thành phần giao diện chính

- **Header Section:** Hình ảnh nền lớn, tiêu đề "About Our Airline" và mô tả ngắn.
- **Breadcrumbs:** Thanh điều hướng giúp người dùng xác định vị trí hiện tại (Home > About Us).
- **Nội dung chính ('ourInformation':**
 - Hiển thị "Our Story" và các thông tin giới thiệu chung khác được lấy từ API ('about.getAllAboutUs()').
 - Nội dung được sắp xếp để "Our Story" luôn hiển thị đầu tiên.
- **Các Thành Phần Con (Components):**
 - **Giá Trị Cốt Lõi ('CoreValues.js')**: Hiển thị các giá trị cốt lõi của công ty (lấy từ API 'about.getAllCoreValues()') kèm theo biểu tượng và mô tả.
 - **Đội Ngũ Lãnh Đạo ('LeadershipTeam.js')**: Giới thiệu các thành viên trong đội ngũ lãnh đạo (lấy từ API 'about.getLeadershipTeam()') với hình ảnh, tên, chức vụ và tiểu sử.
 - **Thành Tựu ('Achievements.js')**: Trình bày các thành tựu nổi bật theo dòng thời gian (lấy từ API 'about.getAllAchievements()').
 - **Trải Nghiệm Bay ('Experience.js')**: Mô tả các khía cạnh khác nhau của trải nghiệm bay (lấy từ API 'about.getAllExperiences()') với hình ảnh và mô tả chi tiết, có thể liên kết đến các trang con.
- **Kêu gọi hành động (Call to Action):** Các nút điều hướng người dùng đến trang "Liên Hệ" và "FAQ".



Hình 11: Giao diện Trang Giới Thiệu

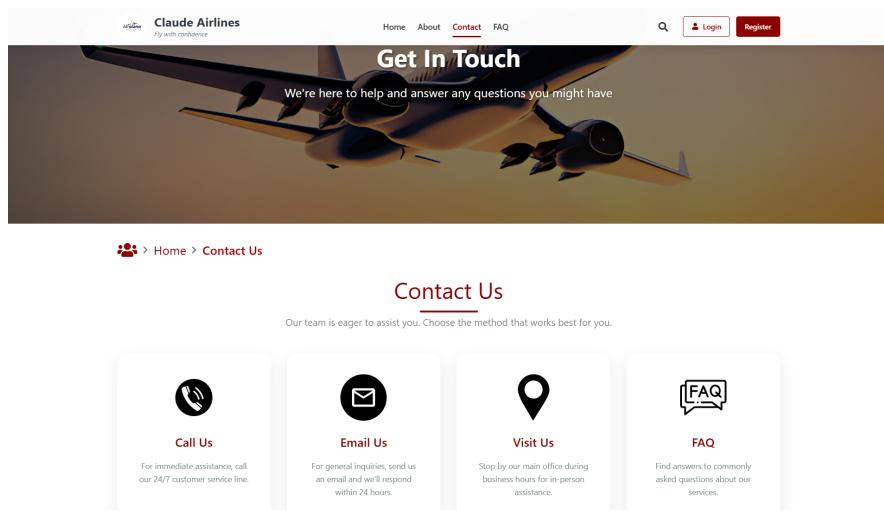


4.2.4.3 Chức năng chính

- **Tải dữ liệu động:** Sử dụng ‘useEffect’ để gọi API và lấy dữ liệu cho các phần thông tin chung, giá trị cốt lõi, đội ngũ lãnh đạo, thành tựu và trải nghiệm bay khi component được render.
- **Hiển thị thông tin:** Trình bày dữ liệu một cách có cấu trúc và trực quan thông qua các section và component con.
- **Điều hướng người dùng:** Cung cấp các liên kết nội bộ đến các trang liên quan.



