

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  
KHOA CÔNG NGHỆ THÔNG TIN 1

o0o



BÁO CÁO THỰC TẬP CƠ SỞ

Tên đề tài: Ứng dụng đặt phòng Homestay

Nguyễn Ngọc Quang MSSV: D21DCCN630

Giảng viên hướng dẫn: Ths. Nguyễn Hoàng Anh

HÀ NỘI, 05/2024

## LỜI CẢM ƠN

Trong thời gian hoàn thành chuyên đề: “Ứng dụng đặt phòng Homestay” em xin chân thành cảm ơn sự giúp đỡ nhiệt tình của thầy Nguyễn Hoàng Anh đã giúp đỡ, cung cấp nhiều thông tin quý báu và tạo điều kiện cho em trong quá trình thực hiện chuyên đề này. Cảm ơn thầy đã tận tình hướng dẫn, giúp đỡ chỉ bảo cho em thực hiện và hoàn thành đề tài.

Mặc dù bản thân đã rất cố gắng nhưng do thời gian, kiến thức và kinh nghiệm có hạn nên bài làm của em còn có nhiều thiếu sót trong việc trình bày, đánh giá và đề xuất ý kiến. Em rất mong nhận được sự thông cảm và đóng góp ý kiến của quý thầy cô và các bạn.

## MỤC LỤC

<b>CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI.....</b>	<b>1</b>
1.1 Đặt vấn đề.....	1
1.2 Mục tiêu và phạm vi đề tài.....	1
1.2.1 Mục Tiêu .....	1
1.2.2 Phạm Vi.....	2
1.3 Định hướng giải pháp.....	2
1.4 Bố cục bài tập lớn .....	3
<b>CHƯƠNG 2. CÔNG NGHỆ SỬ DỤNG.....</b>	<b>4</b>
2.1 Ngôn ngữ lập trình TypeScript .....	4
2.2 Framework.....	4
2.2.1 Backend: NestJS .....	4
2.2.2 Frontend: ReactJS .....	4
2.2.3 Database: PostgreSQL trên Docker .....	5
<b>CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ .....</b>	<b>7</b>
3.1 Mô tả kiến trúc .....	7
3.1.1 Mô hình Client Side Rendering .....	7
3.1.2 Biểu đồ phân rã chức năng .....	8
3.1.3 Biểu đồ flowchart mô tả flow hệ thống.....	9
3.1.4 Cơ sở dữ liệu .....	10
<b>CHƯƠNG 4. THIẾT KẾ GIAO DIỆN .....</b>	<b>13</b>
4.1 Thiết kế giao diện .....	13
4.1.1 Trang Welcome.....	13
4.1.2 Trang Home .....	13
4.1.3 Trang danh sách Homestay.....	14

4.1.4 Trang tìm kiếm .....	14
4.1.5 Trang đăng ký.....	15
4.1.6 Trang đăng nhập .....	15
4.1.7 Trang Profile .....	17
4.1.8 Trang dành riêng cho Host .....	18
4.1.9 Trang dành riêng cho Admin .....	18
4.1.10 Trang chi tiết Homestay .....	18
4.1.11 Trang đặt phòng .....	20
4.1.12 Trang thanh toán .....	20

## **CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN ..... 22**

5.1 Kết luận .....	22
5.2 Hướng phát triển.....	22

## DANH MỤC HÌNH VẼ

Hình 2.1	Mô tả các API phía Backend . . . . .	5
Hình 2.2	Mô tả API của Homestay Service . . . . .	5
Hình 2.3	Trang web sử dụng React . . . . .	6
Hình 2.4	Database Postgresql chạy trên Docker . . . . .	6
Hình 2.5	Postgresql sử dụng trên DBMS DBeaver . . . . .	6
Hình 3.1	Sơ đồ kiến trúc CSR . . . . .	8
Hình 3.2	Biểu đồ phân rã chức năng . . . . .	8
Hình 3.3	Biểu đồ flowchart . . . . .	10
Hình 3.4	Thực thể Otp . . . . .	11
Hình 3.5	Mối quan hệ giữa thực thể Homestay và Users . . . . .	11
Hình 4.1	Màn hình Home . . . . .	13
Hình 4.2	Màn hình Home . . . . .	13
Hình 4.3	Màn hình danh sách Homestay . . . . .	14
Hình 4.4	Màn hình danh sách Homestay . . . . .	14
Hình 4.5	Màn hình đăng ký tài khoản . . . . .	15
Hình 4.6	Màn hình đăng nhập . . . . .	16
Hình 4.7	Màn hình OTP . . . . .	16
Hình 4.8	Màn hình pop up sai OTP . . . . .	16
Hình 4.9	Trang Profile người dùng . . . . .	17
Hình 4.10	Trang danh sách sở hữu homestay của Host . . . . .	18
Hình 4.11	Trang danh sách homestay được đặt của Host . . . . .	18
Hình 4.12	Trang tạo Homestay của Host . . . . .	19
Hình 4.13	Trang Dashboard Admin . . . . .	19
Hình 4.14	Trang danh sách homestay được theo dõi bởi Admin . . . . .	19
Hình 4.15	Trang theo dõi người dùng bởi Admin . . . . .	19
Hình 4.16	Trang chi tiết homestay . . . . .	20
Hình 4.17	Trang đặt phòng . . . . .	20
Hình 4.18	Trang thanh toán . . . . .	21

## DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT

Thuật ngữ	Ý nghĩa
API	Giao diện lập trình ứng dụng (Application Programming Interface)
CSS	Định dạng lại các phần tử được tạo ra bởi các ngôn ngữ đánh dấu (Cascading Style Sheets)
HTML	Ngôn ngữ đánh dấu siêu văn bản (HyperText Markup Language)

# CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

## 1.1 Đặt vấn đề

Trong bối cảnh phát triển không ngừng của ngành du lịch, nhu cầu về dịch vụ lưu trú đa dạng và tiện lợi đang ngày càng tăng cao. Homestay, một loại hình dịch vụ lưu trú đặc biệt, mang đến cho du khách những trải nghiệm chân thực về văn hóa, đời sống địa phương. Không chỉ là nơi nghỉ ngơi, homestay còn là cầu nối giúp du khách và chủ nhà giao lưu, học hỏi lẫn nhau.

Tuy nhiên, việc tìm kiếm và đặt phòng homestay hiện tại vẫn còn gặp nhiều khó khăn. Các nền tảng đặt phòng truyền thống như khách sạn, resort thường không đáp ứng được nhu cầu đặc thù của loại hình này. Du khách thường phải dựa vào các trang web không chuyên, mất nhiều thời gian để tìm kiếm, xác nhận thông tin và đặt phòng. Đối với các chủ homestay, việc quản lý và tiếp cận khách hàng tiềm năng cũng gặp nhiều hạn chế do thiếu kênh quảng bá hiệu quả.

## 1.2 Mục tiêu và phạm vi đề tài

Nhằm giải quyết những bất cập trên, ứng dụng đặt phòng homestay ra đời như một giải pháp hữu ích, kết nối trực tiếp giữa du khách và chủ nhà, tạo điều kiện thuận lợi cho việc tìm kiếm, đặt phòng và quản lý dịch vụ lưu trú. Ứng dụng này không chỉ mang lại lợi ích cho du khách trong việc dễ dàng tiếp cận thông tin và đặt phòng, mà còn hỗ trợ các chủ homestay trong việc tối ưu hóa quản lý và thu hút khách hàng.

### 1.2.1 Mục Tiêu

#### 1. Tạo ra nền tảng đặt phòng homestay tiện lợi và hiệu quả:

- Xây dựng một ứng dụng di động và website giúp du khách dễ dàng tìm kiếm, so sánh và đặt phòng homestay phù hợp với nhu cầu cá nhân.

