**1. Nghiên cứu vấn đề**

* 1. **Giới thiệu chung:**

1. **Xuất xứ website:**

Website Alpine Ascents Vietnam được thành lập với mục tiêu mang đến những trải nghiệm du lịch mạo hiểm độc đáo tại các địa danh nổi tiếng ở Việt Nam. Kế thừa tinh hoa từ Alpine Ascents International và được thiết kế đặc biệt để hòa hợp với vẻ đẹp tự nhiên và di sản văn hóa đặc sắc của Việt Nam, Alpine Ascents Vietnam mang đến những trải nghiệm thám hiểm đỉnh cao, từ hành trình chinh phục đỉnh Fansipan hùng vĩ, khám phá hệ thống hang động kỳ vĩ tại Phong Nha - Kẻ Bàng, tận hưởng vẻ đẹp huyền bí của Bà Nà, cho đến những chuyến trekking thú vị tại Núi Chúa.

1. **Lý do cần thiết lập trình website này:**

Trong bối cảnh du lịch mạo hiểm ngày càng phát triển tại Việt Nam, nhu cầu về một nền tảng trực tuyến chuyên nghiệp và toàn diện cho việc tìm kiếm, đặt chỗ, và tham gia các chuyến thám hiểm ngày càng tăng cao. Website Alpine Ascents Vietnam không chỉ là một kênh để giới thiệu các tour thám hiểm mà còn là nơi cung cấp thông tin, kiến thức, và cộng đồng cho những người đam mê du lịch mạo hiểm. Việc thiết lập website này sẽ giúp công ty tiếp cận khách hàng dễ dàng hơn, nâng cao uy tín thương hiệu, và tạo ra sự khác biệt trên thị trường.

* 1. **Hệ thống website hiện tại**

Hiện nay, trên thị trường Việt Nam có một số công ty cung cấp dịch vụ du lịch mạo hiểm và thám hiểm như Oxalis Adventure, Viettrekking, và Adventure Vietnam.

1. **Điểm mạnh:**

* Giao diện trực quan, dễ sử dụng.
* Chức năng giới thiệu tour, đặt chỗ trực tuyến.
* Cung cấp thông tin chi tiết về các chuyến đi và trang thiết bị cần thiết.

1. **Điểm yếu của đối thủ:**

* Thiếu nội dung blog phong phú, chuyên sâu về trải nghiệm thực tế.
* Giao diện chưa thật sự thân thiện với người dùng trên mọi thiết bị.
* Ít tương tác với khách hàng, không có cộng đồng trực tuyến.
  1. **Hệ thống dự kiến bao gồm các chức năng:**
     1. **Các chương trình tour thám hiểm:**
     2. Mô tả:
* Hiển thị danh sách các tour thám hiểm hiện có, bao gồm tên tour, giá, hình ảnh, mô tả ngắn gọn về tour.
* Chi tiết từng tour sẽ bao gồm thông tin đầy đủ như địa điểm, yêu cầu, hành trình, trang bị cần thiết, và những lưu ý khi tham gia.
* Khách hàng có thể xem thông tin về tour, sau đó thực hiện đặt tour nếu có nhu cầu.
  + 1. Chức năng:
* Danh sách tour
* Trang chi tiết tour với thông tin chi tiết và hình ảnh.
* Nút "Đặt tour" chuyển đến form đặt hàng.
  + 1. **Quản lý các tour thám hiểm:**

1. Mô tả:

* Admin có thể quản lý danh sách các tour thám hiểm, thêm mới, chỉnh sửa, hoặc xóa tour.
* Admin có thể cập nhật các thông tin về giá cả, hình ảnh, và thông tin chi tiết khác về tour.
* Admin có quyền hủy các tour.

1. Chức năng:

CRUD (Create, Read, Update, Delete) cho các tour thám hiểm.

Cập nhật giá.

Quản lý hình ảnh, mô tả và các thông tin chi tiết về tour.

* + 1. **Gallery**

1. Mô tả:

* Khách hàng có thể xem thư viện hình ảnh và video về các tour, được phân chia theo từng tour thám hiểm.
* Thư viện được chia thành các album riêng lẻ cho từng tour, người dùng có thể xem ảnh và video liên quan đến từng tour.

1. Chức năng:

* Hiển thị danh sách các album tương ứng với từng tour.
* Hỗ trợ chế độ xem slideshow.
  + 1. **Quản lý Gallery**

1. Mô tả:

* Admin có quyền quản lý thư viện hình ảnh và video của từng tour.
* Admin có thể thêm mới, cập nhật, hoặc xóa các hình ảnh, video liên quan đến các tour thám hiểm.
* Quản lý các thư mục hoặc album theo từng tour.

1. Chức năng:

* Thêm, xóa, và cập nhật ảnh và video cho từng tour.
* Tạo và quản lý các album (thư mục) cho từng tour.
  + 1. **Branches:**

1. Mô tả:

* Hiển thị danh sách các chi nhánh của công ty ở các quốc gia hoặc khu vực khác nhau.
* Mỗi chi nhánh sẽ có thông tin về địa chỉ, số điện thoại, và vị trí trên bản đồ.
* Tích hợp Maps API để người dùng có thể xem vị trí chính xác của từng chi nhánh.

1. Chức năng:

* Hiển thị thông tin chi nhánh như tên, địa chỉ, số điện thoại.
* Xem vị trí chi nhánh trên bản đồ.
  + 1. **Quản lý Branches:**

1. Mô tả:

* Admin có quyền thêm, xóa, và chỉnh sửa thông tin các chi nhánh.
* Quản lý thông tin chi tiết của chi nhánh như tên, địa chỉ, số điện thoại, và vị trí địa lý.

1. Chức năng:

* CRUD (Create, Read, Update, Delete) cho thông tin các chi nhánh.
* Cập nhật vị trí tọa độ (latitude, longitude) cho bản đồ.
  + 1. **Information**

1. Mô tả:

* Cung cấp thông tin chung về leo núi: lịch sử, phân loại, kỹ thuật….
* Thông tin có thể bao gồm giới thiệu về công ty, sứ mệnh, giá trị cốt lõi, và các thông tin liên hệ.

1. Chức năng:

* Hiển thị thông tin tĩnh về công ty.
* Là nơi tra cứu, nghiên cứu các thông tin hỗ trợ về leo núi.
  + 1. **Quản lý tài khoản cho người dùng**

1. Mô tả:

* Người dùng có thể đăng ký, đăng nhập, và quản lý tài khoản của họ.
* Người dùng có thể chỉnh sửa thông tin cá nhân như tên, số điện thoại, email, mật khẩu.

1. Chức năng:

* Đăng ký, đăng nhập, và đăng xuất tài khoản.
* Quản lý thông tin cá nhân và cập nhật thông tin.
  + 1. **Quản lý tài khoản cho admin**

1. Mô tả:

* Admin có quyền quản lý thông tin tài khoản của tất cả người dùng trên hệ thống.
* Quản lý phân quyền giữa các tài khoản admin và người dùng thường.
* Admin có thể thêm mới hoặc xóa tài khoản người dùng, cập nhật thông tin và reset mật khẩu.

1. Chức năng:

* CRUD cho tài khoản người dùng (thêm, xóa, sửa).
* Phân quyền người dùng (admin hoặc user).
* Quản lý trạng thái tài khoản (kích hoạt hoặc chưa kích hoạt).

* + 1. **Đặt hàng**

1. Mô tả:

* Người dùng có thể đặt tour thông qua form đặt hàng.
* Form bao gồm thông tin cá nhân của khách hàng (tên, số điện thoại, email) và thông tin chi tiết về tour (tên tour, số lượng người tham gia, tổng giá trị đơn hàng).
* Sau khi đặt hàng, người dùng nhận được thông báo về kết quả đặt hàng (thành công hoặc lỗi).

1. Chức năng:

* Form đặt hàng với các trường thông tin cá nhân và chi tiết tour.
* Tính toán tổng giá trị đơn hàng dựa trên số lượng và giá tour.
* Lưu trữ đơn hàng vào cơ sở dữ liệu và hiển thị kết quả.
  + 1. **Quản lý đơn hàng**

1. Mô tả:

* Admin có thể xem danh sách tất cả các đơn hàng được đặt bởi khách hàng.
* Admin có thể xem chi tiết từng đơn hàng bao gồm thông tin khách hàng, tour được đặt, số lượng người, và tổng giá trị.

