

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)

Xây dựng Website Hàng hàng không Claude Airline

Nhóm: 8 - L01

GVHD: Nguyễn Hữu Hiếu

SVTH: Nguyễn Dinh Bằng - 2210298 – bang.nguyendinh@hcmut.edu.vn
Trần Minh Quân - 2212822 – quan.tranminh411@hcmut.edu.vn
Lê Anh Khoa – 2211595



Phân chia công việc

Họ và tên	Nhiệm vụ	Tỉ lệ đóng góp
Nguyễn Dinh Băng, Trần Minh Quân	<ul style="list-style-type: none">- Thiết kế mô hình ứng dụng- Thiết kế các mẫu (template) chung cho website- Giao diện và tính năng đăng ký / đăng nhập, phân quyền người dùng- Tính năng thay đổi thông tin, mật khẩu, avatar cho người dùng- Tính năng quản lý người dùng dành cho quản trị viên	—
Nguyễn Dinh Băng	<ul style="list-style-type: none">- Giao diện: Trang chủ, Trang liên hệ- Tính năng quản lý: Quản lý thông tin trên các trang đã thiết kế (giới thiệu, số điện thoại, địa chỉ công ty, hình ảnh, logo,...)- Quản lý các liên hệ của khách hàng (xem thông tin, đánh dấu đã đọc/chưa đọc/đã phản hồi, xoá liên hệ,...)- Viết báo cáo	100%
Trần Minh Quân	<ul style="list-style-type: none">- Giao diện: Trang giới thiệu, Trang hỏi đáp- Tính năng quản lý: Quản lý thông tin trên các trang đã thiết kế (giới thiệu, hình ảnh,...)- Quản lý các câu hỏi/đáp (xem, thêm, sửa, xoá)- Viết báo cáo	100%
Lê Anh Khoa	<ul style="list-style-type: none">- Giao diện: Trang danh sách bài viết, Trang đọc bài viết- Tính năng quản lý: Quản lý tin tức (xem/tìm kiếm, thêm, sửa, xoá bài viết)- Quản lý bình luận/danh giá của người dùng trên bài viết- Viết báo cáo	100%



Contents

1 Giới thiệu	4
1.1 Giới thiệu về Công ty/Doanh nghiệp	4
1.2 Mục tiêu xây dựng website	5
2 Cơ sở lý thuyết	6
2.1 Các công nghệ và thư viện sử dụng	6
2.1.1 Frontend	6
2.1.2 Backend	7
2.1.3 Giao diện quản trị (Admin Panel)	8
2.2 Ưu điểm và nhược điểm của từng công nghệ	8
2.3 Các lỗ hổng bảo mật thường gặp trong ứng dụng web	9
2.4 Các yếu tố tối ưu hóa SEO trong ứng dụng	11
3 Thiết kế ứng dụng	13
3.1 Thiết kế cơ sở dữ liệu	13
3.1.1 Entity Relationship Diagram	13
3.1.2 Cấu trúc và chức năng của các bảng dữ liệu	14
3.1.3 Ánh xạ cơ sở dữ liệu thành dạng bảng	18
3.2 Mô hình ứng dụng	19
3.3 Cấu trúc mã nguồn	21
3.4 Các tính năng chính của hệ thống	22
3.4.1 Tính năng người dùng	22
3.4.2 Tính năng dành cho quản trị viên	22
3.4.3 Giao diện người dùng	23
3.5 Các sơ đồ thể hiện các quy trình chính	24
3.5.1 Sơ đồ UseCase cho toàn bộ hệ thống	24
3.5.2 Sơ đồ Activity cho hoạt động quản lý của Admin	26
4 Hiện thực	28
4.1 Các tính năng đã hiện thực	28
4.1.1 Tính năng phía Người dùng (Client-side)	28
4.1.2 Tính năng phía Quản trị viên (Admin-side)	29
4.1.3 Tính năng Kỹ thuật và Chung	30
4.2 Demo giao diện	31
4.2.1 Trang Đăng Nhập (Login)	31
4.2.2 Trang Đăng Ký (Register)	32
4.2.3 Trang Chủ (Homepage)	33
4.2.4 Trang Giới Thiệu (About Us)	36
4.2.5 Trang Liên Hệ (Contact Us)	40
4.2.6 Trang Câu Hỏi Thường Gặp (FAQ)	42
4.2.7 Trang Danh sách Bài đăng Vlog	44
4.2.8 Trang Chi tiết Bài đăng Vlog (Vlog Post Page)	47



5 Cài đặt và triển khai	51
5.1 Yêu cầu môi trường	51
5.2 Hướng dẫn cài đặt ứng dụng	51
5.2.1 Bước 1: Tải mã nguồn	51
5.2.2 Bước 2: Cài đặt và chạy frontend	51
5.2.3 Bước 3: Cài đặt và chạy backend	51
6 Kết luận	53



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.



Hình 1: Biểu tượng logo Claude Airline



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.



- **Swiper:** Thư viện JavaScript mạnh mẽ dùng để tạo các carousel, slideshow với hiệu ứng mượt mà và hỗ trợ cảm ứng. Thư viện này rất phù hợp cho các ứng dụng web mobile và desktop, với tính năng responsive và khả năng tùy chỉnh cao.