#### 2. Nâng cao trải nghiệm người dùng:

- Cung cấp giao diện thân thiện, dễ sử dụng và các tính năng hỗ trợ như đánh giá, bình luận, và đề xuất homestay dựa trên sở thích và lịch sử tìm kiếm.

#### 3. Hỗ trợ quản lý cho chủ homestay:

- Phát triển các công cụ quản lý đặt phòng, lịch trống, và tương tác với khách hàng, giúp chủ homestay tối ưu hóa việc quản lý và nâng cao chất lượng dịch vụ.

### 1.2.2 Phạm Vi

#### 1. Đối tượng sử dụng:

- Ứng dụng hướng tới ba nhóm đối tượng chính: du khách (người có nhu cầu đặt phòng homestay), chủ homestay (người cung cấp dịch vụ lưu trú) và người quản trị ứng dụng (người theo dõi quá trình kinh doanh)

#### 2. Chức năng của ứng dụng:

- **Cho du khách:** Tìm kiếm homestay theo địa điểm, giá cả, tiện nghi; xem đánh giá và bình luận từ khách hàng trước; đặt phòng và thanh toán trực tuyến; quản lý đặt phòng và lịch trình.
- **Cho chủ homestay:** Đăng ký và quản lý thông tin homestay; cập nhật lịch trống; nhận và quản lý đặt phòng; tương tác với khách hàng qua hệ thống tin nhắn.
- **Cho người quản trị ứng dụng:** Theo dõi homestay và du khách trên hệ thống.

#### 3. Phạm vi kỹ thuật:

- Ứng dụng sẽ được phát triển trên nền tảng website, sử dụng công nghệ hiện đại đảm bảo hiệu suất và bảo mật thông tin người dùng.

#### 4. Phạm vi địa lý:

- Ban đầu tập trung vào thị trường trong nước, với khả năng mở rộng ra các thị trường quốc tế trong tương lai.

## 1.3 Định hướng giải pháp

#### 1. Định hướng công nghệ sử dụng:

Giải quyết vấn đề bằng cách sử dụng công nghệ web hiện đại như HTML5, CSS, TypeScript cho phía front-end và một framework web như React.js cho phía client-side. Đối với phía back-end sử dụng Node.js với framework là NestJs để xây dựng API và quản lý dữ liệu.

#### 2. Mô tả ngắn gọn giải pháp:

Ứng dụng web sẽ được phát triển với giao diện thân thiện và đáp ứng, cho phép người dùng dễ dàng tìm kiếm, so sánh và đặt phòng homestay. Sử dụng các công nghệ hiện đại và thiết kế linh hoạt, ứng dụng sẽ cung cấp trải nghiệm người dùng tốt nhất. Dữ liệu của người dùng và homestay sẽ được quản lý và lưu trữ một cách an toàn trên máy chủ.

#### 3. Đóng góp chính và kết quả đạt được:

Đóng góp chính của bài tập lớn này là xây dựng một ứng dụng web đặt phòng homestay thân thiện với người dùng, tích hợp các công nghệ hiện đại để nâng cao trải nghiệm người dùng và hiệu quả quản lý cho chủ homestay. Kết quả đạt được bao gồm một ứng dụng hoạt động hiệu quả, dễ sử dụng và cung cấp giá trị cho cả người dùng và chủ homestay.

### 1.4 **Bố cục bài tập lớn**

Chương 2 giới thiệu công nghệ được sử dụng để tạo nên ứng dụng này bao gồm ngôn ngữ lập trình, Framework, Database, Backend...

Ở chương 3 này sẽ mô tả tổng quan sơ lược về thiết kế hệ thống ứng dụng, flow và thiết kế database trong ứng dụng.

Chương 4 sẽ giới thiệu về giao diện của ứng dụng, mô tả thành phần bên trong và chức năng của chúng.

Chương 5 sẽ tổng kết lại quá trình làm đề tài và kinh nghiệm rút ra.

## CHƯƠNG 2. CÔNG NGHỆ SỬ DỤNG

### 2.1 Ngôn ngữ lập trình TypeScript

TypeScript là một ngôn ngữ lập trình mã nguồn mở phát triển bởi Microsoft. Nó là một phần mở rộng của JavaScript, được thiết kế để viết mã lớn và phức tạp hơn một cách dễ dàng hơn.

Với TypeScript, bạn có thể sử dụng các tính năng như kiểu dữ liệu tĩnh, lập trình hướng đối tượng, và các tính năng của ES6 (ECMAScript 6) mà không cần lo lắng về sự không tương thích giữa các trình duyệt. TypeScript cho phép bạn viết mã rõ ràng, dễ hiểu hơn và dễ bảo trì hơn.

Một số ưu điểm của TypeScript bao gồm:

- Kiểu dữ liệu tĩnh giúp phát hiện lỗi sớm hơn trong quá trình phát triển.
- Mã TypeScript có thể chuyển đổi thành mã JavaScript tương đối dễ dàng, giúp tích hợp với các dự án sử dụng JavaScript.
- Cộng đồng lớn và sự hỗ trợ mạnh mẽ từ Microsoft và các công ty khác.

### 2.2 Framework

Trong dự án này, sử dụng hai framework chính cho phần backend và frontend như sau:

#### 2.2.1 Backend: NestJS

NestJS là một framework Node.js mã nguồn mở được xây dựng trên cơ sở của TypeScript. Nó cung cấp một cách tiếp cận hiện đại cho việc xây dựng các ứng dụng server-side, với sự linh hoạt và khả năng mở rộng cao. NestJS sử dụng cấu trúc gọn gàng, hỗ trợ Dependency Injection, và tích hợp tốt với các thư viện như Express và Fastify. Với NestJS, có thể viết mã server-side một cách dễ dàng và bảo trì.

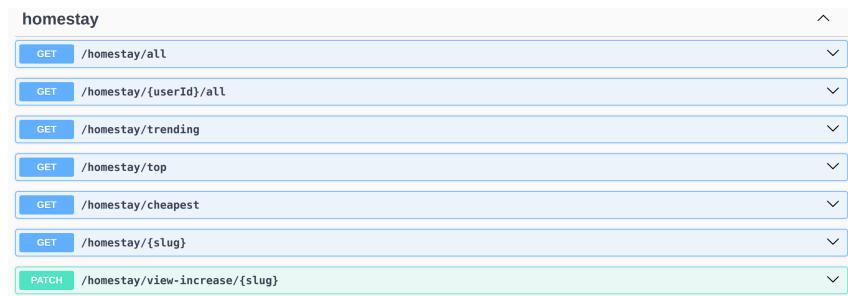
#### 2.2.2 Frontend: ReactJS

ReactJS là một thư viện JavaScript mã nguồn mở được phát triển bởi Facebook. Nó là một trong những thư viện phổ biến nhất cho việc xây dựng giao diện người dùng (UI). ReactJS sử dụng các thành phần (components) để tái sử dụng mã và tạo ra giao diện người dùng tương tác. Với cộng đồng lớn và sự hỗ trợ mạnh mẽ từ cộng đồng phát triển, ReactJS giúp xây dựng giao diện người dùng linh hoạt, dễ bảo trì và mở rộng.