1. Chức năng:

* Hiển thị danh sách đơn hàng với thông tin chi tiết.
* Cập nhật và thay đổi thông tin đơn hàng.
* Xem chi tiết từng đơn hàng, bao gồm thông tin người dùng và tour đặt.
  1. **Giới hạn của hệ thống:**
* Chưa hỗ trợ người dùng cập nhật, chỉnh sửa avatar: Hiện tại, hệ thống chưa có tính năng cho phép người dùng tải lên hoặc thay đổi avatar của họ.
* Chưa thống kê được lịch sử đơn hàng và đặt hàng của người dùng: Hệ thống không cung cấp tính năng theo dõi lịch sử các tour đã đặt của người dùng.
* Chưa có thanh toán trực tuyến: Hiện tại, hệ thống chưa tích hợp thanh toán trực tuyến.
* Hạn chế trong tính năng tìm kiếm người dùng: Hệ thống có thể chưa có tính năng tìm kiếm người dùng (user) hoặc quản trị viên (admin) theo các tiêu chí cụ thể như tên, email, hoặc trạng thái tài khoản.
* Chưa có tính năng giỏ hàng: Hệ thống chưa cung cấp tính năng giỏ hàng để cho phép người dùng thêm nhiều tour vào giỏ và xử lý thanh toán cho tất cả cùng một lúc.
* Chưa có tính năng lọc đơn hàng theo các tiêu chí: Hệ thống chưa có khả năng lọc đơn hàng theo trạng thái, ngày đặt hoặc các tiêu chí khác.
* Chưa xử lý được các đơn hàng rác: Hệ thống chưa có các cơ chế để phát hiện và xử lý các đơn hàng rác hoặc đơn hàng không hợp lệ.
* Chưa có tính năng quản lý trạng thái đơn hàng: Hệ thống chưa hỗ trợ cập nhật hoặc thay đổi trạng thái đơn hàng (ví dụ: chờ xác nhận, đã thanh toán, đã hoàn thành).
* Chưa có tính năng lọc tours theo các tiêu chí: Hệ thống chưa cho phép người dùng lọc tours theo các tiêu chí như vị trí, giá cả, thời gian đi, hay tính chất của tour.
* Chưa có tính năng lọc chi nhánh theo các vị trí địa lý cụ thể: Hệ thống chưa cung cấp chức năng lọc chi nhánh theo vị trí địa lý.
* Chưa có chức năng báo cáo và thống kê: Hệ thống hiện tại chưa cung cấp các báo cáo và thống kê về doanh số, số lượng tour bán ra, hoặc hoạt động của người dùng.
* Chưa có chức năng phân quyền chi tiết và vai trò cụ thể: Hệ thống chưa hỗ trợ phân quyền chi tiết cho admin và các vai trò cụ thể (ví dụ: admin chỉ quản lý gallery, admin chỉ quản lý đơn hàng).
* Chưa test với số lượng bản ghi trên 1 triệu: Hệ thống chưa được kiểm tra hiệu năng và khả năng xử lý với số lượng bản ghi lớn.
* Chưa hỗ trợ đa ngôn ngữ: Hệ thống hiện tại chỉ hỗ trợ một ngôn ngữ duy nhất, không có khả năng chuyển đổi giữa các ngôn ngữ khác nhau.
* Chưa có tính năng bảo mật nâng cao (2FA, mã hóa dữ liệu): Hệ thống chưa tích hợp các phương thức bảo mật nâng cao như xác thực hai yếu tố (2FA) hoặc mã hóa dữ liệu cá nhân và giao dịch.
* Chưa có hệ thống thông báo (notifications): Hệ thống chưa hỗ trợ tính năng gửi thông báo (notifications) qua email hoặc thông báo đẩy (push notifications) khi có sự kiện quan trọng như đặt hàng thành công, hủy đơn hàng, hoặc cập nhật trạng thái đơn hàng.
* Chưa có hệ thống phản hồi và đánh giá: Hệ thống chưa cung cấp tính năng để người dùng có thể phản hồi hoặc đánh giá các tour đã tham gia.
* Chưa có hệ thống khuyến mãi và mã giảm giá: Hệ thống hiện tại chưa có chức năng quản lý các chương trình khuyến mãi hoặc mã giảm giá cho người dùng khi đặt tour.
* Chưa có hệ thống lưu trữ và xử lý log lỗi: Hệ thống chưa có tính năng lưu trữ và phân tích các log lỗi để phục vụ cho việc phát hiện và sửa lỗi một cách nhanh chóng.
  1. **Yêu cầu về phần cứng và phần mềm**

**Hardware**

Intel Core i3/i5 Processor or higher

8 GB RAM or above

Color SVGA

500 GB Hard Disk space

Mouse

Keyboard

**Software**

Operating Portal: Windows

Browsers: Edge, Chrome, Mozilla Firefox, Safari

1. **Phân tích yêu cầu**
   1. **Người dùng của hệ thống:**

**Khách hàng:** Những người quan tâm đến các chuyến thám hiểm, trekking và leo núi. Họ sử dụng website để tìm kiếm thông tin, đặt tour, chiêm ngưỡng thư viện ảnh, tra cứu danh sách các chi nhánh trên thế giới, nghiên cứu những thông tin cần thiết liên quan đến leo núi. Họ cũng có thể quản lý thông tin cá nhân khi đăng ký và đăng nhập vào hệ thống.

**Quản trị viên:** Những người quản lý website, thêm mới và cập nhật thông tin tour, xử lý các đơn đặt chỗ, quản lý thư viện ảnh, quản lý danh sách đối tác, quản lý thông tin khách hàng.

* 1. **Các chức năng của hệ thống**

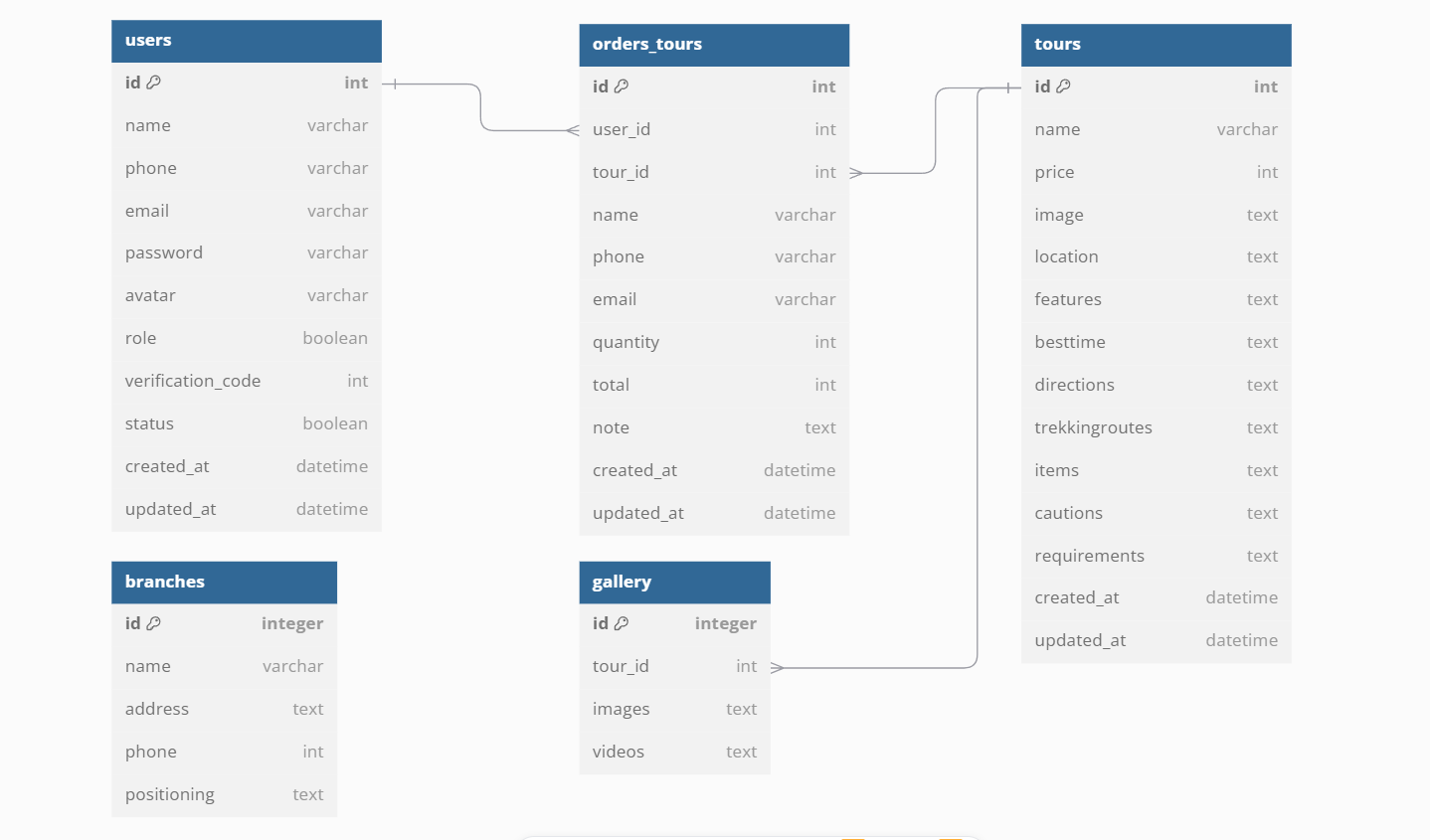
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chức năng** | **Thông tin đầu vào** | **Thông tin đầu ra** | **Cách thức xử lý** | **Dữ liệu cần lưu trữ** |
| **Tours** | - Danh sách các chương trình tour, bao gồm tên tour, mô tả, giá cả, lịch trình, và các thông tin đặc biệt khác | - Danh sách các tour có thể đặt, với chi tiết mô tả và hình ảnh  - Các tùy chọn để đặt tour trực tuyến | - Truy xuất dữ liệu từ database về các tour  - Hiển thị thông tin chi tiết về từng tour  - Cung cấp tùy chọn đặt tour | - Thông tin về các chương trình tour (tên, mô tả, giá cả, lịch trình)  - Hình ảnh và video liên quan đến các tour |
| **Quản lý tours** | - Thông tin về tour: tên tour, giá, hình ảnh đại diện, địa điểm, đặc điểm nổi bật, thời gian đẹp nhất để đi, hướng dẫn đường đi, lộ trình trekking, vật dụng cần mang theo, cảnh báo, yêu cầu tham gia đối với khách hàng.  - Yêu cầu của người dùng (admin) khi thêm, chỉnh sửa, hoặc xóa tour. | - Danh sách tất cả các tour, hiển thị chi tiết từng tour (tên, giá, địa điểm, ...).  - Chi tiết về một tour cụ thể khi người dùng yêu cầu.  - Kết quả của việc thêm, chỉnh sửa, hoặc xóa tour (thành công hay lỗi). | - Người dùng (admin) gửi yêu cầu thêm, chỉnh sửa, hoặc xóa thông tin tour.  - Hệ thống kiểm tra dữ liệu đầu vào: tất cả các trường bắt buộc (tên, giá, địa điểm, ...) phải được điền đầy đủ.  - Nếu dữ liệu hợp lệ, hệ thống thực hiện các thao tác với database (thêm, sửa, xóa).  - Hệ thống trả kết quả về giao diện người dùng. | - Bảng tours: lưu trữ thông tin chi tiết về mỗi tour.  - Các trường dữ liệu: id, name, price, image, location, features, besttime, directions, trekkingroutes, items, cautions, requirements, created\_at, updated\_at. |
| **Chức năng** | **Thông tin đầu vào** | **Thông tin đầu ra** | **Cách thức xử lý** | **Dữ liệu cần lưu trữ** |
| **Gallery** | - Hình ảnh, video, các liên kết (ví dụ: YouTube) liên quan đến từng tour. | - Trang hiển thị các hình ảnh và video của từng tour. | - Hiển thị ảnh dưới dạng slideshow theo từng tour. | images (lưu trữ đường dẫn ảnh)  videos (lưu liên kết đến video YouTube)  tours\_id (liên kết đến tour cụ thể) |
| **Quản lý gallery** | - Danh sách hình ảnh và video cho mỗi tour.  - ID của tour liên kết đến gallery.  - URL hoặc liên kết YouTube cho video. | - Danh sách các gallery tương ứng với từng tour.  - Danh sách hình ảnh và video thuộc về từng tour.  - Kết quả của việc thêm hoặc xóa hình ảnh/video trong gallery. | - Khi người dùng truy cập trang gallery của một tour, hệ thống lấy danh sách hình ảnh và video của tour từ bảng galleries.  - Nếu admin cần thêm/xóa ảnh hoặc video, họ sẽ gửi yêu cầu qua giao diện quản trị.  - Hệ thống kiểm tra tính hợp lệ của dữ liệu (hình ảnh phải có định dạng hợp lệ, video phải có link hợp lệ).  - Cập nhật bảng galleries và lưu lại thông tin mới. | - Bảng galleries: lưu trữ các liên kết đến hình ảnh và video cho từng tour.  - Các trường dữ liệu: id, tours\_id, images (lưu trữ dưới dạng JSON chứa danh sách link ảnh), videos (lưu trữ dưới dạng JSON chứa danh sách link video), created\_at, updated\_at. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chức năng** | **Thông tin đầu vào** | **Thông tin đầu ra** | **Cách thức xử lý** | **Dữ liệu cần lưu trữ** |
| **Branches** | Tên chi nhánh, địa chỉ, số điện thoại, vị trí địa lý. | Danh sách các chi nhánh kèm bản đồ hiển thị vị trí của mỗi chi nhánh. | - Lấy thông tin chi nhánh từ database.  - Sử dụng API để hiển thị vị trí các chi nhánh trên bản đồ. | name, address,  phone, positioning |
| **Quản lý branches** | - Thông tin chi nhánh: tên, địa chỉ, số điện thoại, thông tin định vị (positioning).  - Yêu cầu của admin khi thêm, chỉnh sửa, hoặc xóa chi nhánh. | - Danh sách các chi nhánh của công ty trên toàn thế giới.  - Chi tiết về một chi nhánh cụ thể.  - Kết quả của việc thêm, chỉnh sửa, hoặc xóa chi nhánh (thành công hoặc lỗi). | - Người dùng (admin) gửi yêu cầu thêm, chỉnh sửa, hoặc xóa thông tin chi nhánh.  - Hệ thống kiểm tra các trường thông tin bắt buộc (tên, địa chỉ, số điện thoại).  - Hệ thống thực hiện thao tác với database (thêm, sửa, hoặc xóa).  - Trả kết quả về cho admin. | - Bảng branches: lưu trữ thông tin chi nhánh.  - Các trường dữ liệu: id, name, address, phone, positioning (thông tin định vị), created\_at, updated\_at. |
| **Information** | Thông tin liên quan đến lịch sử, loại hình, kỹ thuật, lưu trú, mối nguy hiểm của các hoạt động leo núi, trekking. | Hiển thị các bài viết, thông tin theo danh mục. | - Lưu trữ thông tin liên quan dưới dạng bài viết (blogs).  - Phân loại và hiển thị theo danh mục cụ thể. | title, content, category, status |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chức năng** | **Thông tin đầu vào** | **Thông tin đầu ra** | **Cách thức xử lý** | **Dữ liệu cần lưu trữ** |
| **Quản lý tài khoản** | - Thông tin tài khoản khi đăng ký: tên, số điện thoại, email, mật khẩu.  - Khi cập nhật thông tin: avatar, trạng thái tài khoản (status), mã xác minh (verification\_code).  - Thông tin đăng nhập: email, mật khẩu.  - Thông tin yêu cầu khôi phục mật khẩu nếu có. | - Thông tin tài khoản người dùng (tên, email, avatar, role).  - Trạng thái đăng nhập: thành công hay thất bại.  - Kết quả của việc đăng ký, cập nhật tài khoản, hoặc thay đổi mật khẩu.  - Thông báo xác minh hoặc khôi phục tài khoản nếu có. | - Khi người dùng đăng ký tài khoản, hệ thống kiểm tra xem email và số điện thoại đã tồn tại chưa.  - Nếu hợp lệ, tạo tài khoản mới và lưu thông tin vào database.  - Hệ thống sẽ gửi mã xác minh qua email, nếu xác minh đúng sẽ thay đổi trạng thái tài khoản đang hoạt động.  - Khi người dùng đăng nhập, hệ thống kiểm tra email, mật khẩu và trả về kết quả đăng nhập.  - Khi người dùng cập nhật thông tin, hệ thống xác nhận quyền truy cập và lưu thay đổi vào database.  - Nếu người dùng yêu cầu khôi phục mật khẩu, hệ thống sẽ gửi mã xác minh qua email. | - Bảng users: lưu thông tin tài khoản người dùng.  - Các trường dữ liệu: id, name, phone, email, password, avatar, role, verification\_code, status, created\_at, updated\_at. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chức năng** | **Thông tin đầu vào** | **Thông tin đầu ra** | **Cách thức xử lý** | **Dữ liệu cần lưu trữ** |
| **Quản lý người dùng** | - Thông tin người dùng như tên, email, số điện thoại, avatar, mật khẩu, vai trò. | - Hiển thị danh sách người dùng, chi tiết từng người dùng. | - Tạo, cập nhật, xóa thông tin người dùng.  - Quản lý quyền truy cập (admin, người dùng thường). | - name, email, phone, password, avatar, role |
| **Đặt hàng** | - Thông tin khách hàng khi đặt hàng: tên, số điện thoại, email, ghi chú (nếu có).  - Thông tin về tour được đặt: ID tour, số lượng, tổng giá trị đơn hàng. | - Danh sách các đơn hàng đã đặt.  - Thông tin chi tiết về một đơn hàng cụ thể (tên khách hàng, tour đặt, số lượng, giá trị đơn hàng).  - Kết quả của việc đặt hàng (thành công hay lỗi). | - Khi người dùng đặt hàng, hệ thống lấy thông tin tour, số lượng, và tính tổng giá trị đơn hàng.  - Người dùng nhập thông tin cá nhân.  - Hệ thống xác nhận đơn hàng và lưu vào database.  - Hệ thống trả về kết quả đặt hàng. | - Bảng orders\_tours: lưu thông tin về đơn hàng.  - Các trường dữ liệu: id, user\_id (ID của người dùng), tour\_id (ID của tour), name (tên khách hàng), phone, email, quantity (số lượng tour), total (tổng giá trị), note, created\_at, updated\_at. |
| **Quản lý đơn hàng** | - Thông tin người đặt hàng, tour được đặt, số lượng, ghi chú. | - Danh sách các đơn hàng đã đặt, chi tiết từng đơn hàng. | - Tạo đơn hàng khi khách đặt tour.  - Tính toán tổng số tiền, quản lý thông tin đơn hàng. | user\_id, tour\_id, quantity, total, note |