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	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.
Swiper	Hỗ trợ cảm ứng, hiệu ứng đẹp, dễ tích hợp trong React.	Dung lượng hơi lớn nếu chỉ dùng cho hiệu ứng đơn giản.
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 khiến kẻ 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 lõi

```
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 Airline 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).



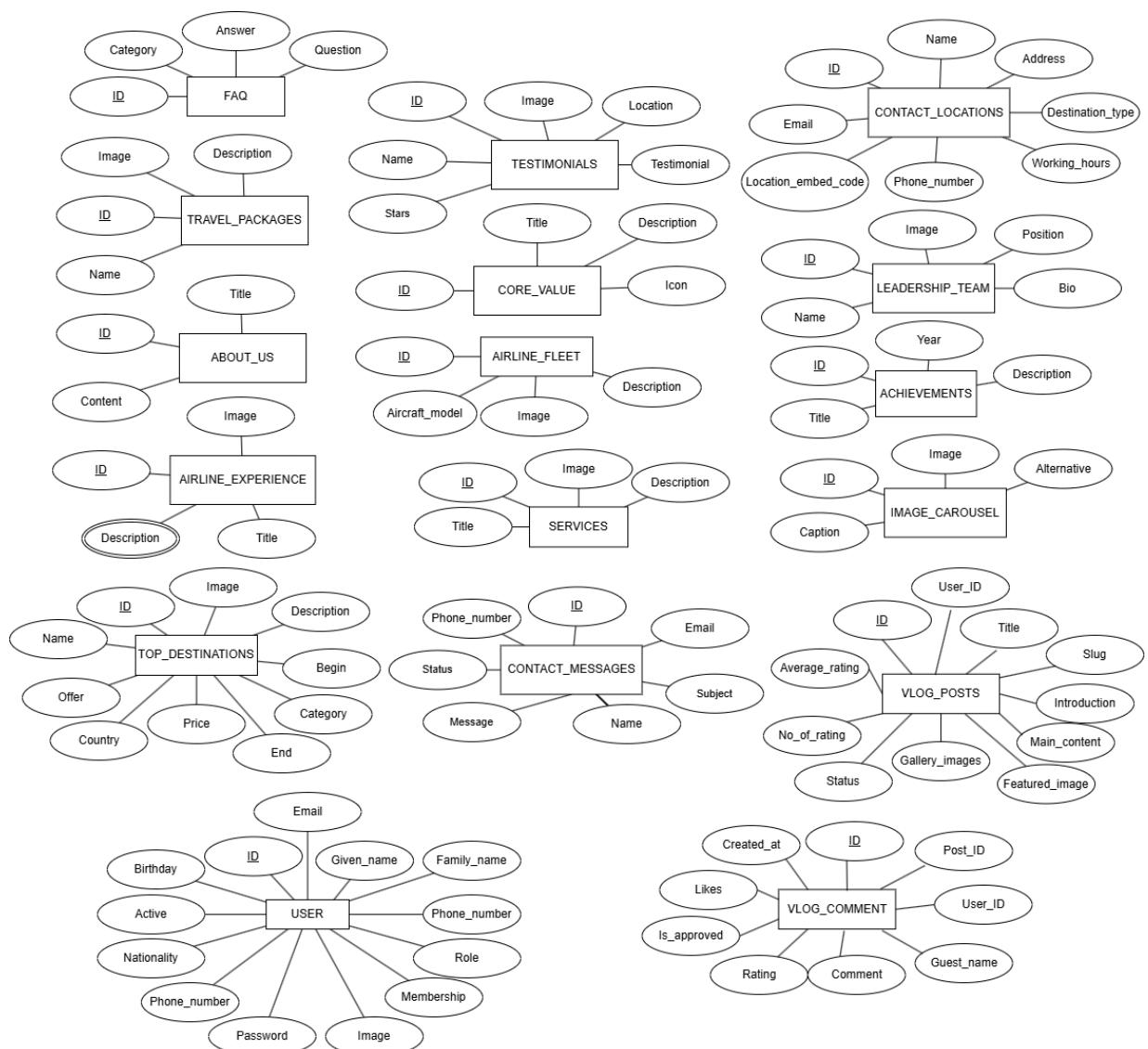
- **Cấu hình sitemap.xml và robots.txt:** Các file này được đặt trong thư mục `public` của frontend. `sitemap.xml` giúp công cụ tìm kiếm khám phá tất cả các trang quan trọng của ứng dụng, trong khi `robots.txt` chỉ dẫn cho các trình thu thập dữ liệu về các phần của trang web mà chúng nên hoặc không nên 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 2: 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 CONTACT_LOCATIONS

1. **Mô tả:** Lưu các địa điểm liên hệ của hàng 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.
 - description: Lưu mô tả các trải nghiệm dịch vụ, là thuộc tính đa trị, not null.

Bảng CONTACT_MESSAGES

1. **Mô tả:** Lưu trữ các tin nhắn liên hệ được gửi từ người dùng thông qua form liên hệ trên website.
2. **Đặc tả bảng:**
 - id, name, email, message, subject: not null.
 - phone.
 - status: Lưu trạng thái xử lý tin nhắn - enum(unread, read, replied), mặc định = unread.



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.

Bảng VLOG_POSTS

1. **Mô tả:** Lưu trữ các bài đăng vlog và thông tin liên quan.
2. **Đặc tả bảng:**
 - id: Khóa chính, tự tăng.
 - user_id, title, slug, main_content, status: Các thông tin cơ bản bắt buộc của bài đăng; slug duy nhất; user_id tham chiếu đến USERS.



- **introduction, featured_image, gallery_images:** Nội dung bổ sung (tóm tắt, ảnh đại diện, thư viện ảnh).
- **no_of_ratings, average_rating:** Dữ liệu tổng hợp về đánh giá.
- **created_at, updated_at:** Thời điểm tạo và cập nhật.

Bảng VLOG _ COMMENTS

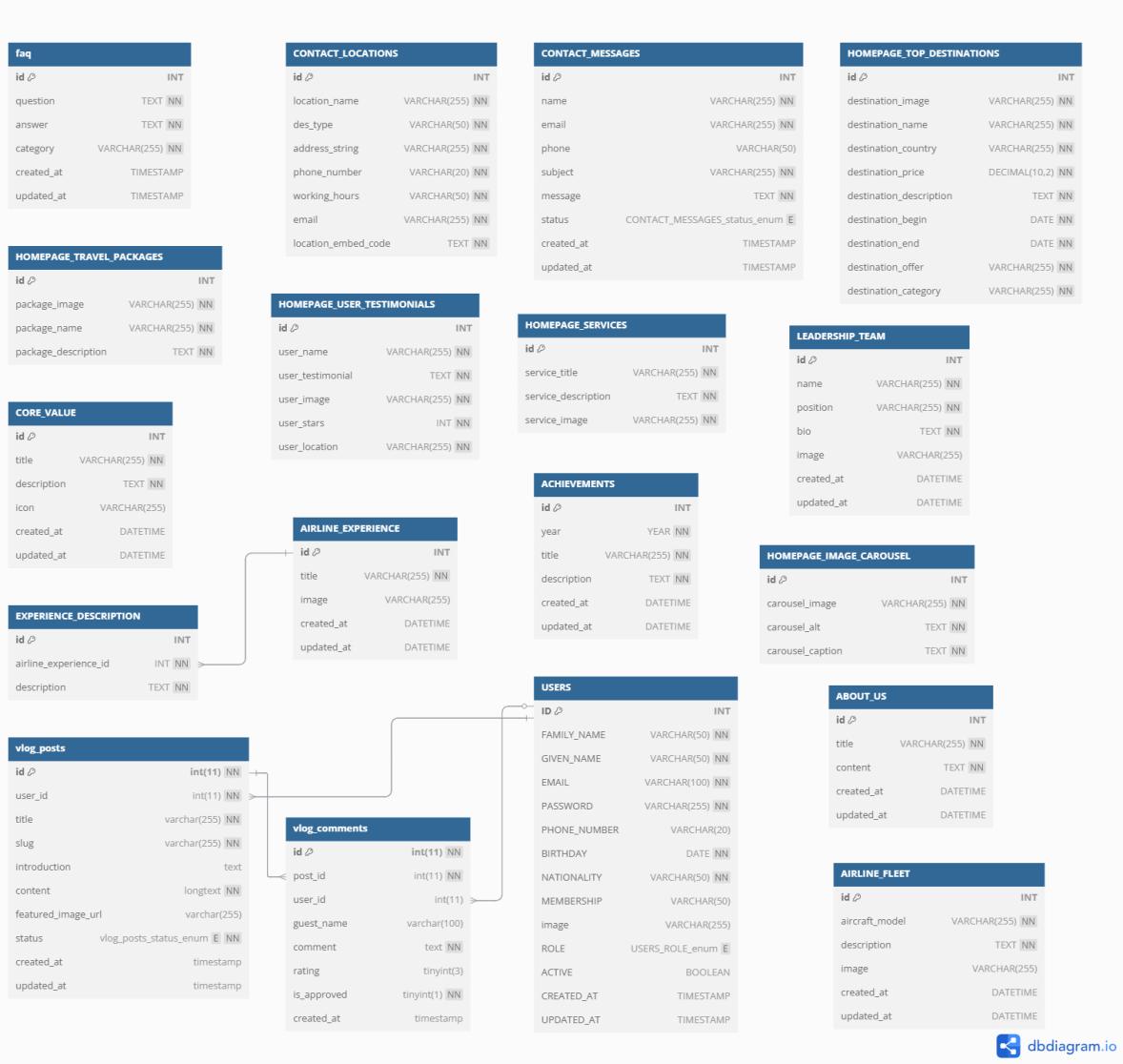
1. **Mô tả:** Lưu trữ bình luận và đánh giá cho các bài đăng vlog.

2. **Đặc tả bảng:**

- **id:** Khóa chính, tự tăng.
- **post_id, comment, is_approved, likes:** Thông tin chính của bình luận; post_id tham chiếu đến vlog_posts; is_approved, likes có giá trị mặc định.
- **user_id, guest_name:** Thông tin người bình luận (ID người dùng hoặc tên khách).
- **rating:** Điểm đánh giá (từ 1-5), có thể null.
- **created_at:** Thời điểm tạo bình luận.



3.1.3 Ánh xạ cơ sở dữ liệu thành dạng bảng



Hình 3: 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.

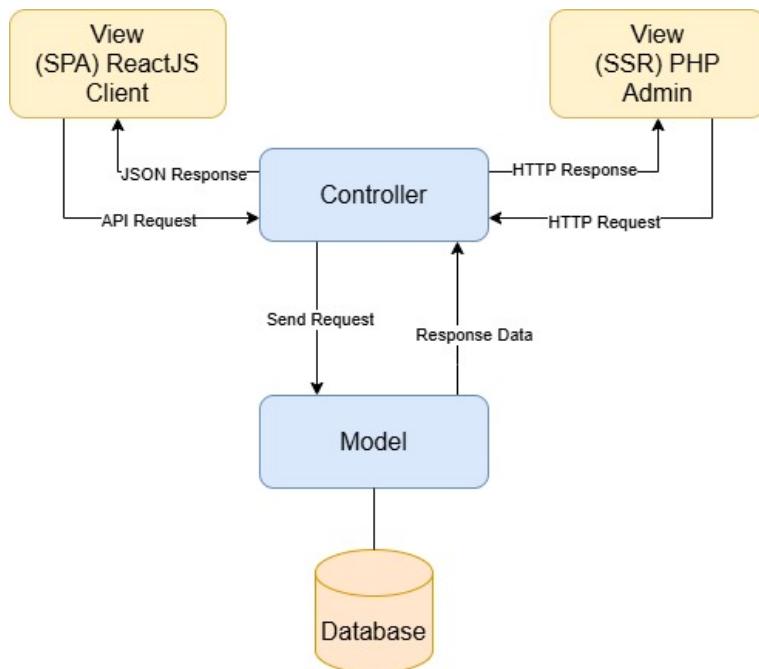
3. View (V) – Giao diện hiển thị dữ liệu

(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 4: 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.
- **Đọc và tìm kiếm bài đăng:** Người dùng có thể truy cập phần đăng bài quảng bá du lịch của doanh nghiệp và tìm kiếm bài đăng theo từ khóa.
- **Bình luận và đánh giá các bài đăng:** Người dùng được phép tìm kiếm, để lại bình luận và đánh giá các bài đăng từ 1 đến 5 sao. Người dùng chưa đăng nhập cần nhập tên vào ô thông tin trước khi được đăng bình luận, đánh giá.

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/Đáp

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.
- Đánh dấu liên hệ đã đọc/chưa đọc/dã phản hồi.
- Xóa liên hệ không cần thiết.



- **Quản lý hỏi/đáp (FAQ):**

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

- **Quản lý các bài đăng:**

- Xem, thêm mới, xóa và chỉnh sửa các nội dung của bài viết (tiêu đề, nội dung, ảnh, trạng thái duyệt bài viết).
 - Các bài viết cần được gắn Author_ID là user_ID của tác giả được phép truy cập vào trang quản lý.

- **Quản lý bình luận của người dùng:**