## CHƯƠNG 2. CÔNG NGHỆ SỬ DỤNG



**Hình 2.1:** Mô tả các API phía Backend



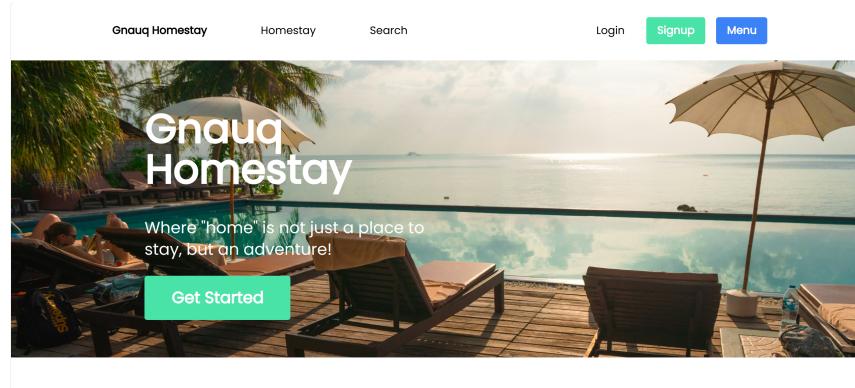
**Hình 2.2:** Mô tả API của Homestay Service

### 2.2.3 Database: PostgreSQL trên Docker

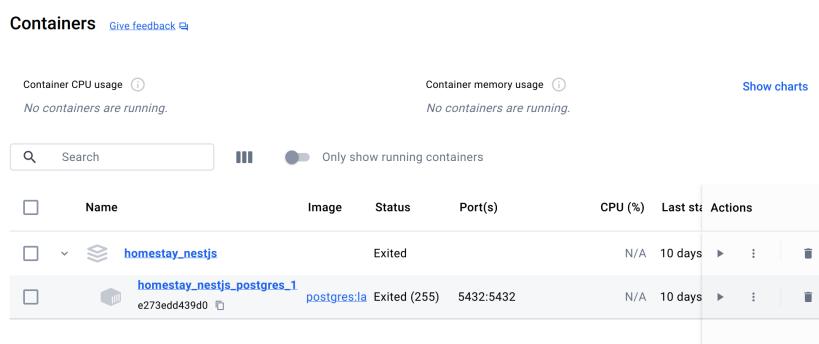
Trong dự án này sử dụng PostgreSQL làm hệ quản trị cơ sở dữ liệu (DBMS) chính và triển khai nó trên Docker để tăng tính di động và linh hoạt của ứng dụng.

PostgreSQL là một hệ quản trị cơ sở dữ liệu quan hệ mạnh mẽ và linh hoạt, được sử dụng rộng rãi trong các dự án phần mềm do tính ổn định, khả năng mở rộng và tính linh hoạt cao.

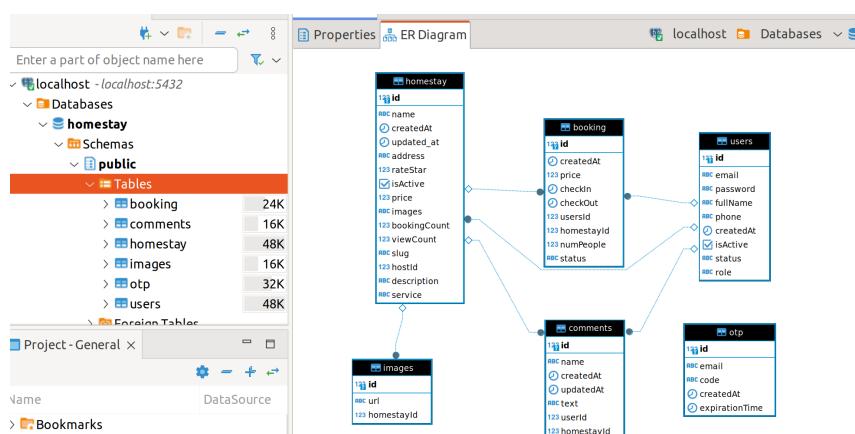
Docker là một nền tảng ảo hóa cấp container, cho phép triển khai ứng dụng và dịch vụ trong các container độc lập với môi trường máy chủ. Việc sử dụng PostgreSQL trên Docker giúp đơn giản hóa quá trình triển khai và quản lý cơ sở dữ liệu, đồng thời tạo điều kiện cho việc chạy ứng dụng một cách nhất quán trên các môi trường khác nhau từ máy phát triển đến máy chủ sản xuất.



**Hình 2.3:** Trang web sử dụng React



**Hình 2.4:** Database Postgresql chạy trên Docker



**Hình 2.5:** Postgresql sử dụng trên DBMS DBeaver

## CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ

### 3.1 Mô tả kiến trúc

#### 3.1.1 Mô hình Client Side Rendering

Mô hình Client Side Rendering (CSR) tập trung vào việc xử lý và render giao diện người dùng ở phía client. Dưới đây là các thành phần chính trong mô hình CSR:

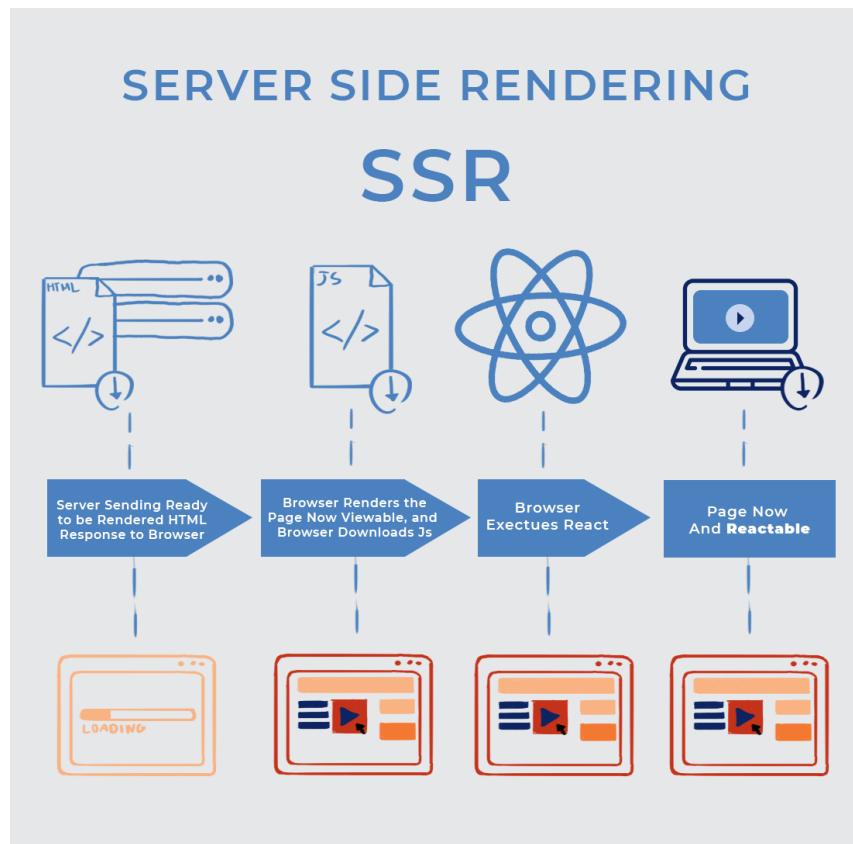
- **Client (Trình duyệt web):** Trình duyệt là nơi diễn ra toàn bộ quá trình render. Nó sẽ tải và thực thi các tập tin HTML, CSS và JavaScript được gửi từ server.
- **Server:** Server chủ yếu cung cấp các tài nguyên tĩnh như HTML, CSS, và JavaScript. Server không tham gia trực tiếp vào quá trình render trang mà chỉ phản hồi các yêu cầu dữ liệu từ client.
- **JavaScript:** JavaScript đóng vai trò quan trọng trong mô hình CSR. Nó xử lý việc tạo và cập nhật giao diện người dùng dựa trên dữ liệu nhận được từ server thông qua các yêu cầu AJAX hoặc Fetch API.
- **API:** API server cung cấp các điểm cuối (endpoints) để client có thể yêu cầu dữ liệu động mà không cần tải lại toàn bộ trang. Điều này giúp cải thiện trải nghiệm người dùng bằng cách làm cho ứng dụng phản hồi nhanh hơn.