1. System Designs
   1. Entity Relationship Diagram



* 1. Database Design
     1. Danh sách bảng:
        1. Bảng **users**
* **Mô tả**: Bảng lưu trữ thông tin người dùng, bao gồm cả thông tin xác minh và vai trò người dùng.
* **Các trường**:
  + id: Khóa chính (Primary Key), kiểu BIGINT.
  + name: Tên người dùng, kiểu VARCHAR.
  + phone: Số điện thoại người dùng, kiểu VARCHAR, có tính duy nhất (UNIQUE).
  + email: Địa chỉ email người dùng, kiểu VARCHAR, có tính duy nhất (UNIQUE).
  + password: Mật khẩu người dùng, kiểu VARCHAR.
  + avatar: Đường dẫn đến ảnh đại diện người dùng, kiểu VARCHAR, có thể null (nullable).
  + role: Quyền hạn của người dùng, kiểu BOOLEAN, mặc định là false (user thường).
  + verification\_code: Mã xác minh tài khoản, kiểu INTEGER.
  + status: Trạng thái xác thực tài khoản, kiểu BOOLEAN, mặc định là false.
  + timestamps: Gồm created\_at và updated\_at.
* **Mô tả dạng chuẩn hoá 3NF**
  + Khóa chính: id
  + Các thuộc tính còn lại (name, phone, email, password, avatar, role, verification\_code, status) đều phụ thuộc trực tiếp vào khóa chính (id).
  + Bảng này ở dạng chuẩn 3NF vì không có thuộc tính nào phụ thuộc bắc cầu hoặc không phụ thuộc vào khóa chính.
    - 1. **Bảng tours**
* **Mô tả**: Lưu thông tin chi tiết của các tour du lịch, bao gồm mô tả, đặc điểm, các yêu cầu và hướng dẫn.
* **Các trường**:
  + id: Khóa chính (Primary Key), kiểu BIGINT.
  + name: Tên tour, kiểu VARCHAR.
  + price: Giá tour, kiểu INTEGER.
  + image: Hình ảnh đại diện của tour, kiểu TEXT, có thể null.
  + location: Địa điểm của tour, kiểu TEXT.
  + features: Các đặc điểm nổi bật của tour, kiểu TEXT.
  + besttime: Thời gian lý tưởng để đi tour, kiểu TEXT.
  + directions: Hướng dẫn đường đi, kiểu TEXT.
  + trekkingroutes: Các tuyến đường trekking, kiểu TEXT.
  + items: Các vật dụng cần chuẩn bị, kiểu TEXT, có thể null.
  + cautions: Các lưu ý đặc biệt khi tham gia tour, kiểu TEXT, có thể null.
  + requirements: Các yêu cầu cần thiết để tham gia tour, kiểu TEXT.
  + timestamps: Gồm created\_at và updated\_at.
* **Mô tả dạng chuẩn hoá 3NF**
  + Khóa chính: id
  + Các thuộc tính (name, price, image, location, features, besttime, directions, trekkingroutes, items, cautions, requirements) đều phụ thuộc trực tiếp vào khóa chính (id).
  + Bảng này cũng ở dạng chuẩn 3NF vì các thuộc tính đều có quan hệ trực tiếp với khóa chính mà không có phụ thuộc bắc cầu.
    - 1. **Bảng orders\_tours**
* **Mô tả**: Lưu trữ thông tin các đơn hàng của khách liên quan đến việc đặt tour.
* **Các trường**:
  + id: Khóa chính (Primary Key), kiểu BIGINT.
  + user\_id: Khóa ngoại (Foreign Key) tham chiếu đến bảng users, kiểu BIGINT, có thể null.
  + tour\_id: Khóa ngoại (Foreign Key) tham chiếu đến bảng tours, kiểu BIGINT.
  + name: Tên người đặt tour, kiểu VARCHAR.
  + phone: Số điện thoại của người đặt, kiểu VARCHAR, mặc định là 000-000-0000.
  + email: Email của người đặt tour, kiểu VARCHAR, có thể null.
  + quantity: Số lượng người tham gia tour, kiểu INTEGER.
  + total: Tổng số tiền cho đơn hàng, kiểu INTEGER.
  + note: Ghi chú thêm của khách hàng, kiểu TEXT, có thể null.
  + timestamps: Gồm created\_at và updated\_at.
* **Mô tả dạng chuẩn hoá 3NF**
  + Khóa chính: id
  + Khóa ngoại:
    - user\_id: Tham chiếu đến bảng users để xác định người đặt tour.
    - tour\_id: Tham chiếu đến bảng tours để xác định tour mà người dùng đặt.
  + Các thuộc tính khác (name, phone, email, quantity, total, note) đều phụ thuộc trực tiếp vào khóa chính (id).
  + Bảng này có quan hệ bắc cầu với users và tours thông qua các khóa ngoại nhưng vẫn tuân thủ chuẩn 3NF.
    - 1. **Bảng branches**
* **Mô tả**: Lưu trữ thông tin các chi nhánh của công ty trên toàn thế giới.
* **Các trường**:
  + id: Khóa chính (Primary Key), kiểu BIGINT.
  + name: Tên chi nhánh, kiểu VARCHAR.
  + address: Địa chỉ chi nhánh, kiểu TEXT.
  + phone: Số điện thoại liên lạc, kiểu VARCHAR.
  + positioning: Thông tin định vị (có thể là tọa độ GPS) của chi nhánh, kiểu TEXT.
  + timestamps: Gồm created\_at và updated\_at.
* **Mô tả dạng chuẩn hoá 3NF**
  + Khóa chính: id
  + Các thuộc tính (name, address, phone, positioning) đều phụ thuộc trực tiếp vào khóa chính (id).
  + Bảng này ở dạng 3NF.
    - 1. **Bảng galleries**
* **Mô tả**: Lưu trữ thông tin về thư viện hình ảnh và video liên quan đến từng tour.
* **Các trường**:
  + id: Khóa chính (Primary Key), kiểu BIGINT.
  + tours\_id: Khóa ngoại (Foreign Key) tham chiếu đến bảng tours, kiểu BIGINT. Nếu tour bị xóa, toàn bộ gallery liên quan sẽ bị xóa theo (onDelete('cascade')).
  + images: Lưu danh sách ảnh dưới dạng JSON, kiểu JSON, có thể null.
  + videos: Lưu danh sách video dưới dạng JSON (chứa các link YouTube), kiểu JSON, có thể null.
  + timestamps: Gồm created\_at và updated\_at.
* **Mô tả dạng chuẩn hoá 3NF**
  + Khóa chính: id
  + Khóa ngoại: tours\_id: Tham chiếu đến bảng tours để xác định tour nào chứa gallery này.
  + Các thuộc tính khác (images, videos) đều phụ thuộc trực tiếp vào khóa chính (id).
  + Bảng này có quan hệ với bảng tours, nhưng vì nó phụ thuộc trực tiếp vào khóa chính và khóa ngoại, nên vẫn tuân thủ chuẩn 3NF.
    1. Sơ đồ quan hệ giữa các bảng
  + users ↔ orders\_tours: Mối quan hệ giữa người dùng và đơn hàng tour. Mỗi đơn hàng có thể thuộc về một người dùng (1:N).
  + tours ↔ orders\_tours: Mối quan hệ giữa các tour và đơn hàng. Mỗi đơn hàng sẽ thuộc về một tour (1:N).
  + tours ↔ galleries: Mỗi tour có thể có một hoặc nhiều hình ảnh và video trong gallery (1:N).