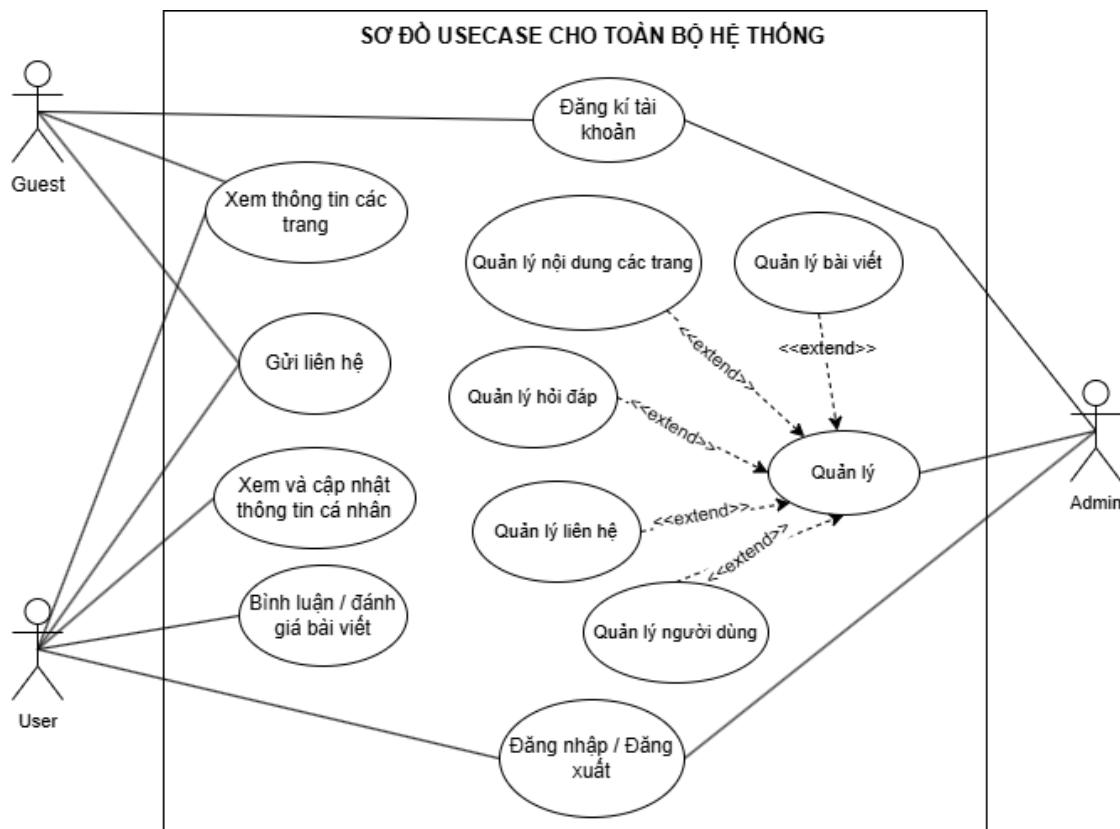
- Xem, duyệt hoặc phủ quyết từng bình luận của một bài đăng nào đó, xóa bình luận trong database.

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 Bài viết:** Các thông tin và bài đăng quảng bá du lịch của doanh nghiệp.
- **Trang Hỏi/Đáp:** Danh sách các câu hỏi và trả lời phổ biến.

3.5 Các sơ đồ thể hiện các quy trình chính

3.5.1 Sơ đồ UseCase cho toàn bộ hệ thống



Hình 5: Sơ đồ UseCase cho toàn bộ hệ thống

Sơ đồ UseCase này mô tả tổng quan các tương tác chính giữa các tác nhân (Actor) và hệ thống website Claude Airline. Có ba tác nhân chính được xác định:

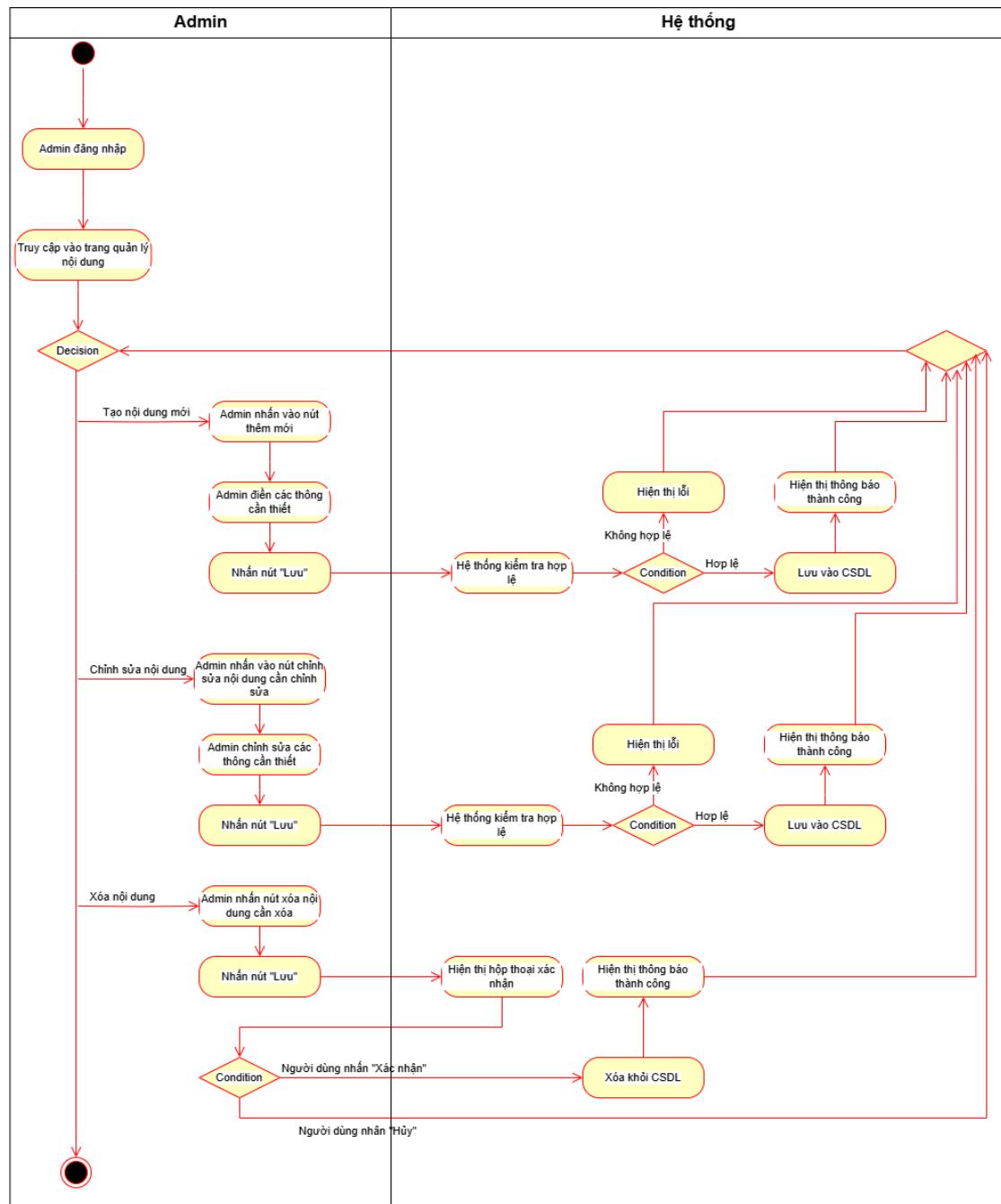
- Guest (Người dùng khách):** Đại diện cho người dùng chưa đăng nhập vào hệ thống. Guest có thể thực hiện các hành động cơ bản như *Xem thông tin các trang* (ví dụ: trang chủ, giới thiệu, liên hệ, FAQ) và *Đăng ký tài khoản* mới.
- User (Người dùng đã đăng nhập):** Đại diện cho người dùng đã có tài khoản và đăng nhập thành công. Ngoài các quyền của Guest, User có thêm các khả năng như: *Gửi liên hệ*, *Xem và cập nhật thông tin cá nhân*, *Bình luận/đánh giá bài viết* (nếu có tính năng blog/vlog), và thực hiện hành động *Đăng nhập/Đăng xuất*.
- Admin (Quản trị viên):** Đại diện cho người có quyền quản trị cao nhất trong hệ thống. Admin có thể thực hiện tất cả các chức năng của User, đồng thời có các quyền quản lý mở rộng thông qua một giao diện quản trị riêng. Các UseCase quản lý này bao gồm *Quản lý nội dung các trang*, *Quản lý bài viết*, *Quản lý hỏi đáp (FAQ)*, *Quản lý liên hệ từ khách hàng*, và *Quản lý người dùng*. Các UseCase này được biểu thị bằng mối quan hệ «extend»



từ UseCase *Quản lý chung*, cho thấy chúng là các chức năng mở rộng và chuyên biệt của việc quản lý hệ thống.

Sơ đồ này giúp hình dung rõ ràng phạm vi chức năng của hệ thống và cách các nhóm người dùng khác nhau tương tác với nó.

3.5.2 Sơ đồ Activity cho hoạt động quản lý của Admin



Hình 6: Sơ đồ Activity cho hoạt động quản lý

Sơ đồ Activity mô tả chi tiết luồng hoạt động của Quản trị viên (Admin) khi thực hiện các tác vụ quản lý nội dung trên website Claude Airline. Luồng hoạt động bắt đầu sau khi Admin Đăng



nhập thành công và *Truy cập vào trang quản lý nội dung*.

Tại đây, Admin đứng trước một điểm quyết định (Decision node) với ba lựa chọn chính:

1. Tạo nội dung mới:

- Admin nhấp vào nút "Thêm mới".
- Hệ thống hiển thị form để Admin điền các thông tin cần thiết cho nội dung mới.
- Admin nhấp nút "Lưu".
- Hệ thống kiểm tra tính hợp lệ của dữ liệu.
 - Nếu Không hợp lệ, hệ thống hiển thị lỗi và quay lại bước điền thông tin.
 - Nếu Hợp lệ, hệ thống lưu vào CSDL (Cơ sở dữ liệu) và hiển thị thông báo thành công.

2. Chính sửa nội dung hiện có:

- Admin nhấp vào nút "Chỉnh sửa" của một nội dung cụ thể.
- Hệ thống hiển thị form với thông tin hiện tại để Admin chỉnh sửa các thông tin cần thiết.
- Admin nhấp nút "Lưu".
- Tương tự như khi tạo mới, hệ thống kiểm tra tính hợp lệ, sau đó hiển thị lỗi hoặc lưu vào CSDL và báo thành công.

3. Xóa nội dung:

- Admin nhấp vào nút "Xóa" của một nội dung.
- Hệ thống hiển thị hộp thoại xác nhận việc xóa.
- Admin đưa ra quyết định (Condition node):
 - Nếu Người dùng nhấn "Xác nhận", hệ thống sẽ Xóa khỏi CSDL và hiển thị thông báo thành công.
 - Nếu Người dùng nhấn "Hủy", luồng hoạt động kết thúc cho tác vụ xóa này.

Sau khi hoàn thành mỗi tác vụ (hoặc hủy bỏ), Admin có thể tiếp tục thực hiện các thao tác quản lý khác hoặc kết thúc phiên làm việc. Sơ đồ này làm rõ các bước xử lý và các điểm rẽ nhánh trong quy trình quản lý nội dung của Admin.



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 lớp 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 Airline, được lấy động từ backend qua API, bao gồm:
 - Thông tin chung về công ty.