Dưới đây là quy trình hoạt động của mô hình CSR:

1. Trình duyệt gửi yêu cầu đến server để tải trang.
2. Server phản hồi bằng cách gửi các tệp HTML, CSS, và JavaScript.
3. Trình duyệt tải và thực thi các tệp JavaScript để render giao diện người dùng.
4. Khi người dùng tương tác với ứng dụng (ví dụ: nhấn nút hoặc điền form), JavaScript sẽ gửi các yêu cầu dữ liệu đến server thông qua AJAX hoặc Fetch API.
5. Server xử lý các yêu cầu dữ liệu và gửi phản hồi lại cho client.
6. JavaScript nhận dữ liệu từ server và cập nhật giao diện người dùng mà không cần tải lại trang.

Ưu điểm của CSR bao gồm:

- **Tương tác người dùng mượt mà:** Do việc xử lý diễn ra ở client, các thay đổi giao diện người dùng diễn ra nhanh chóng và mượt mà.
- **Giảm tải cho server:** Server chỉ cần cung cấp tài nguyên tĩnh mà không cần



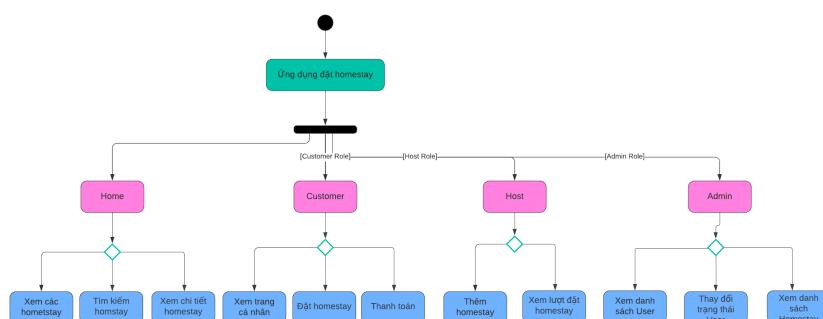
Hình 3.1: Sơ đồ kiến trúc CSR

xử lý rendering từng trang.

Nhược điểm của CSR bao gồm:

- **Thời gian tải đầu tiên lâu hơn:** Người dùng phải tải toàn bộ tài nguyên và thực thi JavaScript trước khi thấy nội dung trang.
- **SEO kém:** Các công cụ tìm kiếm có thể gặp khó khăn khi lập chỉ mục các trang web render bằng CSR do nội dung không có sẵn trong mã HTML ban đầu.

### 3.1.2 Biểu đồ phân rã chức năng



Hình 3.2: Biểu đồ phân rã chức năng

### a, Home

Home là màn hình chính mà mọi người dùng đều có thể truy cập. Từ đây, người dùng có thể:

- Xem các Homestay hiện có.
- Tìm kiếm Homestay dựa trên các tiêu chí cụ thể.
- Xem chi tiết từng Homestay.

### b, Customer Role

Sau khi đăng nhập, khách hàng có thể truy cập các chức năng sau:

- Xem trang cá nhân để quản lý thông tin cá nhân và lịch sử đặt phòng.
- Đặt Homestay, nơi họ chọn Homestay và đặt phòng.
- Thực hiện thanh toán cho các Homestay đã đặt.

### c, Host Role

Chủ nhà có thể thực hiện các chức năng quản lý liên quan đến Homestay của họ, bao gồm:

- Thêm Homestay mới vào hệ thống.
- Xem lượt đặt Homestay để theo dõi tình trạng đặt phòng của các Homestay mà họ quản lý.

### d, Admin Role

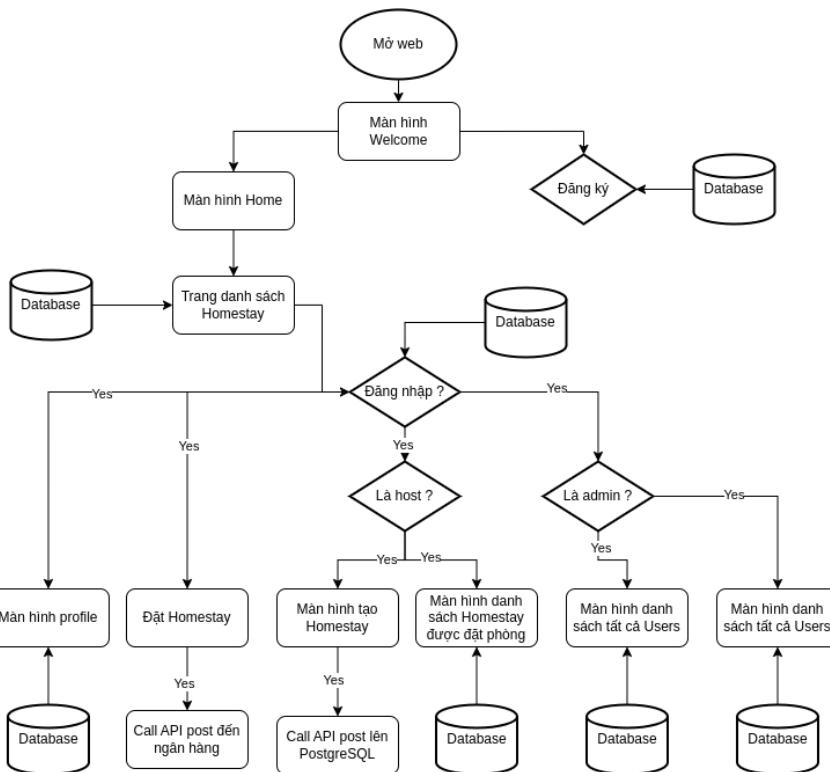
Quản trị viên có quyền quản lý toàn bộ hệ thống với các chức năng sau:

- Xem danh sách tất cả người dùng.
- Thay đổi trạng thái người dùng.
- Xem danh sách Homestay.

#### 3.1.3 Biểu đồ flowchart mô tả flow hệ thống

Hệ thống quản lý Homestay cho phép người dùng truy cập trang web và bắt đầu từ màn hình Welcome. Tại đây, người dùng có thể đăng ký hoặc đăng nhập. Nếu đăng ký, thông tin sẽ được lưu trữ trong cơ sở dữ liệu. Sau khi đăng nhập, hệ thống kiểm tra vai trò của người dùng: khách hàng, host hay admin.

Khách hàng có thể duyệt qua danh sách Homestay trên trang Home, xem chi tiết từng Homestay bao gồm mô tả, giá cả, và đánh giá từ các khách hàng trước. Họ có thể đặt phòng trực tuyến, với thông tin giao dịch được gửi tới ngân hàng qua API để thực hiện thanh toán. Khách hàng cũng có thể truy cập vào màn hình hồ sơ cá nhân để quản lý thông tin cá nhân, xem lịch sử đặt phòng và theo dõi trạng thái các



**Hình 3.3:** Biểu đồ flowchart

đặt phòng hiện tại.

Host có thể tạo mới và quản lý danh sách Homestay của họ. Các thông tin về Homestay mới sẽ được lưu vào cơ sở dữ liệu và gửi tới PostgreSQL qua API. Host có quyền theo dõi các Homestay của mình và xem danh sách các phòng đã được đặt.