Dưới đây là sơ đồ ERD (Entity-Relationship Diagram) cho mối quan hệ này:

Users

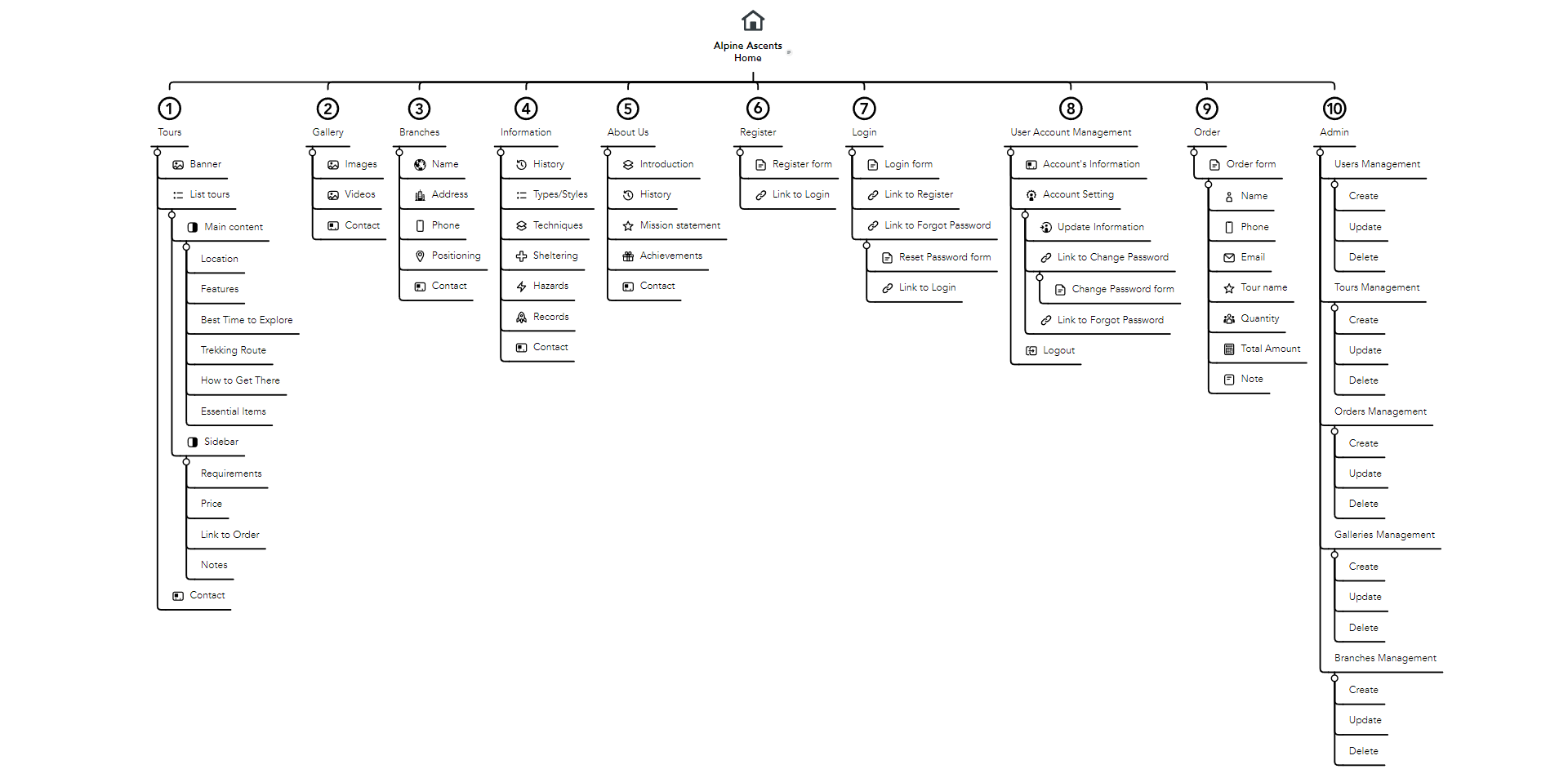
└──(1:N)── Orders\_Tours

Tours

└──(1:N)── Orders\_Tours

└──(1:N)── Galleries

* 1. Sitemap



* 1. **System functions design**
     1. **Chức năng Tours**
        1. **Chi tiết chức năng**

1. **Thông tin đầu vào:**

* Danh sách các chương trình tour:
* Tên tour: tên của từng chương trình tour du lịch (Ví dụ: "Tour Fansipan", "Tour Phong Nha - Kẻ Bàng").
* Mô tả tour: thông tin chi tiết về chương trình, bao gồm các điểm đến, các đặc điểm nổi bật, thời điểm lý tưởng khám phá, các cách di chuyển tới điểm đến, các lộ trình chinh phục, các vật dụng cần thiết…
* Giá tour: chi phí cho từng tour (đơn vị tiền tệ có thể là VND hoặc USD).
* Thông tin về những lưu ý: về thời tiết, các yêu cầu chuẩn bị cá nhân, điều kiện tham gia tour, các lưu ý khác
* Hình ảnh minh họa cho từng điểm đến

1. **Thông tin đầu ra:**

* Danh sách các tour có thể đặt:
* Danh sách các tour hiện có, hiển thị tên, giá, hình ảnh, mô tả ngắn gọn về từng tour.
* Thông tin chi tiết khi click vào từng tour bao gồm hình ảnh đại diện, mô tả dài, các đặc điểm nổi bật, thời điểm lý tưởng khám phá, các cách di chuyển tới điểm đến, các lộ trình chinh phục, các vật dụng cần thiết, điều kiện tham gia, các lưu ý của tour…
* Tùy chọn đặt tour trực tuyến:
* Form đặt tour cho phép khách hàng chọn tour, số lượng người, và ghi chú đặc biệt (nếu có).

1. **Cách thức xử lý:**

* Truy xuất dữ liệu từ database về các tour:

Truy vấn từ database lấy danh sách các tour hiện có, cùng với thông tin về mô tả, giá cả, các thông tin chi tiết về tour.

* Hiển thị thông tin chi tiết về từng tour:

Khi người dùng chọn một tour cụ thể, trang chi tiết sẽ hiển thị thông tin đầy đủ bao gồm hình ảnh đại diện, mô tả dài, các đặc điểm nổi bật, thời điểm lý tưởng khám phá, các cách di chuyển tới điểm đến, các lộ trình chinh phục, các vật dụng cần thiết, điều kiện tham gia, các lưu ý…

* Cung cấp tùy chọn đặt tour:

Hiển thị nút “Đặt tour” để khách hàng có thể truy cập vào form đặt chỗ.

Hệ thống kiểm tra thông tin đầu vào và lưu thông tin đặt chỗ vào database khi khách hàng gửi đơn đặt hàng.

1. **Dữ liệu cần lưu trữ:**

* Thông tin về các chương trình tour:

Bảng tours: lưu trữ tên, mô tả, giá cả và các thông tin đặc biệt khác của tour.