 - Giá trị cốt lõi của Claude Airline.
 - 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 Airline, 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.
- Trang Danh sách Bài đăng Vlog:
 - Hiển thị các bài đăng vlog dưới dạng lưới, hỗ trợ tìm kiếm và phân trang.
 - Tải dữ liệu bài đăng từ API, có hiệu ứng chờ và thông báo lỗi.
 - Giao diện bao gồm hero section, mục "Ever Wondered?" và trình chiếu "Thematic Adventures".
 - Tối ưu SEO cho tiêu đề và mô tả trang.
- Trang Chi tiết Bài đăng Vlog:
 - Hiển thị nội dung chi tiết bài đăng: tiêu đề, ảnh đại diện, tác giả, ngày đăng, nội dung, thư viện ảnh.
 - Hệ thống bình luận và đánh giá: cho phép người dùng (khách và đã đăng nhập) gửi bình luận, đánh giá sao. Bình luận chờ duyệt, hiển thị các bình luận đã duyệt, hỗ trợ thích bình luận.
 - Chức năng chia sẻ bài đăng lên mạng xã hội.
 - Tải động nội dung bài đăng và bình luận từ API.
 - Tối ưu SEO với các thẻ meta phù hợp.
 - Có mục đăng ký nhận bản tin.

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 Dội 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.
- Quản lý Bài đăng Vlog: Cung cấp giao diện xem danh sách bài đăng dạng bảng, có tìm kiếm và phân trang. Quản trị viên có thể thêm mới hoặc chỉnh sửa bài đăng thông qua form modal, bao gồm các thông tin như tiêu đề, tác giả, trạng thái, nội dung, quản lý hình ảnh đại diện (tải lên, xem trước, xóa) và thư viện hình ảnh (thêm URL, sắp xếp, xóa). Chức năng xóa bài đăng (kèm xóa bình luận liên quan) cũng được hỗ trợ. Các API backend đảm nhận việc xử lý CRUD cho bài đăng.
- Quản lý Bình luận Vlog: Hiển thị danh sách bình luận dưới dạng bảng, hỗ trợ tìm kiếm và phân trang. Thông tin bao gồm người bình luận, nội dung, bài đăng liên quan, đánh giá, lượt thích, trạng thái. Quản trị viên có thể thực hiện duyệt, hủy duyệt, và xóa bình luận. Các API backend phục vụ các thao tác quản lý này.

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 7: 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 8: 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, SĐT, 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.

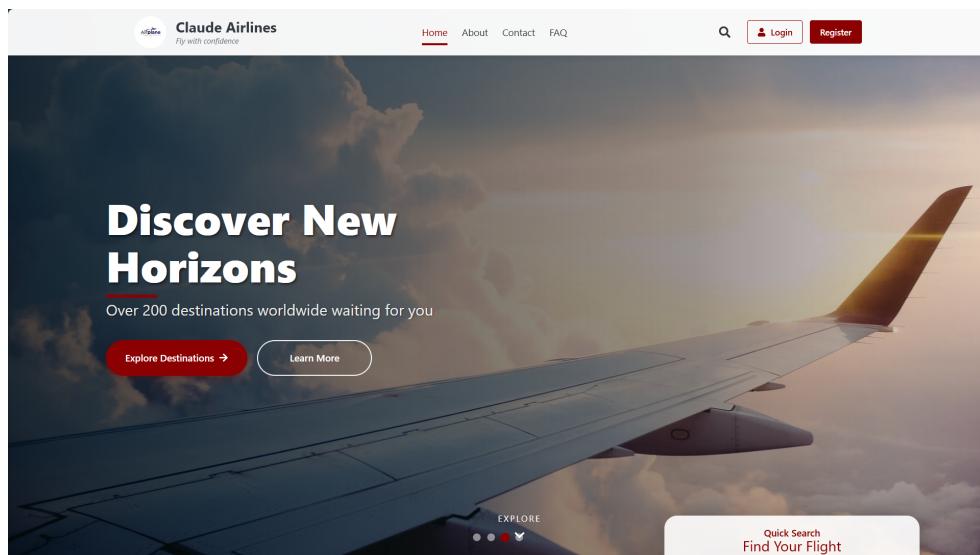


4.2.2.3 Chức năng chính

- **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)

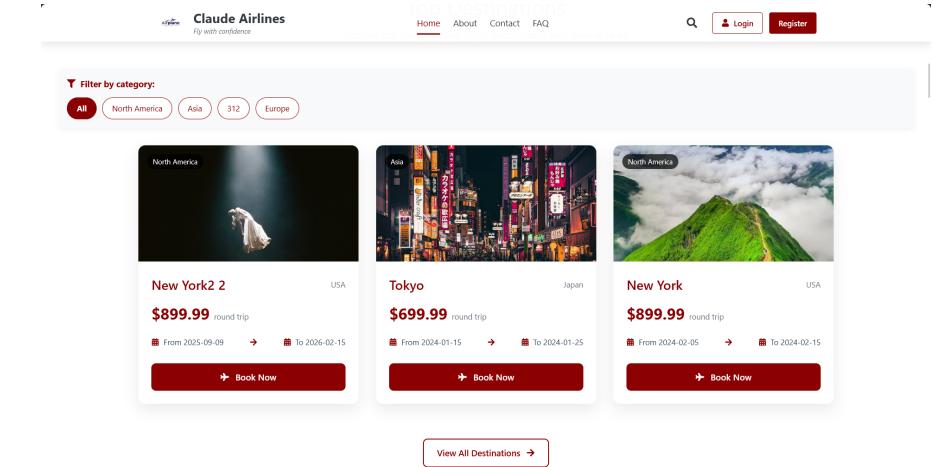
4.2.3.1 Mô tả chung Trang Chủ (Homepage.js) là giao diện chính của Claude Airline, được thiết kế để cung cấp tổng quan về dịch vụ, điểm đến và ưu đãi, thu hút người dùng ngay từ cái nhìn đầu tiên.



Hình 9: Giao diện Trang Chủ

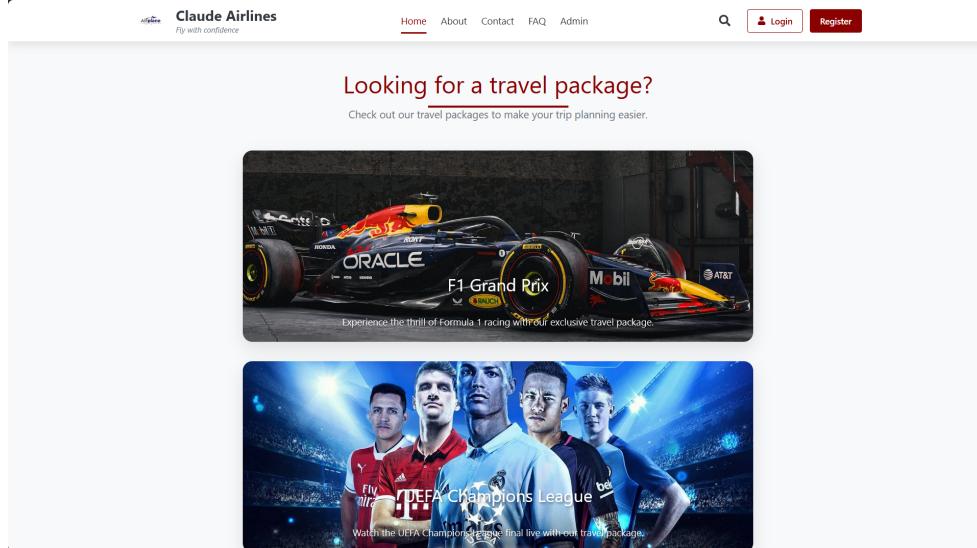
4.2.3.2 Các thành phần chính và chức năng

- **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.



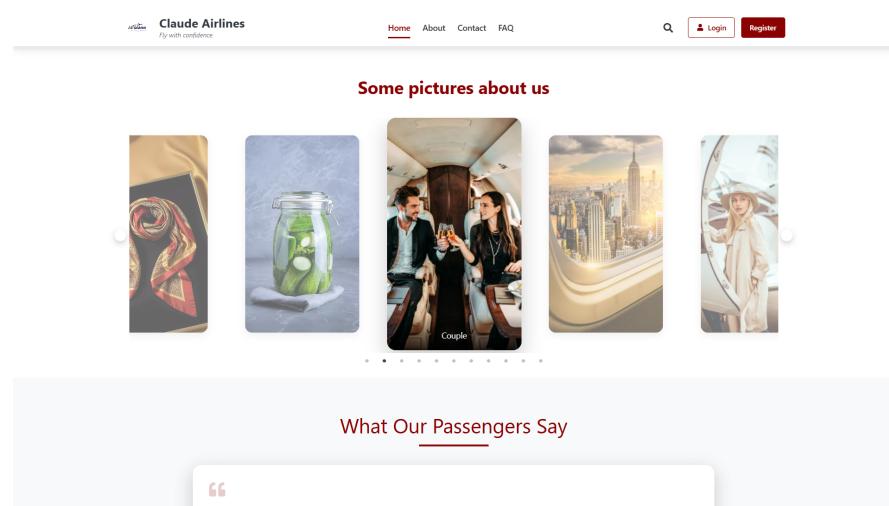
Hình 10: Giao diện Các điểm đến nổi bật

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



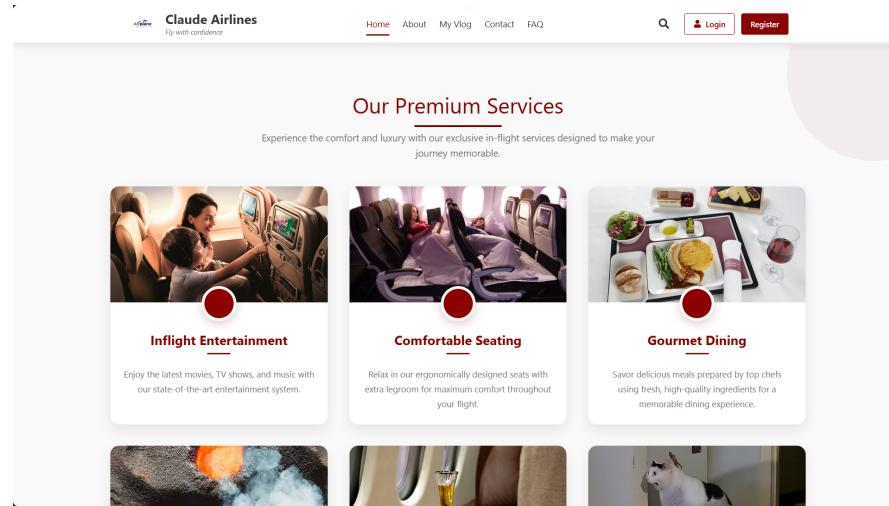
Hình 11: Giao diện Các gói du lịch (travel packages)

- **Trình Chiếu Hình Ảnh (ImageCarousel.js):** Sử dụng 'react-slick' để tạo carousel hiển thị các hình ảnh giới thiệu về hãng, dịch vụ, hoặc điểm đến, được tải động từ API.



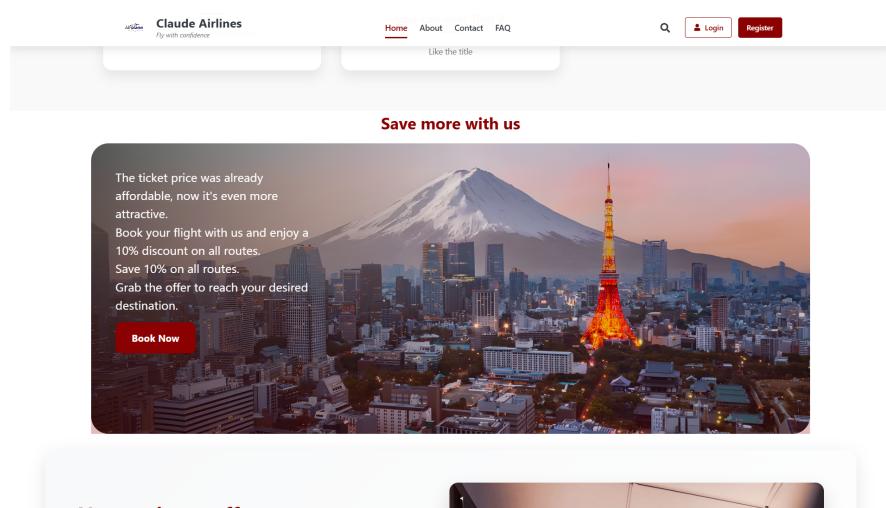
Hình 12: Giao diện Thư viện ảnh ở trang chủ

- **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.



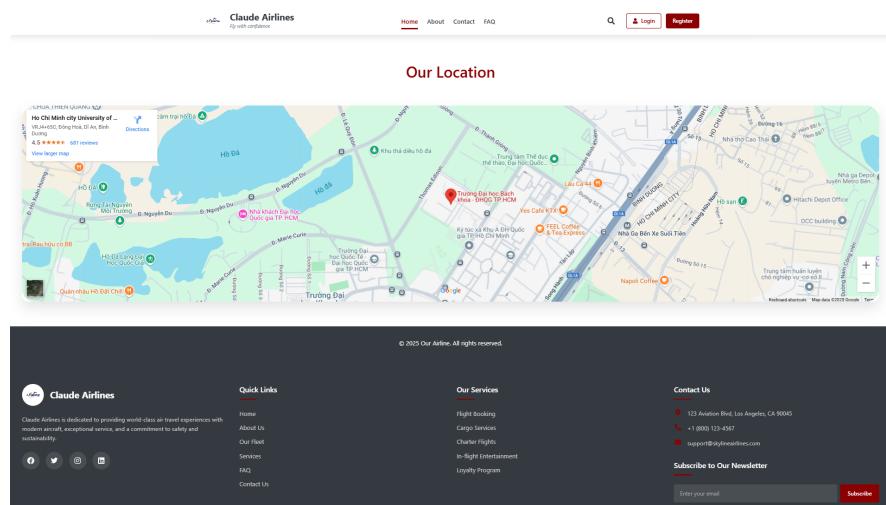
Hình 13: Giao diện Danh mục dịch vụ

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



Hình 14: Giao diện Danh mục khuyến mãi phụ

- **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.



Hình 15: Giao diện Danh mục bản đồ

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 Airline, bao gồm lịch sử, sứ mệnh, giá trị cốt lõi, đội ngũ, các thành tựu và các loại máy bay được sử dụng. 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".