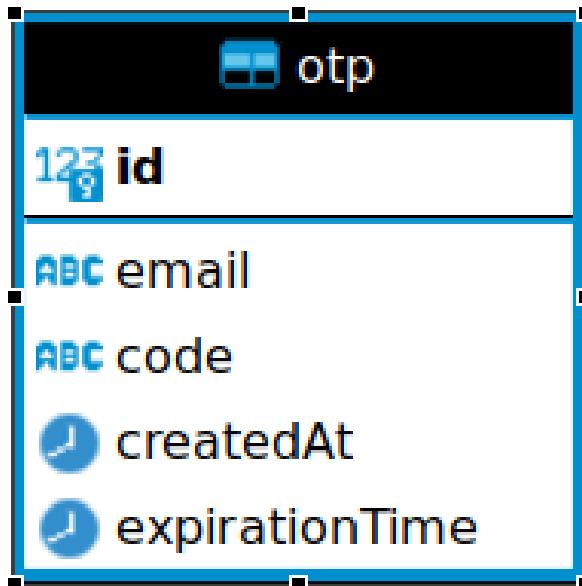
Admin có quyền truy cập toàn diện, cho phép họ xem và quản lý danh sách tất cả Homestay đã đặt và danh sách tất cả người dùng. Điều này giúp admin điều hành và quản lý hệ thống một cách hiệu quả. Mỗi bước trong quy trình đều được hỗ trợ bởi cơ sở dữ liệu, đảm bảo tính toàn vẹn và bảo mật thông tin.

Hệ thống này được thiết kế để cung cấp trải nghiệm người dùng mượt mà và thuận tiện, từ lúc đăng ký, đăng nhập, duyệt và đặt Homestay, đến quản lý thông tin cá nhân và giao dịch. Với sự phân chia rõ ràng vai trò giữa khách hàng, host và admin, hệ thống đảm bảo mỗi người dùng có các quyền hạn và chức năng phù hợp với vai trò của mình.

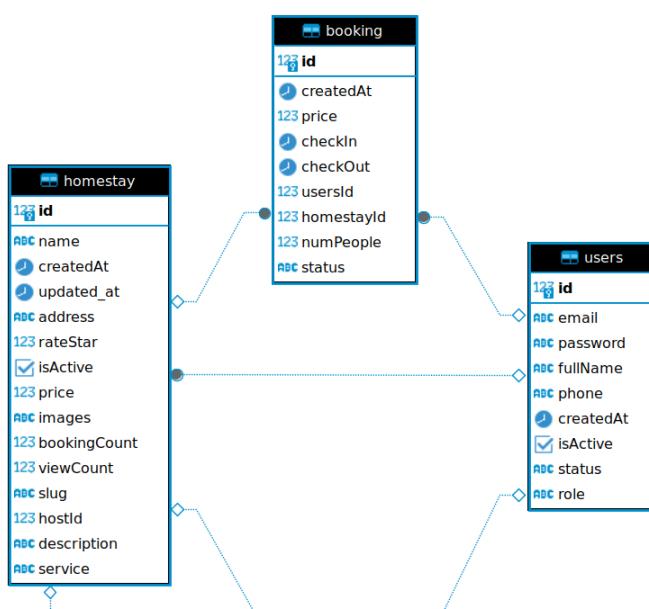
#### 3.1.4 Cơ sở dữ liệu

Thực thể Otp để lưu lại mã Otp do phía Backend tạo ra và để xác thực người dùng.

Thực thể Users có mối quan hệ nhiều với thực thể Homestay. Một user có



Hình 3.4: Thực thể Otp



Hình 3.5: Mối quan hệ giữa thực thể Homestay và Users

thể đặt nhiều homestay, một homestay có thể được đặt bởi nhiều người. Thông tin đặt phòng sẽ được lưu tại Booking với khóa ngoại (FK) là userId và homestayId.

Thực thể Users cũng có mối quan hệ một nhiều với thực thể Homestay. Một user có thể là chủ (host) của nhiều homestay. Bảng Homestay có khóa ngoại (FK) là hostId.

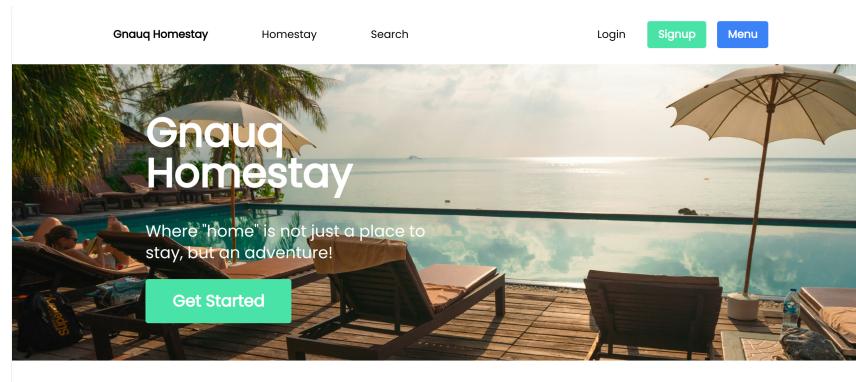
## CHƯƠNG 4. THIẾT KẾ GIAO DIỆN

### 4.1 Thiết kế giao diện

#### 4.1.1 Trang Welcome

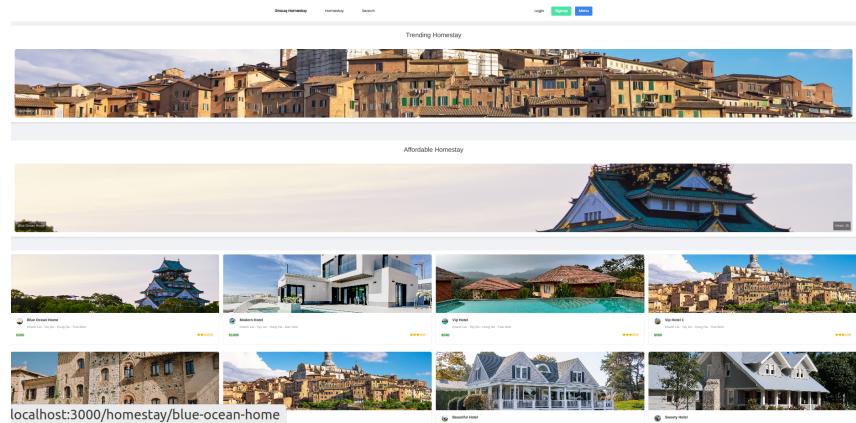
Trang welcome của web được thiết kế đơn giản và trực quan. Mục đích là chào mừng khách hàng tới trang web.

- Button để có thể truy cập vào trang Home
- Tên trang web
- Slogan trang web



Hình 4.1: Màn hình Home

#### 4.1.2 Trang Home



Hình 4.2: Màn hình Home

Trang Home với giao diện hiển thị ra các homestay đặc biệt.