* Hình ảnh liên quan đến các tour:

Bảng galleries: lưu các liên kết đến hình ảnh và video cho từng tour.

* Dữ liệu đặt chỗ và trạng thái tour:

Bảng orders\_tours: lưu trữ thông tin người đặt chỗ, số lượng khách.

* + - 1. **Thiết kế giao diện (UI/UX):**
    1. **Trang danh sách tour:**
* Hiển thị danh sách các tour với hình ảnh đại diện, tên tour, giá và mô tả ngắn.
* Mỗi tour có nút "See More" để chuyển đến trang chi tiết.
  + 1. **Trang chi tiết tour:**
* Mô tả chi tiết về chương trình tour, hiển thị các hình ảnh liên quan.
* Nút "Đặt tour" để truy cập vào form đặt chỗ.
  + 1. **Form đặt chỗ:**
* Cho phép khách hàng chọn tour, chọn số lượng người tham gia và các ghi chú đặc biệt (nếu có).
* Nút "Xác nhận" để gửi đơn đặt hàng.
  + - 1. **Sơ đồ hoạt động (Flowchart)**

1. **Lưu đồ tổng quan chức năng Tours:**

Người dùng truy cập trang danh sách tour

Hiển thị danh sách các tour

Người dùng chọn tour để xem chi tiết

Hiển thị thông tin chi tiết về tour

Người dùng đặt tour

Hiển thị form đặt tour

Kiểm tra thông tin đầu vào

Hiển thị tùy chọn đặt tour

Lưu thông tin đặt tour vào database

Hiển thị thông báo

1. **Mô tả chi tiết các bước:**

* Người dùng truy cập danh sách tour:

Hệ thống lấy danh sách các tour từ database và hiển thị trên trang.

* Hiển thị chi tiết tour:

Khi người dùng click vào một tour, trang chi tiết được hiển thị với thông tin đầy đủ về tour đó.

* Đặt tour:

Người dùng có thể chọn tour, số lượng người tham gia, hệ thống kiểm tra thông tin đầu vào và lưu thông tin vào bảng orders\_tours.

* Xử lý đặt tour:

Hệ thống hiển thị thông báo thành công và lưu đơn đặt hàng.

* + - 1. **Thuật toán giải quyết:**
* Truy xuất dữ liệu tour từ database:

Sử dụng truy vấn SQL để lấy thông tin từ bảng tours, bao gồm tên, mô tả, giá, lịch trình và hình ảnh.

* Lưu thông tin đặt tour:

Sau khi kiểm tra thông tin đầu vào hợp lệ, đơn hàng sẽ được lưu vào bảng orders\_tours.

* + 1. Chức năng Quản lý tours
       1. Chi tiết chức năng

1. Thông tin đầu vào:

* Thông tin về tours:
  + Tên tour: Tên của chương trình tour (Ví dụ: "Tour Fansipan").
  + Giá tour: Chi phí cho từng tour.
  + Hình ảnh đại diện: Ảnh minh họa chính của tour.
  + Địa điểm: Vị trí của tour, ví dụ như núi, vườn quốc gia, di tích lịch sử.
  + Đặc điểm nổi bật: Các điểm nổi bật của tour, ví dụ như phong cảnh đẹp, thử thách đặc biệt.
  + Thời gian đẹp nhất để đi: Khoảng thời gian tốt nhất trong năm để tham gia tour (Ví dụ: mùa khô, mùa thu).
  + Hướng dẫn đường đi: Chỉ dẫn chi tiết về cách đến được điểm bắt đầu của tour.
  + Lộ trình trekking: Mô tả chi tiết đường trekking của tour.
  + Vật dụng cần mang theo: Danh sách các vật dụng cần chuẩn bị cho khách.
  + Cảnh báo: Cảnh báo về độ khó, thời tiết, sức khỏe.
  + Yêu cầu tham gia: Các điều kiện bắt buộc để tham gia tour (Ví dụ: yêu cầu về sức khỏe, độ tuổi).
* Yêu cầu của người dùng (admin):

Yêu cầu thêm, chỉnh sửa hoặc xóa thông tin tour từ hệ thống quản trị.

1. Thông tin đầu ra:

* Danh sách các tour: Hiển thị danh sách tất cả các tour bao gồm tên, giá, địa điểm, và các thông tin cơ bản.
* Chi tiết một tour: Hiển thị thông tin đầy đủ về một tour khi người dùng yêu cầu.
* Kết quả thêm, sửa, xóa tour: Thông báo cho người dùng về kết quả (thành công hoặc thất bại).

1. Cách thức xử lý

* Thêm tour mới:
  + Người dùng (admin) truy cập vào trang thêm mới tour.
  + Điền đầy đủ các thông tin bắt buộc.
  + Hệ thống kiểm tra dữ liệu, nếu hợp lệ, thông tin sẽ được lưu vào bảng tours trong database.
  + Hiển thị thông báo thêm tour thành công.
* Chỉnh sửa tour:
  + Người dùng (admin) chọn một tour cần chỉnh sửa, hệ thống lấy thông tin hiện tại từ database.
  + Admin cập nhật các thông tin cần thiết và gửi yêu cầu.
  + Hệ thống kiểm tra tính hợp lệ, nếu hợp lệ thì cập nhật lại trong bảng tours.
  + Hiển thị thông báo chỉnh sửa thành công.
* Xóa tour:
  + Người dùng (admin) chọn một tour để xóa.
  + Hệ thống kiểm tra xem tour có liên kết dữ liệu nào khác không (ví dụ: có đơn đặt hàng liên quan hay không).
  + Nếu không có liên kết, xóa tour khỏi bảng tours.
  + Hiển thị thông báo xóa thành công.

1. Dữ liệu cần lưu trữ

Bảng tours: các trường dữ liệu:

* id: ID duy nhất của tour.
* name: Tên tour.
* price: Giá tour.
* image: Hình ảnh đại diện.
* location: Địa điểm của tour.
* features: Đặc điểm nổi bật.
* besttime: Thời gian đẹp nhất để đi.
* directions: Hướng dẫn đường đi.
* trekkingroutes: Lộ trình trekking.
* items: Vật dụng cần mang theo.
* cautions: Cảnh báo về tour.
* requirements: Yêu cầu tham gia tour.
* created\_at: Ngày tạo.
* updated\_at: Ngày cập nhật.
  + - 1. Thiết kế giao diện (UI/UX)

1. Giao diện thêm mới tour:

* Form nhập các trường thông tin như tên, giá, địa điểm, hình ảnh, đặc điểm nổi bật, ...
* Nút "Lưu tour" để thêm mới thông tin tour vào hệ thống.

1. Giao diện chỉnh sửa tour:

* Hiển thị các thông tin hiện tại của tour, cho phép admin cập nhật.
* Nút "Cập nhật" để lưu thay đổi.

1. Giao diện danh sách tour:

* Hiển thị danh sách tất cả các tour với thông tin cơ bản.
* Nút "Chỉnh sửa" hoặc "Xóa" để thực hiện các hành động tương ứng.

1. Giao diện chi tiết tour:

* Hiển thị thông tin chi tiết của từng tour khi người dùng chọn.
  + - 1. Sơ đồ hoạt động (Flowchart)

1. Lưu đồ tổng quan chức năng Quản lý Tours:

Admin truy cập chức năng quản lý tours

Lấy dữ liệu từ database

Hiển thị danh sách tours

Chọn hành động: Thêm, Sửa, Xoá tours

Sửa tour hiện có

Thêm tour mới

Xoá tour

Lấy dữ liệu tour

Điền thông tin tour mới

Kiểm tra liên kết và xoá

Chỉnh sửa thông tin

Kiểm tra tính hợp lệ của dữ liệu

Xoá tour khỏi database

Cập nhật thông tin tour vào database

Lưu tour mới vào database

Hiển thị thông báo

1. Mô tả chi tiết các bước:

* Thêm tour mới:
* Admin nhập thông tin tour vào form thêm mới.
* Hệ thống kiểm tra dữ liệu, nếu hợp lệ sẽ lưu thông tin vào bảng tours và hiển thị thông báo thành công.
* Chỉnh sửa tour:
* Admin chọn một tour hiện có, hệ thống tải dữ liệu hiện tại.
* Admin cập nhật thông tin và hệ thống kiểm tra tính hợp lệ, sau đó lưu lại thông tin mới.
* Xóa tour:
* Admin chọn tour cần xóa, hệ thống kiểm tra xem tour có liên kết với dữ liệu khác không.
* Nếu không, hệ thống sẽ xóa tour và hiển thị thông báo thành công.
  + - 1. Thuật toán giải quyết:

1. Thêm tour mới:

* Bước 1: Nhận dữ liệu từ form nhập liệu.
* Bước 2: Kiểm tra các trường bắt buộc như tên, giá, địa điểm.
* Bước 3: Nếu hợp lệ, lưu dữ liệu vào bảng tours.

1. Chỉnh sửa tour:

* Bước 1: Lấy dữ liệu tour hiện tại từ database.
* Bước 2: Admin chỉnh sửa các thông tin và gửi yêu cầu cập nhật.
* Bước 3: Kiểm tra dữ liệu, nếu hợp lệ sẽ cập nhật vào bảng tours.

1. Xóa tour:

* Bước 1: Admin chọn tour cần xóa.
* Bước 2: Kiểm tra liên kết dữ liệu (ví dụ: có đơn đặt hàng liên quan không).
* Bước 3: Nếu không có liên kết, xóa tour khỏi bảng tours.
  + 1. Chức năng gallery
       1. Chi tiết chức năng
    2. Thông tin đầu vào:
* Hình ảnh (Images): Tệp hình ảnh được tải lên từ người dùng (admin) hoặc đường dẫn tới hình ảnh đã được lưu trữ.
* Video (Videos): Liên kết đến video trên các nền tảng trực tuyến (ví dụ: YouTube) hoặc video được tải lên trực tiếp.
* Liên kết với Tour: Từng hình ảnh và video phải được liên kết với một tour cụ thể (bằng trường tours\_id).
  + 1. Thông tin đầu ra:
* Trang hiển thị gallery: Hiển thị danh sách tất cả các hình ảnh và video liên quan đến từng tour dưới dạng slideshow hoặc carousel.
  + 1. Cách thức xử lý:

Hiển thị gallery:

* Khi người dùng truy cập trang gallery, hệ thống truy vấn hình ảnh và video liên kết từ database.
* Tất cả hình ảnh và video sẽ được hiển thị dưới dạng slideshow hoặc carousel.
* Video sẽ được hiển thị dưới dạng liên kết YouTube, có thể nhúng và phát trực tiếp trên trang.
  + 1. Dữ liệu cần lưu trữ:

Không cần lưu trữ dữ liệu mới.