4.2.5 Trang Liên Hệ (Contact Us)



Hình 12: Giao diện Trang Liên Hệ

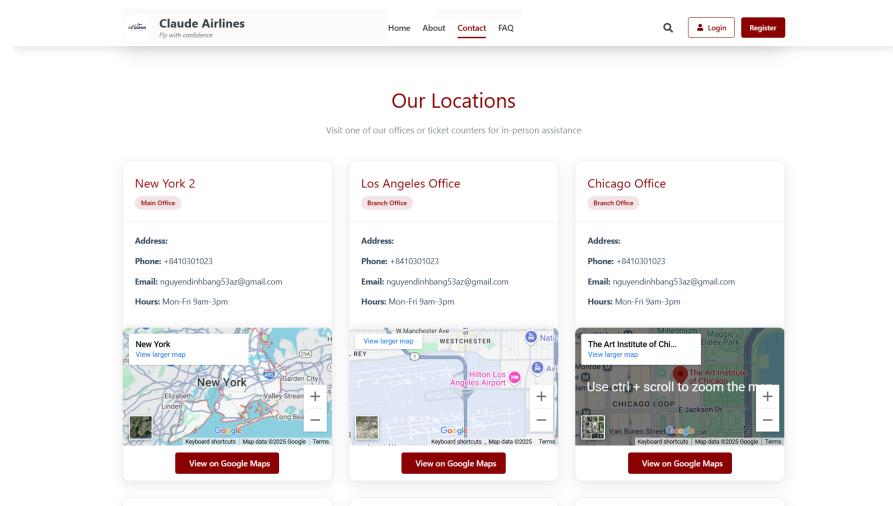
4.2.5.1 Mô tả chung Trang Liên Hệ ('ContactPage.js') cung cấp nhiều phương thức để người dùng kết nối với Claude Airlines, từ thông tin liên hệ trực tiếp, biểu mẫu gửi tin nhắn, đến danh sách các văn phòng và bản đồ.

4.2.5.2 Các thành phần giao diện chính

- **Hero Section:** Hình ảnh nền lớn với tiêu đề "Get In Touch" và mô tả ngắn, tạo ấn tượng ban đầu.
- **Breadcrumbs:** Thanh điều hướng (Home > Contact Us).
- **Các Phương Thức Liên Hệ Nhanh ('ContactUnit.js'):**
 - Hiển thị các khối thông tin trực quan (Call Us, Email Us, Visit Us, FAQ) với icon, mô tả và nút hành động (ví dụ: số điện thoại, link email, link đến mục địa điểm, link đến trang FAQ).
- **Biểu Mẫu Liên Hệ ('ContactForm.js'):**
 - Cho phép người dùng gửi tin nhắn trực tiếp.
 - Các trường: Full Name (bắt buộc), Email (bắt buộc, có kiểm tra định dạng), Phone Number (tùy chọn), Subject (bắt buộc), Message (bắt buộc).
 - Xác thực dữ liệu phía client và hiển thị thông báo lỗi.
 - Sau khi gửi thành công, hiển thị thông báo "Thank You!" và tự động ẩn sau 5 giây.
 - Bên cạnh form là thông tin văn phòng chính, giờ làm việc và các icon mạng xã hội.
- **Danh Sách Địa Điểm Văn Phòng ('contactLocations'):**
 - Tải động danh sách các địa điểm văn phòng từ API.



- Hiển thị dưới dạng thẻ (card), mỗi thẻ bao gồm: tên địa điểm, loại văn phòng, địa chỉ, SDT, email, giờ làm việc.
 - Tích hợp bản đồ Google Maps cho từng địa điểm bằng component ‘LocationMap.js’, cho phép xem trực tiếp trên trang hoặc mở trên Google Maps.
 - Nếu không có dữ liệu từ API, sẽ hiển thị một số địa điểm mẫu.
- **Bản Đồ Tổng Quan:** Một bản đồ Google Maps lớn hiển thị vị trí (ví dụ: sân bay chính hoặc trụ sở).
 - **Kêu Gọi Hành Động (Call to Action):** Phần cuối trang với các nút "Call Now" và "Send Message" để khuyến khích người dùng liên hệ ngay.