- Thanh navbar có các chức năng như xem danh sách homestay, tìm kiếm homestay, đăng ký, đăng nhập
- Carousel thứ nhất hiển thị trending homestay là các homestay có lượt xem của

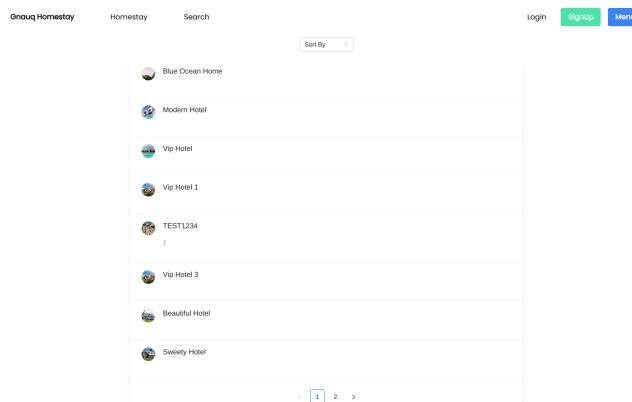
người dùng nhiều nhất

- Carousel thứ hai hiển thị affordable homestay là các homestay có giá thành rẻ nhất
- Dưới cùng là danh sách các homestay được phân trang

#### 4.1.3 Trang danh sách Homestay

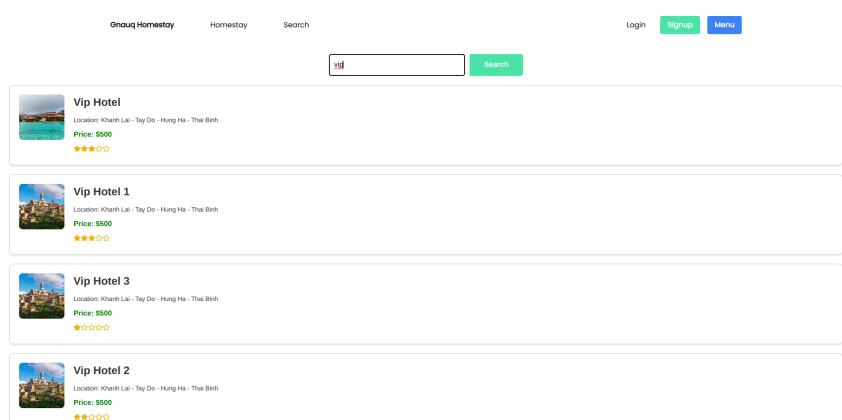
Trang danh sách Homestay với giao diện đơn giản, dễ nhìn.

- Các homestay được liệt kê dạng danh sách
- Nút chuyển trang để xem tiếp danh sách homestay ở trang sau



**Hình 4.3:** Màn hình danh sách Homestay

#### 4.1.4 Trang tìm kiếm



**Hình 4.4:** Màn hình danh sách Homestay

Trang tìm kiếm để tìm kiếm homestay theo tên hoặc địa chỉ của nó.

- Các homestay khi search sẽ hiện ra theo danh sách
- Tìm kiếm homestay theo địa chỉ hoặc theo tên homestay sẽ cho ra kết quả chính xác

### 4.1.5 Trang đăng ký

Trang đăng ký sẽ gồm các thông tin cơ bản do user nhập vào như sau:

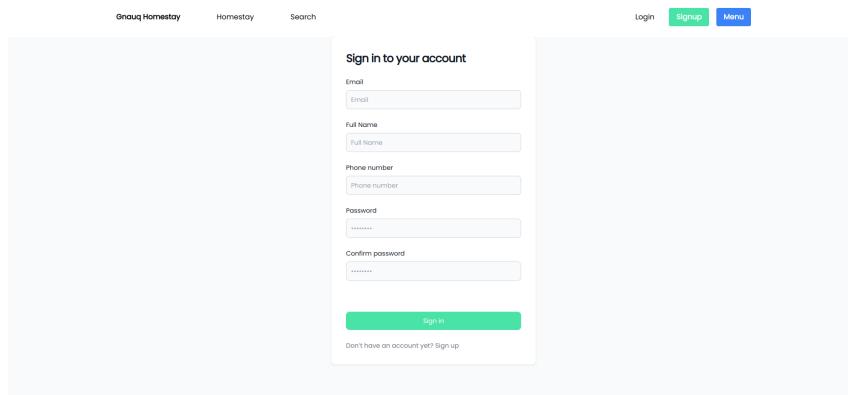
- Input nhập email
- Input nhập họ và tên
- Input nhập số điện thoại
- Input nhập mật khẩu
- Input nhập xác nhận mật khẩu

Nếu người dùng nhập sai ô xác nhận mật khẩu thì màn hình hiển thị thông báo pop up lỗi.

Nếu người dùng nhập email đã tồn tại trên hệ thống thì màn hình pop up là email đã tồn tại.

Các thông tin ở ô input đều được validate ở phía Frontend và Backend, đảm bảo cho việc thông tin rác không bị lưu vào Database.

Người dùng điền đúng thì sẽ hiện pop up thông báo thành công.



**Hình 4.5:** Màn hình đăng ký tài khoản

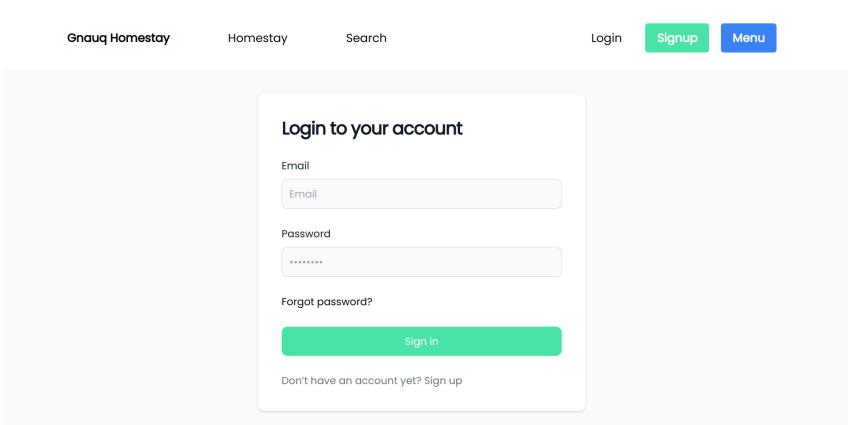
### 4.1.6 Trang đăng nhập

Trang đăng nhập chỉ gồm hai ô input là email và mật khẩu. Nếu nhập sai mật khẩu thì sẽ pop up báo sai mật khẩu.

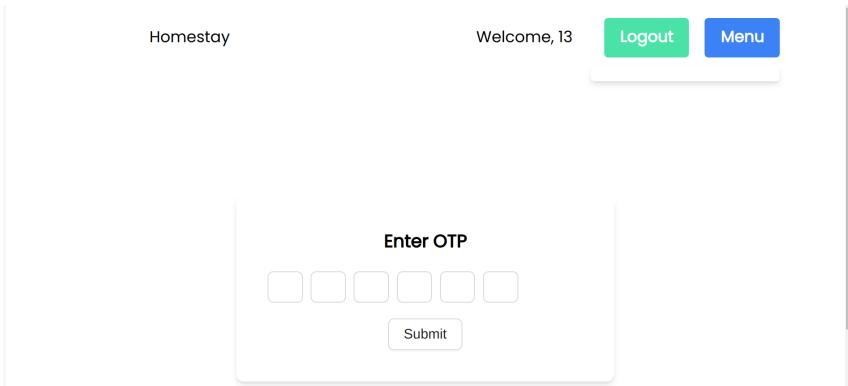
Trong lần đầu tiên đăng nhập, người dùng được đi đến trang OTP để xác thực tài khoản.