* + - 1. Thiết kế giao diện (UI/UX)

Giao diện hiển thị gallery:

* Trang chi tiết gallery của từng tour sẽ hiển thị các hình ảnh và video tương ứng.
* Slideshow hình ảnh: Cho phép người dùng xem hình ảnh theo dạng slideshow, có thể điều hướng giữa các ảnh.
* Video: Video sẽ được nhúng trực tiếp từ YouTube và có nút phát video trên trang.
  + - 1. Sơ đồ hoạt động (Flowchart)

1. Lưu đồ tổng quan chức năng Quản lý Gallery:

Hiển thị gallery

Truy xuất dữ liệu gallery từ database

Hiển thị hình ảnh và video

Hiển thị slideshow hình ảnh

Hiển thị video nhúng Youtube

1. Mô tả chi tiết các bước:

Hiển thị gallery:

* Bước 1: Người dùng truy cập trang gallery.
* Bước 2: Hệ thống truy vấn bảng galleries để lấy tất cả hình ảnh và video liên kết với tours\_id của từng tour.
* Bước 3: Hiển thị hình ảnh dưới dạng slideshow và nhúng video từ YouTube vào trang.
* Bước 4: Người dùng có thể điều hướng giữa các ảnh và phát video trực tiếp trên trang.
  + - 1. Thuật toán giải quyết hiển thị gallery:
* Bước 1: Lấy thông tin tour
* Input: tour ID hoặc slug từ URL.
* Truy xuất thông tin liên quan đến tour đó từ database, bao gồm các đường dẫn hình ảnh và liên kết video.
* Bước 2: Truy xuất hình ảnh và video
* Hệ thống đọc các tệp hình ảnh từ thư mục dựa trên tên tour.
* Video được lấy từ database hoặc liên kết YouTube.
* Bước 3: Hiển thị slideshow
* Tạo slideshow từ các tệp hình ảnh.
* Hiển thị danh sách video kèm theo liên kết YouTube.
* Bước 4: Xử lý hiển thị
* Xác định kích thước và bố cục để các ảnh và video hiển thị đúng trên giao diện.
* Tạo điều hướng (nút trái, phải) để người dùng duyệt qua các ảnh trong slideshow.
  + 1. Chức năng quản lý Gallery
       1. Chi tiết chức năng:

1. Thông tin đầu vào

* Danh sách hình ảnh và video cho mỗi tour:

Nhiều file hình ảnh và video từ máy tính hoặc URL.

* URL hoặc liên kết YouTube cho video:

Nhập URL video từ YouTube hoặc các nền tảng khác.

* ID của tour liên kết đến gallery:

Sử dụng ID của tour để xác định gallery tương ứng.

1. Thông tin đầu ra:

* Danh sách các gallery tương ứng với từng tour:

Hiển thị tên tour và thumbnail của hình ảnh/video liên quan.

* Danh sách hình ảnh và video thuộc về từng tour:

Hiển thị chi tiết tất cả hình ảnh và video trong gallery.

* Kết quả của việc thêm hoặc xóa hình ảnh/video trong gallery:

Thông báo về việc thêm/xóa thành công hoặc thất bại.

1. Cách thức xử lý:

* Truy cập trang quản lý gallery:

Khi người dùng truy cập trang quản lý gallery của tour, hệ thống truy vấn dữ liệu từ bảng galleries bằng cách sử dụng tours\_id.

* Thêm/xóa hình ảnh/video:
* Admin gửi yêu cầu thông qua form.
* Hệ thống kiểm tra tính hợp lệ của dữ liệu:

Hình ảnh: định dạng (JPEG, PNG, GIF).

Video: kiểm tra URL.

* Cập nhật bảng galleries:

Nếu dữ liệu hợp lệ, hệ thống lưu thông tin mới vào bảng galleries.

1. Dữ liệu cần lưu trữ:

Bảng galleries: các trường dữ liệu:

* id: ID duy nhất của gallery.
* tours\_id: Khóa ngoại liên kết đến bảng tours.
* images: Lưu trữ danh sách link ảnh dưới dạng JSON.
* videos: Lưu trữ danh sách link video dưới dạng JSON.
* created\_at: Ngày tạo.
* updated\_at: Ngày cập nhật.
  + - 1. Thiết kế giao diện (UI/UX)

1. Giao diện danh sách gallery:

Hiển thị danh sách tất cả gallery tương ứng từng tour.

Hiển thị thông tin chi tiết về hình ảnh từng tour dạng thumbnail.

Hiển thị thông tin chi tiết về video dạng link Youtube

Nút "Chỉnh sửa" hoặc "Xóa" để thực hiện các hành động.

1. Giao diện thêm mới gallery:

Form nhập các trường như:

Tải lên hình ảnh/video.

Nhập URL video.

Chọn tour liên kết.

Nút "Lưu gallery" để thêm thông tin vào hệ thống.

1. Giao diện chỉnh sửa gallery:

Hiển thị danh sách hình ảnh/video hiện tại với tùy chọn chỉnh sửa.

Nút "Cập nhật" để lưu thay đổi.

* + - 1. Sơ đồ hoạt động (Flowchart)
    1. Lưu đồ tổng quan chức năng quản lý Gallery

Admin truy cập chức năng quản lý gallery

Lấy dữ liệu từ database

Hiển thị danh sách gallery

Chọn hành động: Thêm, Sửa, Xoá gallery

Sửa gallery hiện có

Thêm gallery mới

Xoá gallery

Lấy dữ liệu gallery

Điền thông tin gallery mới

Chỉnh sửa thông tin

Kiểm tra tính hợp lệ của dữ liệu

Xoá gallery khỏi database

Cập nhật thông tin gallery vào database

Lưu gallery mới vào database

Hiển thị thông báo

* + 1. Mô tả chi tiết các bước
* Thêm gallery mới:
* Admin nhập thông tin vào form.
* Hệ thống kiểm tra dữ liệu, nếu hợp lệ sẽ lưu vào bảng galleries và hiển thị thông báo thành công.
* Chỉnh sửa gallery:
* Admin chọn một gallery hiện có, hệ thống tải dữ liệu hiện tại.
* Admin cập nhật thông tin và hệ thống kiểm tra tính hợp lệ, sau đó lưu lại thông tin mới.
* Xóa gallery:
* Admin chọn gallery cần xóa.
* Hệ thống xóa gallery và hiển thị thông báo thành công.
  + - 1. Thuật toán giải quyết:

1. Thêm gallery mới:

* Bước 1: Nhận dữ liệu từ form nhập liệu.
* Bước 2: Kiểm tra các trường bắt buộc (hình ảnh, video).
* Bước 3: Nếu hợp lệ, lưu dữ liệu vào bảng galleries.

1. Chỉnh sửa gallery:

* Bước 1: Lấy dữ liệu gallery hiện tại từ database.
* Bước 2: Admin chỉnh sửa các thông tin và gửi yêu cầu cập nhật.
* Bước 3: Kiểm tra dữ liệu, nếu hợp lệ sẽ cập nhật vào bảng galleries.

1. Xóa gallery:

* Bước 1: Admin chọn gallery cần xóa.
* Bước 2: Xóa gallery khỏi bảng galleries.
  + 1. Chức năng Branches
       1. Chi tiết chức năng Branches
    2. Thông tin đầu vào:
* Tên chi nhánh: Tên của các chi nhánh (ví dụ: "Chi nhánh Hoa Kỳ").
* Địa chỉ: Địa chỉ cụ thể của mỗi chi nhánh.
* Số điện thoại: Số điện thoại liên lạc của chi nhánh.
* Vị trí địa lý: Vĩ độ và kinh độ của chi nhánh để hiển thị lên bản đồ.
  + 1. Thông tin đầu ra:
* Danh sách các chi nhánh: Hiển thị tên, địa chỉ, và số điện thoại của tất cả chi nhánh.
* Bản đồ: Hiển thị vị trí của mỗi chi nhánh trên bản đồ, với các marker cho từng vị trí.
  + 1. Cách thức xử lý:
* Lấy dữ liệu chi nhánh:
* Khi admin hoặc người dùng truy cập trang danh sách chi nhánh, hệ thống sẽ lấy danh sách từ database branches.
* Các thông tin bao gồm tên chi nhánh, địa chỉ, số điện thoại và vị trí địa lý (vĩ độ, kinh độ).
* Hiển thị bản đồ:
* Hệ thống sử dụng Maps API để hiển thị bản đồ.
* Mỗi chi nhánh sẽ được đánh dấu (marker) trên bản đồ với tọa độ vị trí địa lý lấy từ database.
  + 1. Dữ liệu cần lưu trữ:

Không cần lưu trữ dữ liệu mới.

* + - 1. Thiết kế giao diện (UI/UX):
* Giao diện danh sách chi nhánh:
* Hiển thị danh sách chi nhánh với các thông tin như tên, địa chỉ, và số điện thoại.
* Bên cạnh đó là bản đồ với các marker chỉ ra vị trí của từng chi nhánh.
* Bản đồ:
* Hiển thị bản đồ với các marker để chỉ ra vị trí của các chi nhánh.
* Có thể zoom, kéo thả để xem chi tiết vị trí.
  + - 1. Sơ đồ hoạt động (Flowchart):
    1. Lưu đồ tổng quan chức năng Branches

Hiển thị branches

Truy xuất dữ liệu branches từ database

Hiển thị thông tin branches

Hiển thị bản đồ và marker

* + 1. Mô tả chi tiết các bước:

Xem danh sách chi nhánh:

* Người dùng truy cập trang danh sách chi nhánh.
* Hệ thống lấy danh sách chi nhánh từ bảng branches và hiển thị thông tin lên trang web.
* Vị trí của các chi nhánh được hiển thị trên bản đồ thông qua Maps API.
  + - 1. Thuật toán giải quyết:

Hiển thị danh sách chi nhánh và bản đồ:

Bước 1: Lấy danh sách chi nhánh từ database.

Bước 2: Hiển thị danh sách chi nhánh và thông tin tên, địa chỉ, số điện thoại lên trang.

Bước 3: Sử dụng Maps API để hiển thị vị trí của chi nhánh trên bản đồ.