Hình 13: Giao diện Trang Liên Hệ

4.2.5.3 Chức năng chính

- **Cung cấp đa dạng kênh liên hệ:** Điện thoại, email, biểu mẫu trực tuyến, địa chỉ văn phòng.
- **Tải dữ liệu động:** Lấy danh sách các địa điểm văn phòng từ backend.
- **Xác thực biểu mẫu liên hệ:** Đảm bảo người dùng nhập đủ và đúng thông tin cần thiết trước khi gửi.
- **Hiển thị bản đồ tương tác:** Giúp người dùng dễ dàng tìm vị trí các văn phòng.



4.2.6 Trang Câu Hỏi Thường Gặp (FAQ)

The screenshot shows the Claude Airlines FAQ page. At the top, there's a navigation bar with links for Home, About, Contact, and FAQ. Below the navigation is a search bar and a login/register section. The main content area is titled "Frequently Asked Questions" and includes a search bar for finding answers to common questions about services. Below this, there are several categories of questions represented by buttons: All, General, Technical, Billing, Booking, Baggage (which is highlighted in red), Reservation, Membership, Accessibility, Health & Safety, and Travel Guidelines. Three specific questions are expanded: "What is your baggage allowance policy?", "Can I bring sports equipment as checked baggage?", and "What items are prohibited in carry-on bags?". Each question has a detailed answer below it.

Hình 14: Giao diện Trang FAQ

4.2.6.1 Mô tả chung Trang FAQ ('FAQPage.js') cung cấp một danh sách các câu hỏi thường gặp và câu trả lời tương ứng, giúp người dùng nhanh chóng tìm thấy thông tin họ cần. Trang được thiết kế với các tính năng lọc, tìm kiếm và phân trang để nâng cao trải nghiệm người dùng.

4.2.6.2 Các thành phần giao diện chính

- Header Section và Breadcrumbs:** Tiêu đề "Frequently Asked Questions", mô tả ngắn và thanh điều hướng (Home > FAQ).
- Thanh Tìm Kiếm:** Cho phép người dùng nhập từ khóa để tìm kiếm câu hỏi. Kết quả được cập nhật động.
- Bộ Lọc Danh Mục:** Các nút danh mục (ví dụ: "All", "Booking", "Payment", "Services") cho phép người dùng lọc câu hỏi theo chủ đề. Danh sách danh mục được tạo động từ dữ liệu FAQ.
- Danh Sách Câu Hỏi ('FAQItem.js'):**
 - Mỗi câu hỏi và câu trả lời được hiển thị trong một component 'FAQItem'.
 - Thiết kế dạng accordion: người dùng nhấp vào câu hỏi để mở rộng và xem câu trả lời.
 - Sử dụng icon 'FaChevronDown' / 'FaChevronUp' để chỉ trạng thái mở/dóng.
 - Có hiệu ứng chuyển động mượt mà khi mở/dóng.
- Phân Trang ('Pagination.js'):**
 - Nếu có nhiều câu hỏi, chúng sẽ được chia thành nhiều trang.
 - Component 'Pagination' hiển thị các nút số trang, nút "Previous" và "Next".
 - Logic phân trang tính toán số trang dựa trên 'itemsPerPage'.
- Kêu Gọi Hành Động (Call to Action):** Phần "Still have questions?" với các nút "Contact Us" và "Call Us" để người dùng liên hệ trực tiếp nếu không tìm thấy câu trả lời.