Người dùng nhận được mã OTP tại email của chính người dùng đăng ký

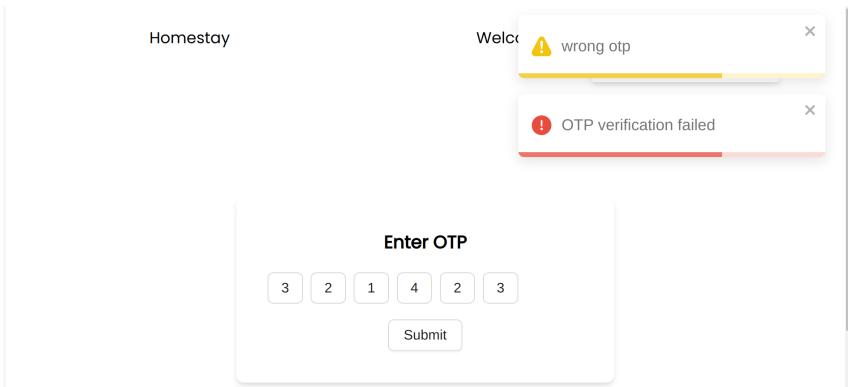
Trong trường hợp người dùng nhập sai mã OTP thì sẽ pop up báo sai mã OTP.



**Hình 4.6:** Màn hình đăng nhập



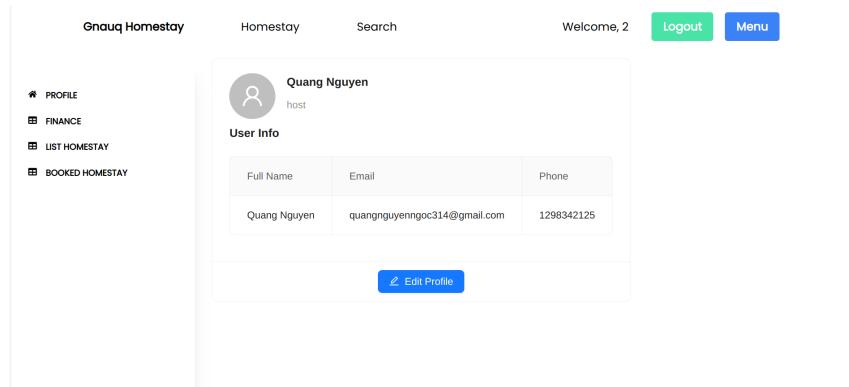
**Hình 4.7:** Màn hình OTP



**Hình 4.8:** Màn hình pop up sai OTP

### 4.1.7 Trang Profile

Trang hiện ra thông tin cá nhân của người dùng. Khi ấn nút Edit, thì người dùng có thể sửa đổi thông tin của chính họ ngoại trừ email.



**Hình 4.9:** Trang Profile người dùng

#### 4.1.8 Trang dành riêng cho Host

Homestay Name	Action
Blue Ocean Home	<a href="#">edit</a> <a href="#">delete</a>
Modern Hotel	<a href="#">edit</a> <a href="#">delete</a>
Vip Hotel	<a href="#">edit</a> <a href="#">delete</a>
Vip Hotel 1	<a href="#">edit</a> <a href="#">delete</a>
Vip Hotel 3	<a href="#">edit</a> <a href="#">delete</a>

Hình 4.10: Trang danh sách sở hữu homestay của Host

Homestay Name	Owner Name	Action
Modern Hotel	Quang Nguyen	<a href="#">detail</a> <a href="#">cancel</a>
Vip Hotel 1	Admin	<a href="#">detail</a> <a href="#">cancel</a>
Vip Hotel	Admin	<a href="#">detail</a> <a href="#">cancel</a>
Vip Hotel 1	Ha Thu 2k3	<a href="#">detail</a> <a href="#">cancel</a>

Hình 4.11: Trang danh sách homestay được đặt của Host

Chủ sở hữu (Host) có thể thực hiện các chức năng cơ bản như thêm mới, chỉnh sửa và xóa các homestay của mình một cách dễ dàng và tiện lợi. Khi muốn thêm một homestay mới, chủ sở hữu có thể điền đầy đủ thông tin chi tiết như tên homestay, địa chỉ, mô tả, giá cả, và các tiện nghi đi kèm. Chức năng chỉnh sửa cho phép chủ sở hữu cập nhật thông tin của homestay khi có sự thay đổi, như cập nhật giá phòng, thay đổi hình ảnh, hay bổ sung các dịch vụ mới. Nếu chủ sở hữu không còn kinh doanh một homestay nào nữa, họ có thể sử dụng chức năng xóa để loại bỏ homestay đó khỏi danh sách hiển thị. Tất cả các thao tác này đều được thiết kế để giúp chủ sở hữu quản lý danh sách homestay của mình một cách hiệu quả và thuận tiện nhất.

#### 4.1.9 Trang dành riêng cho Admin

Admin sẽ có quyền cao nhất. Admin có thể cấm người dùng nào đó khỏi website của mình. Admin theo dõi doanh thu trang web được hiển thị qua biểu đồ.

#### 4.1.10 Trang chi tiết Homestay

Trang chi tiết của homestay hiện ra sẽ bao gồm hình ảnh chất lượng cao và thông tin chi tiết về homestay đó. Thông tin này có thể bao gồm mô tả về không gian, tiện

## CHƯƠNG 4. THIẾT KẾ GIAO DIỆN

Gnaaq Homestay      Homestay      Search      Welcome, 2      Logout      Menu

**Create Homestay**

Name  Address

Description

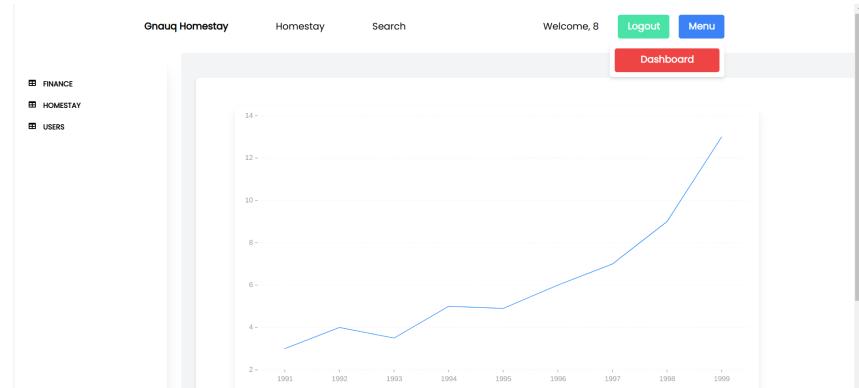
Service

Image URL

Price  0

**Submit**

Hình 4.12: Trang tạo Homestay của Host



Hình 4.13: Trang Dashboard Admin

Gnaaq Homestay      Homestay      Search      Welcome, 8      Logout      Menu

**View List All Homestay**

Homestay	Action
Blue Ocean Home	edit delete
Modem Hotel	edit delete
Vip Hotel	edit delete
Vip Hotel 1	edit delete
TEST1234	edit delete
1	edit delete
Vip Hotel 3	edit delete

< 1 >

Hình 4.14: Trang danh sách homestay được theo dõi bởi Admin

Gnaaq Homestay      Homestay      Search      Welcome, 8      Logout      Menu

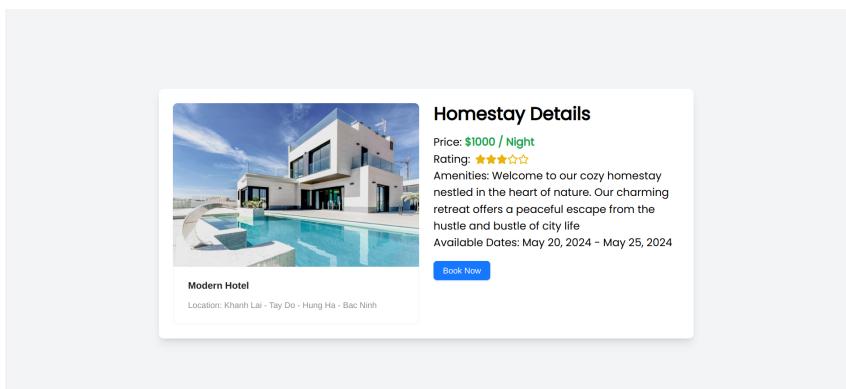
**View List of All Users**

User	Information	Status
Quang Nguyen	Email: quangnguyenengoc32214@gmail.com, Phone: 0376948162	Active
Ha Thu 2K3	Email: tranthuha06012003@gmail.com, Phone: 03124125	Active
Quang Nguyen	Email: quangnguyenengoc314@gmail.com, Phone: 1298342125	Active
Admin	Email: admin@admin.com, Phone: 0376948162	Active
duc	Email: duckie010203@gmail.com, Phone: 0123456789	Active
quangnguyen	Email: tranthuha@gmail.com, Phone: 0123456789	Active