* + 1. Chức năng Quản lý Branches
       1. Chi tiết chức năng

1. Thông tin đầu vào:

* Thông tin chi nhánh:
* Tên chi nhánh: Tên của chi nhánh công ty.
* Địa chỉ: Địa chỉ cụ thể của chi nhánh.
* Số điện thoại: Liên hệ của chi nhánh.
* Thông tin định vị (positioning): Dữ liệu về tọa độ địa lý (kinh độ và vĩ độ).
* Yêu cầu của admin:
* Thêm mới, chỉnh sửa, hoặc xóa thông tin chi nhánh.

1. Thông tin đầu ra:

* Danh sách chi nhánh: Hiển thị tất cả các chi nhánh của công ty trên toàn thế giới kèm theo địa chỉ, số điện thoại và thông tin định vị.
* Kết quả thêm/chỉnh sửa/xóa: Trả về thông báo cho admin (thành công hoặc lỗi).

1. Cách thức xử lý:

* Thêm mới chi nhánh:
* Admin truy cập vào form thêm mới chi nhánh, nhập các thông tin yêu cầu.
* Hệ thống kiểm tra tính hợp lệ của thông tin (tên, địa chỉ, số điện thoại, định vị).
* Nếu hợp lệ, thông tin được lưu vào bảng branches.
* Hệ thống trả về thông báo thêm chi nhánh thành công.
* Chỉnh sửa chi nhánh:
* Admin chọn chi nhánh cần chỉnh sửa.
* Thông tin hiện tại của chi nhánh được lấy từ database và hiển thị trên form chỉnh sửa.
* Admin cập nhật thông tin cần thiết và gửi yêu cầu.
* Hệ thống kiểm tra tính hợp lệ của thông tin mới, nếu hợp lệ sẽ cập nhật vào database.
* Thông báo chỉnh sửa thành công được hiển thị.
* Xóa chi nhánh:
* Admin chọn chi nhánh cần xóa.
* Hệ thống xóa chi nhánh khỏi bảng branches.
* Thông báo xóa thành công được hiển thị.

1. Dữ liệu cần lưu trữ:

Bảng branches:

* id: ID duy nhất của chi nhánh.
* name: Tên chi nhánh.
* address: Địa chỉ chi nhánh.
* phone: Số điện thoại của chi nhánh.
* positioning: Thông tin tọa độ địa lý (chứa kinh độ và vĩ độ).
* created\_at: Ngày tạo.
* updated\_at: Ngày cập nhật.
  + - 1. Thiết kế giao diện (UI/UX):
* Giao diện danh sách chi nhánh:
* Hiển thị danh sách tất cả các chi nhánh cùng với thông tin cơ bản.
* Nút "Chỉnh sửa" hoặc "Xóa" cho từng chi nhánh.
* Giao diện thêm mới chi nhánh:
* Form chứa các trường nhập liệu (tên, địa chỉ, số điện thoại, định vị).
* Nút "Lưu chi nhánh" để lưu thông tin vào hệ thống.
* Giao diện chỉnh sửa chi nhánh:
* Hiển thị các thông tin hiện tại của chi nhánh.
* Cho phép admin chỉnh sửa thông tin và nhấn "Cập nhật".
  + - 1. Sơ đồ hoạt động (Flowchart):

1. Lưu đồ tổng quan chức năng quản lý Branches

Admin truy cập chức năng quản lý branches

Lấy dữ liệu từ database

Hiển thị danh sách branches

Chọn hành động: Thêm, Sửa, Xoá branches

Sửa branch hiện có

Thêm branch mới

Xoá branch

Lấy dữ liệu branch

Điền thông tin branch mới

Chỉnh sửa thông tin

Kiểm tra tính hợp lệ của dữ liệu

Xoá branch khỏi database

Cập nhật thông tin branch vào database

Lưu branch mới vào database

Hiển thị thông báo

1. Mô tả chi tiết các bước

* Thêm branch mới:
* Admin nhập thông tin vào form.
* Hệ thống kiểm tra dữ liệu, nếu hợp lệ sẽ lưu vào bảng branches và hiển thị thông báo thành công.
* Chỉnh sửa branch:
* Admin chọn một branch hiện có, hệ thống tải dữ liệu hiện tại.
* Admin cập nhật thông tin và hệ thống kiểm tra tính hợp lệ, sau đó lưu lại thông tin mới.
* Xóa branch:
* Admin chọn branch cần xóa.
* Hệ thống xóa branch và hiển thị thông báo thành công.
  + - 1. Thuật toán giải quyết:
* Thêm chi nhánh:
* Bước 1: Nhận dữ liệu từ form nhập.
* Bước 2: Kiểm tra các trường bắt buộc (tên, địa chỉ, số điện thoại, định vị).
* Bước 3: Nếu hợp lệ, lưu dữ liệu vào bảng branches.
* Chỉnh sửa chi nhánh:
* Bước 1: Lấy dữ liệu hiện tại từ bảng branches.
* Bước 2: Admin cập nhật thông tin và gửi yêu cầu.
* Bước 3: Kiểm tra hợp lệ, nếu đúng sẽ cập nhật dữ liệu trong bảng.
* Xóa chi nhánh:
* Bước 1: Admin chọn chi nhánh cần xóa.
* Bước 2: Hệ thống xóa chi nhánh khỏi bảng branches.
  + 1. Chức năng Information:
       1. Chi tiết chức năng:

1. Thông tin đầu vào:

* Thông tin về hoạt động leo núi và trekking:
* Lịch sử: Quá trình phát triển của leo núi và trekking qua thời gian.
* Loại hình: Các loại trekking, leo núi (thám hiểm, phiêu lưu, trekking ngắn ngày).
* Kỹ thuật: Hướng dẫn kỹ thuật cần thiết khi leo núi và trekking.
* Lưu trú: Các địa điểm lưu trú, nghỉ ngơi trong hành trình.
* Mối nguy hiểm: Các rủi ro, cảnh báo về thời tiết, sức khỏe, địa hình.
* Yêu cầu của người dùng (admin): Thêm các bài viết mới hoặc cập nhật thông tin theo danh mục cụ thể (lịch sử, loại hình, kỹ thuật, v.v.).

1. Thông tin đầu ra:

* Hiển thị danh sách thông tin: Hiển thị danh sách các bài viết theo danh mục (lịch sử, loại hình, kỹ thuật).
* Chi tiết một bài viết: Hiển thị đầy đủ nội dung của một bài viết khi người dùng yêu cầu xem chi tiết.
* Kết quả hiển thị theo danh mục: Hiển thị thông tin cụ thể theo từng chủ đề.

1. Cách thức xử lý:

* Phân loại và lưu trữ bài viết:
* Các bài viết về lịch sử, loại hình, kỹ thuật được lưu trữ dưới dạng HTML tĩnh. Mỗi bài viết có tiêu đề, nội dung và danh mục rõ ràng.
* Các bài viết được tổ chức theo cấu trúc thư mục
* Hiển thị bài viết:
* Khi người dùng chọn một danh mục (ví dụ: lịch sử), hệ thống sẽ tải và hiển thị danh sách các bài viết tương ứng với tiêu đề và tóm tắt.
* Quản lý các bài viết:
* Admin có thể thêm, cập nhật hoặc xóa nội dung của các bài viết trực tiếp trong mã HTML.

1. Dữ liệu cần lưu trữ:

* Các tệp HTML:
* title: Tiêu đề bài viết (hiển thị trên trang chủ hoặc danh mục).
* content: Nội dung bài viết (mô tả chi tiết trong từng bài viết).
* category: Danh mục của bài viết (lịch sử, loại hình, kỹ thuật, lưu trú, mối nguy hiểm).
  + - 1. Thiết kế giao diện (UI/UX)
* Giao diện thêm và chỉnh sửa bài viết:
* Sử dụng trình soạn thảo HTML để thêm hoặc chỉnh sửa nội dung.
* Tổ chức các tệp HTML theo danh mục.
* Giao diện hiển thị danh mục:
* Khi click vào từng danh mục, hiển thị nội dung bài viết tương ứng.
  + - 1. Sơ đồ hoạt động (Flowchart)

1. Lưu đồ tổng quan chức năng Information

Hiển thị danh mục thông tin

Người dùng chọn một danh mục

Hiển thị bài viết theo danh mục

1. Mô tả chi tiết các bước:

Hiển thị bài viết:

* Khi người dùng truy cập trang thông tin, hệ thống sẽ tải danh mục các bài viết theo chủ đề từ các tệp HTML.
* Người dùng có thể chọn một chủ đề cụ thể (lịch sử, kỹ thuật, lưu trú).
  + - 1. Thuật toán giải quyết
* Bước 1: Tải trang danh mục:

Khi người dùng truy cập, hệ thống sẽ hiển thị danh sách các danh mục (lịch sử, loại hình, kỹ thuật).

* Bước 2: Hiển thị bài viết theo danh mục:

Dựa vào danh mục người dùng chọn, hệ thống sẽ hiển thị các bài viết tương ứng trong thư mục của danh mục đó.

* + 1. Chức năng Quản lý tài khoản cho người dùng
       1. Chi tiết chức năng

1. Thông tin đầu vào:

* Thông tin đăng ký tài khoản:
* Tên: Tên của người dùng.
* Số điện thoại: Liên lạc và xác minh tài khoản.
* Email: Địa chỉ email để đăng nhập và nhận thông tin xác minh.
* Mật khẩu: Được mã hóa và lưu trữ bảo mật.
* Mã xác minh (verification\_code): Dùng để xác minh khi người dùng đăng ký tài khoản mới
* Thông tin cập nhật tài khoản:
* Email: dùng để đổi thông tin email của người dùng
* Số điện thoại: dùng để đổi thông tin số điện thoại của người dùng
* Mật khẩu: dùng để đổi mật khẩu hiện tại của người dùng
* Mã xác minh (verification\_code): Dùng để xác minh khi người dùng thay đổi thông tin email hoặc số điện thoại.
* Thông tin đăng nhập:
* Email.
* Mật khẩu.
* Yêu cầu khôi phục mật khẩu (nếu có):
* Email để gửi mã khôi phục.
* Mã xác minh (verification\_code): Dùng để xác minh khi khách khôi phục mật khẩu

1. Thông tin đầu ra:

* Thông tin tài khoản người dùng:

Tên, email, avatar, số điện thoại của người dùng.

* Trạng thái đăng nhập:

Thông báo đăng nhập thành công hay thất bại.