- **Trang phụ:** Được truy cập khi nhấn vào thẻ **Our Fleet** sẽ được điều hướng tới trang Our Fleet ('planes.js') giới thiệu đội bay hiện tại của hãng, cung cấp thông tin chi tiết về từng loại máy bay được sử dụng trong các chuyến bay thương mại, giúp người dùng dễ dàng khám phá và tìm hiểu về các dòng máy bay của hãng.

Hình 16: Giao diện Trang Giới Thiệu



The screenshot shows a timeline of awards from 2021 to 2024. Each award is presented in a box with a year header and a small image.

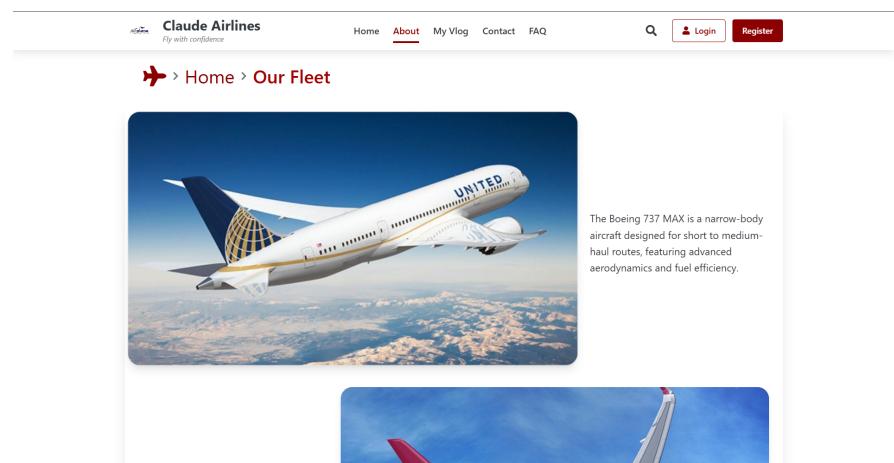
- 2024:** Best Airline in Asia. Awarded by Skytrax World Airline Awards for the second consecutive year.
- 2023:** Best Airline in Southeast Asia. Awarded by Skytrax World Airline Awards for the third consecutive year.
- 2022:** Top 10 Airlines Worldwide. Ranked by AirlineRatings.com for safety and service excellence.
- 2021:** Claude Airline of the Year.

Hình 17: Giao diện Trang Giới Thiệu

The screenshot shows two main sections: 'Comfort Above All' and 'Culinary Excellence'. Each section includes a photograph and a descriptive paragraph.

- Comfort Above All:** Our aircraft are equipped with ergonomically designed seats, providing optimal comfort even on long-haul flights. With generous legroom and adjustable headrests, you can relax and arrive at your destination feeling refreshed.
Premium cabins feature fully-flat beds and private suites for an unparalleled travel experience.
- Culinary Excellence:** We believe that airline food should be something to look forward to. Our menus are crafted by award-winning chefs, featuring both international

Hình 18: Giao diện Trang Giới Thiệu



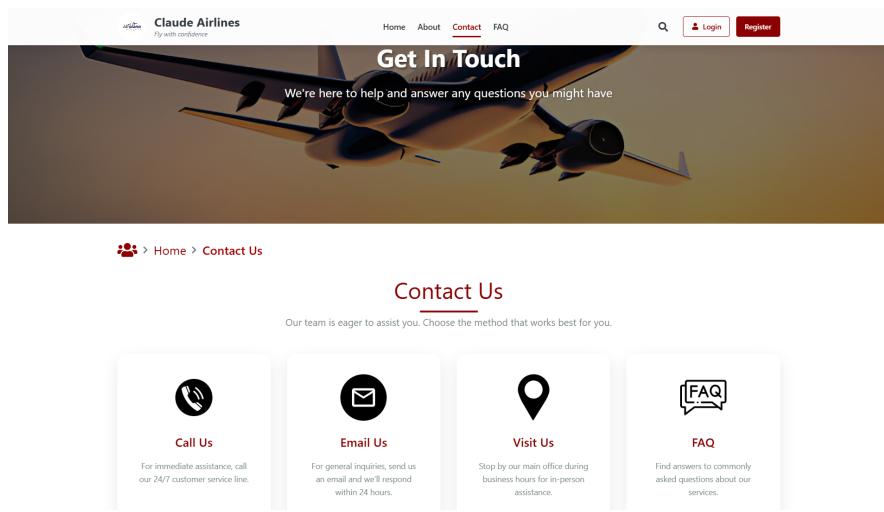
Hình 19: Giao diện Trang Giới Thiệu Về Các Loại Máy Bay

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 20: 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 Airline, 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.



The screenshot shows a contact form titled "Send Us a Message". At the top left, there's a logo for "Claude Airlines" with the tagline "Fly with confidence". The form has several input fields: "Full Name *", "Email *", "Phone Number", "Subject *", and "Message *". Below these, there's a "Business Hours" section with the following details:
Main Office: 123 Aviation Blvd, Los Angeles, CA 90045
Monday - Friday: 8am - 8pm
Saturday: 9am - 5pm
Sunday: Closed
At the bottom right of the form is a "Send Message" button.

Hình 21: Giao diện Form tin nhắn ở trang Liên hệ

- **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.



The screenshot shows the 'Our Locations' section of the Claude Airlines website. It features three separate boxes for different office locations:

- New York 2** (Main Office): Address: 123 Main Street, New York, NY 10001. Phone: +8410301023. Email: nguyendinhbang53az@gmail.com. Hours: Mon-Fri 9am-3pm. Map showing the location in New York City.
- Los Angeles Office** (Branch Office): Address: 456 South Figueroa Street, Los Angeles, CA 90017. Phone: +8410301023. Email: nguyendinhbang53az@gmail.com. Hours: Mon-Fri 9am-3pm. Map showing the location in downtown Los Angeles.
- Chicago Office** (Branch Office): Address: 789 North Michigan Avenue, Chicago, IL 60611. Phone: +8410301023. Email: nguyendinhbang53az@gmail.com. Hours: Mon-Fri 9am-3pm. Map showing the location in the Loop area of Chicago.

Hình 22: 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 Frequently Asked Questions (FAQ) page of the Claude Airlines website. The page includes a search bar, category filters, and a detailed answer to a specific question.

Frequently Asked Questions

Find answers to the most common questions about our services

Search for questions...

All Booking Baggage Reservation Membership Accessibility Health & Safety Travel Guidelines

How do I book a flight?