< 1 >

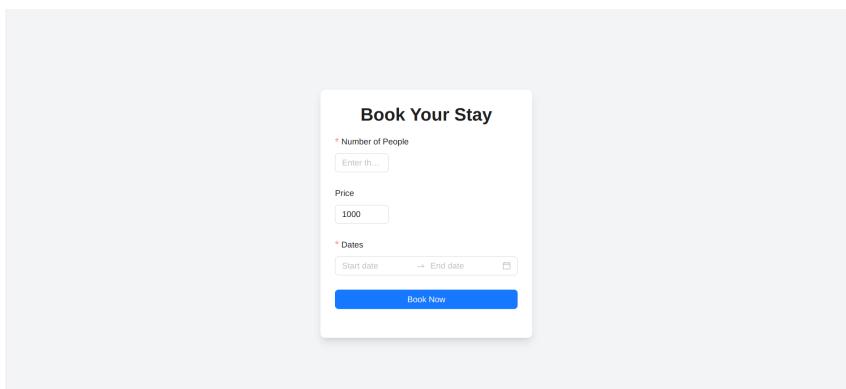
Hình 4.15: Trang theo dõi người dùng bởi Admin

nghi, dịch vụ đi kèm, địa chỉ cụ thể, giá cả và các đánh giá từ khách hàng trước đó. Trang chi tiết này được thiết kế thân thiện và trực quan, giúp khách hàng dễ dàng tìm hiểu và cảm nhận về homestay. Đặc biệt, trên trang này có một nút bấm "Book" nổi bật, cho phép khách hàng nhanh chóng tiến hành đặt chỗ cho kỳ nghỉ của mình chỉ với vài thao tác đơn giản. Nút "Book" này sẽ dẫn khách hàng tới quy trình đặt chỗ, đảm bảo trải nghiệm liền mạch và thuận tiện.



**Hình 4.16:** Trang chi tiết homestay

#### 4.1.11 Trang đặt phòng



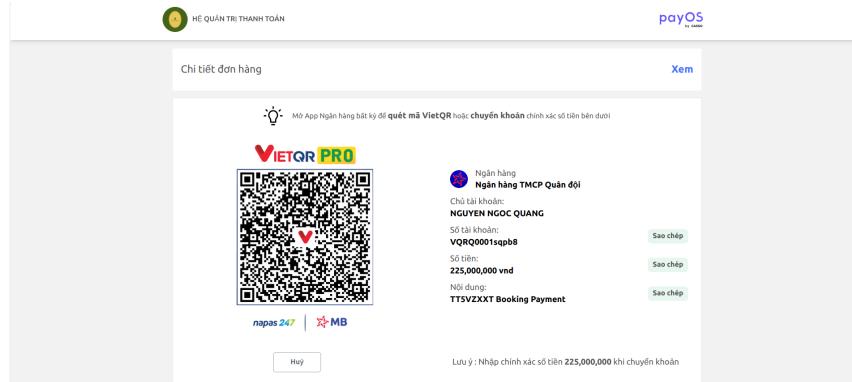
**Hình 4.17:** Trang đặt phòng

Trang đặt phòng được thiết kế thân thiện và dễ sử dụng, giúp khách hàng điền đầy đủ thông tin cần thiết để hoàn tất quá trình đặt phòng. Trên trang này, khách hàng sẽ thấy các ô để điền thông tin về số lượng người dự kiến ở. Ngoài ra, trang còn yêu cầu khách hàng nhập ngày check-in và check-out cụ thể để xác định khoảng thời gian lưu trú.

#### 4.1.12 Trang thanh toán

Sau khi nhấn "Book" ở trang đặt phòng, trang thanh toán sẽ hiện ra. Tại đây, người dùng có thể hoàn tất thanh toán qua tài khoản ngân hàng. Quy trình thanh toán được thiết kế đơn giản và an toàn.

## CHƯƠNG 4. THIẾT KẾ GIAO DIỆN



**Hình 4.18:** Trang thanh toán

## CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1 Kết luận

Việc xây dựng và giải quyết vấn đề đặt phòng cho khách du lịch không chỉ giúp họ dễ dàng tìm kiếm và đặt chỗ ở trước khi đến một địa điểm mới, mà còn mang lại sự tiện lợi và an tâm cho họ trong hành trình du lịch. Trong quá trình thực hiện đề tài này, em đã học được rất nhiều về các phương pháp thiết kế giao diện người dùng sao cho thân thiện và trực quan, giúp người dùng có thể dễ dàng thao tác. Em cũng đã nắm bắt được nhiều công nghệ mới và các framework hiện đại, từ đó áp dụng để phát triển trang web một cách hiệu quả và chuyên nghiệp. Bên cạnh đó, em cũng tích lũy được nhiều kinh nghiệm quý báu trong việc xử lý các vấn đề kỹ thuật, tối ưu hóa hiệu suất trang web, và đảm bảo tính bảo mật cho thông tin người dùng. Quá trình này không chỉ giúp em nâng cao kỹ năng lập trình mà còn mở rộng kiến thức về quản lý dự án và làm việc nhóm, từ việc phân tích yêu cầu, thiết kế hệ thống, đến triển khai và kiểm thử sản phẩm. Những kinh nghiệm và kiến thức này sẽ là nền tảng vững chắc để em tiếp tục phát triển trong lĩnh vực công nghệ thông tin.

### 5.2 Hướng phát triển

Trang web sinh ra nhằm mục đích giải quyết việc đặt phòng cho khách du lịch. Mục tiêu ngắn hạn là thêm nhiều chức năng cho người dùng

- Thêm chức năng bình luận vào bài viết
- Thêm chức năng lấy lại mật khẩu
- Thêm chức năng đánh giá Homestay
- Thêm chức năng theo dõi tài chính của người dùng

Mục tiêu dài hạn là

- Deploy hệ thống lên server
- Thu hút vốn đầu tư và nhận được nhiều sự tham gia của người dùng
- Cải thiện và tối ưu hiệu suất trang web

## TÀI LIỆU THAM KHẢO

- [?] NestJS Documentation, *NestJS - A progressive Node.js framework*. [Online]. Available: <https://docs.nestjs.com/> (visited on 05/22/2024).
- [?] React Documentation, *React - Learn React*. [Online]. Available: <https://react.dev/learn> (visited on 05/22/2024).
- [?] TypeORM Documentation, *TypeORM - An ORM for TypeScript and JavaScript*. [Online]. Available: <https://typeorm.io/> (visited on 05/22/2024).
- [?] Docker Documentation, *Docker Documentation*. [Online]. Available: <https://docs.docker.com/> (visited on 05/22/2024).
- [?] PostgreSQL Documentation, *PostgreSQL Documentation*. [Online]. Available: <https://www.postgresql.org/docs/> (visited on 05/22/2024).