4.2.6.3 Chức năng chính

- **Tải dữ liệu động:** Lấy danh sách câu hỏi và câu trả lời từ API ('faq.getAllFAQs()').
- **Tìm kiếm động:** Lọc danh sách câu hỏi dựa trên từ khóa người dùng nhập.
- **Lọc theo danh mục:** Cho phép người dùng xem câu hỏi thuộc một chủ đề cụ thể.
- **Hiển thị dạng Accordion:** Giúp giao diện gọn gàng và người dùng dễ dàng tập trung vào câu hỏi họ quan tâm.
- **Phân trang hiệu quả:** Xử lý số lượng lớn câu hỏi mà không làm quá tải giao diện.



5 Cài đặt và triển khai

5.1 Yêu cầu môi trường

- Ngôn ngữ backend: PHP 8.x
- Cơ sở dữ liệu: MySQL 5.7 hoặc cao hơn
- Frontend: Node.js v16.x trở lên, npm v8.x
- Công cụ hỗ trợ:
 - XAMPP (hoặc tương đương) để chạy server PHP và MySQL
 - Git để clone mã nguồn
 - Trình duyệt hiện đại (Google Chrome, Firefox)

5.2 Hướng dẫn cài đặt ứng dụng

5.2.1 Bước 1: Tải mã nguồn

Mở terminal và chạy lệnh sau để sao chép toàn bộ mã nguồn:

```
git clone https://github.com/bawfng04/ClaudeAirlines
```

5.2.2 Bước 2: Cài đặt và chạy frontend

1. Di chuyển vào thư mục frontend
2. Cài đặt thư viện bằng lệnh: `npm install`
3. Chạy frontend: `npm start`
4. Ứng dụng sẽ chạy tại `http://localhost:3000`

5.2.3 Bước 3: Cài đặt và chạy backend

1. Mở phần mềm XAMPP, bật Apache và MySQL
2. Tạo cơ sở dữ liệu và thêm dữ liệu:
 - Vào `http://localhost/phpmyadmin`
 - Nhấn tab **Import**, sau đó:
 - Import file `backend/sql/table.sql` để tạo bảng
 - Import tiếp file `backend/sql/insert.sql` để chèn dữ liệu mẫu.
3. Trong thư mục `backend/app`, sửa file `.env.example` thành `.env` và điền đầy đủ các cấu hình hệ thống: cơ sở dữ liệu, khóa bảo mật,... theo yêu cầu dự án:

```
DB_HOST=
DB_PORT=
DB_USER=
DB_PASS=
DB_NAME=claude_airlines
JWT_SECRET=
BASE_URL=(nếu webroot trả đến thư mục backend thì BASE_URL sẽ là: /backend/public/)
```



4. Trong thư mục **frontend**, sửa file **.env.example** thành **.env** và điền đầy đủ các cấu hình hệ thống: đường dẫn API, ... theo yêu cầu dưới đây:

REACT_APP_BASE_URL=

5. Đảm bảo XAMPP đã được cấu hình để trỏ webroot đến thư mục **backend/public** của dự án.



References

- [1] Claude Airlines's Source Code <https://github.com/bawfng04/ClaudeAirlines>
- [2] Facebook, Inc. *React – A JavaScript library for building user interfaces.* <https://react.dev/>
- [3] Remix Software, Inc. *React Router DOM - Declarative routing for React.* <https://reactrouter.com/en/main>
- [4] Tailwind Labs Inc. *Tailwind CSS - Rapidly build modern websites without ever leaving your HTML.* <https://tailwindcss.com/docs>
- [5] Zuramai. *Mazer Admin Dashboard Free and Open Source Bootstrap 5 Admin Dashboard Template and Landing Page.* <https://github.com/zuramai/mazer>
- [6] Google LLC. *Firebase Hosting Documentation.* <https://firebase.google.com/docs/hosting>
- [7] The PHP Group. *PHP Manual.* <https://www.php.net/manual/en/>
- [8] Oracle Corporation and/or its affiliates. *MySQL Documentation.* <https://dev.mysql.com/doc/>
- [9] OWASP Foundation. *OWASP Top 10 Web Application Security Risks.* <https://owasp.org/www-project-top-ten/>