To book a flight, navigate to our homepage and use the booking form. Enter your departure and arrival destinations, select your travel dates, and click 'Search Flights'. You'll then be presented with available options where you can select your preferred flight and complete the booking process by entering passenger details and payment information.

Hình 23: 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", "Baggage", "Reservation") 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.



4.2.7 Trang Danh sách Bài đăng Vlog

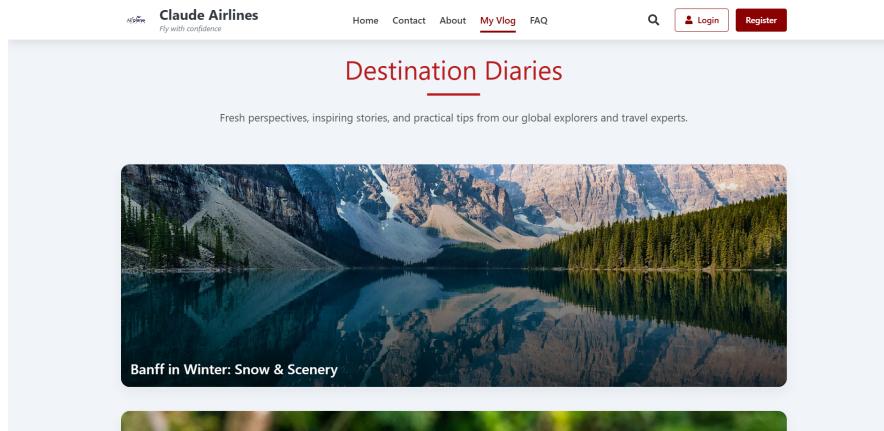
4.2.7.1 Mô tả chung Trang Danh sách Bài đăng Vlog ('VlogListPage.js') được thiết kế để người dùng khám phá các bài viết và câu chuyện du lịch. Trang này hiển thị một danh sách các bài đăng, cho phép tìm kiếm, và cung cấp các mục nội dung hấp dẫn khác để tăng tương tác.



Hình 24: Giao diện Trang Danh sách Bài viết

4.2.7.2 Các thành phần giao diện chính và chức năng

- Hero Section:** Một khu vực nổi bật ở đầu trang với video nền động từ YouTube, tiêu đề lớn và mô tả truyền cảm hứng. Tích hợp một ô tìm kiếm trực tiếp trong hero section, cho phép người dùng nhập từ khóa để tìm kiếm vlog.
- Header Placeholder:** Đảm bảo nội dung không bị che khuất bởi header cố định của trang web.
- Lưới Bài đăng (Vlog Grid):**
 - Hiển thị các bài đăng vlog dưới dạng lưới các thẻ (cards), mỗi thẻ bao gồm hình ảnh đại diện và tiêu đề. Khi rê chuột vào, có hiệu ứng phóng to nhẹ hình ảnh và thay đổi màu tiêu đề.
 - Nhấp vào một thẻ sẽ điều hướng người dùng đến trang chi tiết của bài đăng đó.
 - Sử dụng cơ chế "skeleton loading" (các thẻ giữ chỗ với hiệu ứng pulse) trong khi tải dữ liệu bài đăng để cải thiện trải nghiệm người dùng.
 - Hiển thị thông báo lỗi nếu không tải được bài đăng hoặc không có bài đăng nào phù hợp với từ khóa tìm kiếm.



Hình 25: Giao diện Các bài viết

- **Tìm kiếm Bài đăng:**

- Người dùng có thể nhập từ khóa vào ô tìm kiếm.
- Chức năng tìm kiếm được thực hiện bằng cách gọi API với tham số ‘search’.
- Sử dụng kỹ thuật ‘debounce’ để tối ưu hóa việc gửi yêu cầu tìm kiếm, tránh gọi API liên tục khi người dùng đang gõ.

- **Phân trang:**

- Nếu có nhiều bài đăng, chúng sẽ được chia thành nhiều trang.
- Hiển thị thông tin trang hiện tại và tổng số trang.
- Các nút "Previous" và "Next" cho phép điều hướng qua lại giữa các trang. Khi chuyển trang, màn hình sẽ cuộn lên đầu danh sách bài đăng.

- **Mục "Ever Wondered?":**

- Một section hiển thị các sự thật hoặc câu hỏi thú vị về du lịch.
- Mỗi mục bao gồm câu hỏi và câu trả lời, được trình bày dưới dạng accordion, mở rộng khi rê chuột vào câu hỏi, kèm hiệu ứng chuyển động.



The screenshot shows a travel-themed website with a navigation bar at the top. The main heading is "Ever Wondered?". Below it is a sub-headline: "Dive into these intriguing travel snippets that might just surprise you about the world we explore!". There are three expandable sections with questions: "Can a massive cave truly develop its own internal weather system?", "Is it feasible to have bespoke clothing tailored and ready within just one day?", and "What is the unusual process behind the making of 'weasel coffee'?". Each section has a small upward arrow icon to collapse it.

Hình 26: Giao diện Mục Thông tin du lịch thú vị

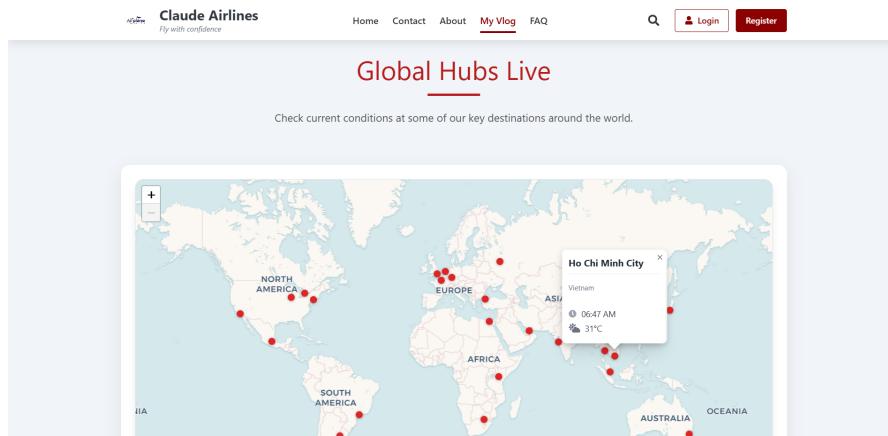
• **Mục "Thematic Adventures":**

- Sử dụng thư viện 'Swiper' để tạo một carousel trình chiếu các chủ đề vlog.
- Mỗi slide trong carousel là một chủ đề với hình ảnh, tên và mô tả ngắn, liên kết đến trang vlog đã được lọc theo chủ đề đó.

The screenshot shows a section titled "Thematic Adventures" with the sub-headline: "Embark on curated visual journeys through our diverse collection of vlog themes.". It displays five cards, each representing a theme with a thumbnail image and a brief description:

- Apartment Cafes**: Apartment coffee gems in old buildings, vibrant pedestrian streets.
- Highland Culture**: Rich traditions and vibrant attire of a mountain ethnic group.
- Floral Cycles**: Colorful bouquets on bicycles brightening city streets.
- Soaring Skyscraper**: Panoramic views from a modern architectural marvel.
- River Delta Life**: Vibrant waterways, floating markets, lush landscapes.

Hình 27: Giao diện Mục Chủ đề ẩn tượng qua ảnh



Hình 28: Giao diện Mục Bản đồ tương tác

- **Bản đồ Tương tác "Global Hubs Live":**

- Sử dụng thư viện ‘react-leaflet’ để nhúng và hiển thị một bản đồ thế giới tương tác trong trang.
- Hiển thị vị trí của nhiều thành phố/trung tâm hàng không lớn trên toàn cầu (khoảng 25 địa điểm) bằng các điểm đánh dấu tùy chỉnh (ví dụ: chấm đỏ nhấp nháy).
- Cho phép người dùng tương tác với bản đồ thông qua kéo rê (pan) và phóng to/thu nhỏ (zoom).
- Khi nhấp vào một điểm đánh dấu (marker), một cửa sổ thông tin (popup) với giao diện được tùy chỉnh sẽ xuất hiện.
- Nội dung popup bao gồm: tên thành phố, quốc gia, thời gian địa phương hiện tại (được tính toán động dựa trên múi giờ) và thông tin thời tiết tĩnh (biểu tượng và nhiệt độ).

- **Tối ưu SEO và Meta Tags:** Sử dụng ‘React Helmet’ để đặt tiêu đề trang, mô tả, từ khóa và thẻ canonical, giúp cải thiện khả năng tìm kiếm và hiển thị trên mạng xã hội.

- **Xử lý API:**

- Gọi API ‘vlogPost/listPublished’ để lấy danh sách các bài đăng đã xuất bản, hỗ trợ tham số ‘limit’, ‘page’, và ‘search’.
- Xử lý các trạng thái tải dữ liệu (loading), lỗi (error), và trường hợp không có dữ liệu. Phân tích phản hồi JSON và xử lý các lỗi HTTP cũng như lỗi logic từ server.

4.2.8 Trang Chi tiết Bài đăng Vlog (Vlog Post Page)

4.2.8.1 Mô tả chung Trang Chi tiết Bài đăng Vlog (‘VlogPostPage.js’) hiển thị đầy đủ nội dung của một bài đăng cụ thể, đồng thời cho phép người dùng tương tác thông qua việc bình luận, đánh giá và chia sẻ.



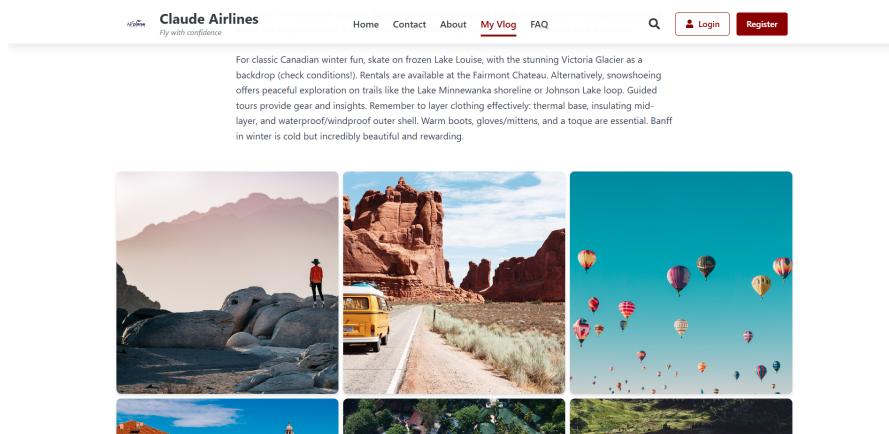
4.2.8.2 Các thành phần giao diện chính và chức năng

- Hiển thị Nội dung Bài đăng:



Hình 29: Giao diện Tiêu đề Bài viết

- **Hero Image/Header:** Nếu bài đăng có hình ảnh đại diện, nó sẽ được hiển thị lớn ở đầu trang, phủ toàn bộ chiều cao màn hình xem ban đầu, kèm theo tiêu đề và mô tả ngắn nổi bật. Nếu không có ảnh đại diện, tiêu đề và thông tin meta sẽ được hiển thị theo cách truyền thống hơn.
- **Thông tin Meta:** Hiển thị tên tác giả, ngày đăng, và điểm đánh giá trung bình cùng số lượt đánh giá nếu có.
- **Nội dung chính:** Hiển thị nội dung chi tiết của bài viết, hỗ trợ định dạng xuống dòng.
- **Thư viện Hình ảnh:** Nếu bài đăng có danh sách các hình ảnh trong ‘gallery_images’, chúng sẽ được hiển thị dưới dạng một lưới các hình ảnh nhỏ. Có thể hiển thị xen kẽ với nội dung chính nếu cấu trúc bài viết phù hợp (ví dụ 4 đoạn văn).

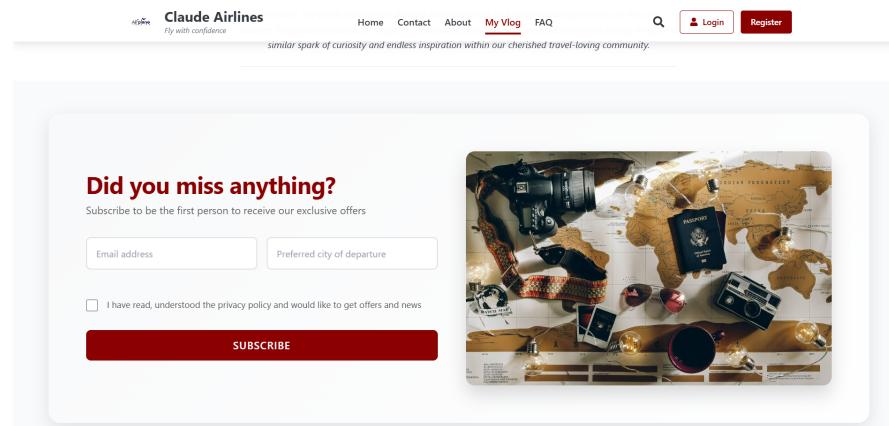


Hình 30: Giao diện Hình ảnh Bài viết

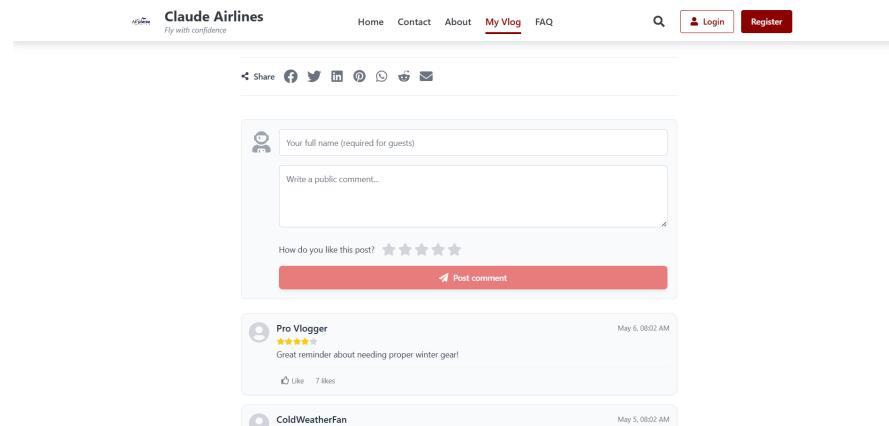


- Chức năng Bình luận và Đánh giá:

- Form Gửi Bình luận:



Hình 31: Giao diện Phần Đăng ký nhận Bản tin



Hình 32: Giao diện Phần Bình luận và Đánh giá bài viết

- * Cho phép cả người dùng đã đăng nhập (sử dụng ‘useAuth’ để lấy thông tin ‘loggedInUser’ và ‘authToken’) và khách gửi bình luận.
- * Khách cần nhập tên đầy đủ ('guestFullName') nếu không đăng nhập để bình luận được hiển thị tên.
- * Người dùng nhập nội dung bình luận và có thể chọn đánh giá bài đăng từ 1 đến 5 sao.
- * Form có xác thực đầu vào (bình luận không được trống, khách phải nhập tên).
- * Hiển thị thông báo lỗi hoặc thành công sau khi gửi. Bình luận thành công sẽ chờ duyệt.
- * Avatar được hiển thị dựa trên trạng thái đăng nhập và ảnh đại diện của người dùng.



– **Hiển thị Danh sách Bình luận:**

- * Tải và hiển thị các bình luận đã được duyệt cho bài đăng hiện tại từ API.
- * Sắp xếp bình luận theo thời gian mới nhất.
- * Mỗi bình luận hiển thị avatar (nếu có), tên người gửi, nội dung, điểm đánh giá (nếu có), và ngày giờ bình luận.
- * **Chức năng Thích (Like) Bình luận:** Người dùng (kể cả khách) có thể thích hoặc bỏ thích một bình luận. Số lượt thích được hiển thị và cập nhật. Trang thái "đã thích" của người dùng hiện tại được quản lý cục bộ. API được gọi để cập nhật lượt thích trên server. Có xử lý optimistic update và revert nếu API lỗi.
- * Hiển thị thông báo nếu không có bình luận nào hoặc đang tải.

• **Chức năng Chia sẻ:**

- Cung cấp các nút để chia sẻ bài đăng lên các mạng xã hội phổ biến như Facebook, Twitter, LinkedIn, Pinterest (nếu có ảnh đại diện), WhatsApp, Reddit và qua Email.
- Sử dụng URL hiện tại, tiêu đề bài đăng, mô tả ngắn và ảnh đại diện để tạo link chia sẻ.

• **Mục Đăng ký Nhận Bản tin:** Một component tĩnh hiển thị form đăng ký nhận email ưu đãi, tương tự như trên các trang khác.

• **Tối ưu SEO và Meta Tags:** Sử dụng 'React Helmet' để thiết lập tiêu đề trang động theo tên bài đăng, mô tả, từ khóa, thẻ canonical, và các thẻ Open Graph cũng như Twitter Cards để tối ưu hóa việc chia sẻ và hiển thị trên công cụ tìm kiếm.

• **Xử lý API và Trạng thái:**

- Tải chi tiết bài đăng từ API.
- Xử lý các trạng thái tải (loading), lỗi (error) và hiển thị thông báo tương ứng.
- Hình ảnh được hiển thị từ API_URL nếu đường dẫn là tương đối.



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.



6 Kết luận

Quá trình thực hiện Bài tập lớn Xây dựng Website Hàng hàng không Claude Airline đã trang bị cho nhóm những kinh nghiệm quý báu trong việc áp dụng kiến thức lý thuyết vào một dự án web thực tế. Từ khâu lên ý tưởng, thiết kế cơ sở dữ liệu, lựa chọn công nghệ, đến phát triển và triển khai, mỗi giai đoạn đều mang lại những bài học và thử thách, giúp các thành viên nâng cao kỹ năng chuyên môn và khả năng làm việc nhóm. Sản phẩm cuối cùng là một website tương đối hoàn chỉnh, thể hiện sự nỗ lực và quyết tâm của cả đội.

Website Claude Airline đã thành công trong việc hiện thực hóa các chức năng cốt lõi cho cả người dùng và quản trị viên. Giao diện người dùng, được xây dựng bằng ReactJS, tập trung vào trải nghiệm tương tác, trực quan với các trang thông tin đa dạng, tính năng xác thực, và đặc biệt là hệ thống bài đăng vlog phong phú cho phép xem, tìm kiếm, bình luận và đánh giá. Song song đó, trang quản trị sử dụng PHP thuần theo mô hình MVC và template Mazer, cung cấp một công cụ mạnh mẽ để quản lý nội dung động, thông tin hàng, người dùng và toàn bộ vòng đời của các bài đăng vlog cũng như bình luận liên quan.

Việc ứng dụng các công nghệ hiện đại như ReactJS, PHP, MySQL, kết hợp với các kỹ thuật tối ưu SEO và các biện pháp bảo mật cơ bản, không chỉ đảm bảo website hoạt động ổn định mà còn mang lại hiệu quả tiếp cận người dùng tốt hơn. Quá trình làm việc với API RESTful, quản lý trạng thái ứng dụng phía client, và xử lý logic phía server đã giúp nhóm hiểu sâu hơn về kiến trúc của một ứng dụng web hiện đại và các thách thức trong quá trình tích hợp hệ thống.

Dù đã đạt được những mục tiêu đề ra, dự án vẫn còn không gian để phát triển thêm các tính năng nâng cao như hệ thống đặt vé trực tuyến, quản lý chuyến bay chi tiết hay tích hợp cổng thanh toán, vốn đòi hỏi nhiều thời gian và nguồn lực hơn. Tuy nhiên, nền tảng hiện tại là một bước đệm vững chắc. Những kiến thức và kinh nghiệm thu nhận được từ dự án này sẽ là hành trang giá trị, hỗ trợ các thành viên trong những dự án phức tạp và thách thức hơn trong sự nghiệp phát triển phần mềm sắp tới.



References

- [1] Claude Airline's Source Code <https://github.com/bawfng04/Claude-Airline>
- [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/>