* Kết quả của các hoạt động:
* Thông báo đăng ký thành công hoặc thất bại.
* Thông báo cập nhật tài khoản thành công hoặc thất bại.
* Thông báo thay đổi mật khẩu thành công hay thất bại.
* Thông báo xác minh tài khoản:

Thông báo yêu cầu xác minh tài khoản qua email.

* Thông báo khôi phục tài khoản:

Thông báo khôi phục mật khẩu thành công hoặc thất bại.

1. Cách thức xử lý:

* Đăng ký tài khoản:
* Người dùng điền thông tin đăng ký.
* Hệ thống kiểm tra xem email và số điện thoại đã được sử dụng chưa.
* Nếu hợp lệ, hệ thống lưu thông tin người dùng vào bảng users và gửi mã xác minh qua email.
* Người dùng nhập mã xác minh, nếu mã chính xác, tài khoản sẽ được kích hoạt.
* Đăng nhập:
* Người dùng nhập email và mật khẩu.
* Hệ thống kiểm tra thông tin đăng nhập trong bảng users.
* Nếu thông tin hợp lệ, người dùng được đăng nhập vào hệ thống.
* Cập nhật tài khoản:
* Người dùng có thể cập nhật thông tin như tên, email, số điện thoại, mật khẩu.
* Người dùng nhập mã xác minh được gửi qua email sau đó khi lưu thông tin cập nhật vào database.
* Khôi phục mật khẩu:
* Người dùng yêu cầu khôi phục mật khẩu.
* Hệ thống gửi mã xác minh qua email mới.
* Người dùng nhập mã xác minh được gửi qua email và đặt mật khẩu mới.

1. Dữ liệu cần lưu trữ:

Bảng users: Lưu thông tin người dùng.

* id: ID duy nhất của người dùng.
* name: Tên người dùng.
* phone: Số điện thoại của người dùng.
* email: Email đăng nhập.
* password: Mật khẩu (mã hóa).
* avatar: Hình đại diện của người dùng.
* role: Vai trò của người dùng (admin hoặc user).
* verification\_code: Mã xác minh khi đăng ký hoặc khôi phục mật khẩu.
* status: Trạng thái tài khoản (đang hoạt động hoặc chưa kích hoạt).
* created\_at: Ngày tài khoản được tạo.
* updated\_at: Ngày tài khoản được cập nhật.
  + - 1. Thiết kế giao diện (UI/UX)

1. Giao diện đăng ký tài khoản:

Form bao gồm: Tên, số điện thoại, email, mật khẩu.

Nút "Đăng ký" để gửi thông tin.

Hiển thị thông báo lỗi nếu email hoặc số điện thoại đã tồn tại.

1. Giao diện đăng nhập:

Form nhập email và mật khẩu.

Nút "Đăng nhập" để kiểm tra thông tin.

Hiển thị thông báo thành công hay thất bại.

1. Giao diện cập nhật tài khoản:

Form cập nhật thông tin tài khoản bao gồm tên, email, số điện thoại, mật khẩu.

Nút "Lưu" để lưu thông tin cập nhật.

1. Giao diện khôi phục mật khẩu:

Form nhập email để yêu cầu khôi phục mật khẩu.

Sau khi nhận mã, người dùng sẽ nhập mã xác minh và mật khẩu mới.

* + - 1. Sơ đồ hoạt động (Flowchart)

1. Lưu đồ tổng quan chức năng Quản lý tài khoản cho người dùng:

Người dùng truy cập chức năng quản lý tài khoản

Chọn hành động: Đăng ký, Đăng nhập, Cập nhật, Khôi phục mật khẩu

Đăng nhập

Đăng ký

Khôi phục mật khẩu

Cập nhật

Kiểm tra thông tin

Kiểm tra thông tin

Xác minh qua email

Lấy dữ liệu hiện tại

Xác nhận đăng nhập

Lưu vào database

Đặt mật khẩu mới

Cập nhật thông tin

Xác minh qua email

Lưu thay đổi vào database

Hiển thị thông báo

Xác minh qua email

Kích hoạt tài khoản

1. Mô tả chi tiết các bước:

* Đăng ký tài khoản:
* Người dùng điền đầy đủ thông tin đăng ký.
* Hệ thống kiểm tra xem email và số điện thoại đã tồn tại chưa.
* Nếu hợp lệ, hệ thống lưu thông tin vào bảng users trong database và gửi mã xác minh qua email.
* Người dùng nhập mã xác minh, nếu mã đúng, tài khoản sẽ được kích hoạt.
* Đăng nhập:
* Người dùng nhập email và mật khẩu.
* Hệ thống kiểm tra xem thông tin có khớp với database không.
* Nếu đúng, hệ thống cho phép người dùng đăng nhập.
* Cập nhật tài khoản:
* Người dùng có thể cập nhật tên, email, số điện thoại, mật khẩu.
* Với trường hợp cập nhật số điện thoại, hệ thống gửi mã xác minh qua email, người dùng nhập mã xác minh, nếu mã đúng, hệ thống lưu thông tin thay đổi vào database.
* Với trường hợp cập nhật email, hệ thống gửi mã xác minh qua email mới, người dùng nhập mã xác minh, nếu mã đúng, hệ thống lưu thông tin thay đổi vào database.
* Khôi phục mật khẩu:
* Người dùng yêu cầu khôi phục mật khẩu bằng cách nhập email.
* Hệ thống gửi mã xác minh qua email và người dùng sẽ nhập mã xác minh trước khi đặt mật khẩu mới.
  + - 1. Thuật toán giải quyết:
* Đăng ký tài khoản:
* Bước 1: Nhận thông tin đăng ký từ form.
* Bước 2: Kiểm tra xem email và số điện thoại có tồn tại chưa.
* Bước 3: Nếu hợp lệ, lưu thông tin vào bảng users và gửi mã xác minh qua email.
* Bước 4: Người dùng nhập mã xác minh, nếu đúng, kích hoạt tài khoản.
* Đăng nhập:
* Bước 1: Nhận thông tin đăng nhập.
* Bước 2: Kiểm tra email và mật khẩu trong database.
* Bước 3: Nếu hợp lệ, cho phép người dùng đăng nhập.
* Cập nhật tài khoản:
* Bước 1: Nhận thông tin cập nhật từ người dùng.
* Bước 2: Gửi mã xác minh qua email.
* Bước 3: Người dùng nhập mã xác minh, nếu đúng sẽ lưu thông tin mới vào bảng users.
* Khôi phục mật khẩu:
* Bước 1: Nhận email yêu cầu khôi phục.
* Bước 2: Gửi mã xác minh qua email.
* Bước 3: Người dùng nhập mã xác minh và đặt mật khẩu mới.
  + 1. Chức năng Quản lý tài khoản cho Admin
       1. Chi tiết chức năng

1. Thông tin đầu vào:

* Thông tin người dùng:
* Tên.
* Email.
* Số điện thoại.
* Avatar.
* Mật khẩu (khi tạo mới).
* Vai trò (role): Admin hoặc Người dùng thường.

1. Thông tin đầu ra:

* Hiển thị danh sách người dùng:

Tên, email, vai trò, avatar, trạng thái tài khoản (hoạt động hoặc chưa kích hoạt).

* Chi tiết từng người dùng:
* Hiển thị thông tin cá nhân và quyền hạn.
* Các hành động: Cập nhật, Xóa.

1. Cách thức xử lý:

* Tạo người dùng:
* Admin nhập thông tin người dùng như tên, email, mật khẩu, số điện thoại, avatar và vai trò.
* Hệ thống kiểm tra xem email đã tồn tại chưa.
* Nếu hợp lệ, hệ thống lưu thông tin vào database và hiển thị người dùng trong danh sách.
* Cập nhật thông tin người dùng:
* Admin chọn một người dùng từ danh sách và cập nhật thông tin như tên, email, số điện thoại, avatar, hoặc vai trò.
* Hệ thống kiểm tra quyền truy cập của admin trước khi cho phép cập nhật.
* Lưu thông tin cập nhật vào database.
* Xóa người dùng:
* Admin chọn người dùng cần xóa từ danh sách.
* Hệ thống kiểm tra có liên kết với dữ liệu nào khác không và xóa người dùng khỏi database.
* Quản lý quyền truy cập:
* Admin có thể thay đổi vai trò của người dùng (từ người dùng thường thành admin hoặc ngược lại).
* Hệ thống kiểm tra quyền của admin trước khi cập nhật vai trò.

1. Dữ liệu cần lưu trữ:

Bảng users:

* id: ID duy nhất của người dùng.
* name: Tên người dùng.
* email: Email của người dùng.
* phone: Số điện thoại của người dùng.
* password: Mật khẩu đã được mã hóa.
* avatar: Hình đại diện của người dùng.
* role: Vai trò của người dùng (admin hoặc người dùng thường).
* status: Trạng thái tài khoản (hoạt động hoặc chưa kích hoạt).
* created\_at: Ngày tạo tài khoản.
* updated\_at: Ngày cập nhật thông tin.
  + - 1. Thiết kế giao diện (UI/UX)
* Giao diện danh sách người dùng:
* Hiển thị danh sách tất cả người dùng, với các cột như tên, email, số điện thoại, mật khẩu đã mã hoá, vai trò, avatar, trạng thái tài khoản.
* Nút "Thêm người dùng" để admin tạo mới người dùng.
* Nút "Cập nhật" hoặc "Xóa" bên cạnh mỗi người dùng.
* Giao diện chi tiết người dùng:
* Hiển thị thông tin chi tiết về người dùng: Tên, email, số điện thoại, vai trò.
* Nút "Cập nhật thông tin" để thay đổi thông tin người dùng, bao gồm cả vai trò.
* Giao diện tạo người dùng mới:

Form tạo người dùng mới, với các trường: Tên, email, số điện thoại, avatar, vai trò, và mật khẩu.

* + - 1. Sơ đồ hoạt động (Flowchart)

1. Lưu đồ tổng quan chức năng Quản lý tài khoản cho Admin:

Admin truy cập chức năng quản lý tài khoản

Lấy dữ liệu từ database

Hiển thị danh sách tài khoản

Chọn hành động: Thêm, Sửa, Xoá tài khoản

Sửa tài khoản

Thêm tài khoản mới

Xoá tài khoản

Lấy dữ liệu tài khoản

Điền thông tin tài khoản mới

Chỉnh sửa thông tin

Kiểm tra tính hợp lệ của dữ liệu

Cập nhật thông tin vào database

Lưu tài khoản mới vào database

Hiển thị thông báo

Chọn tài khoản

Kiểm tra liên kết

Xoá tài khoản

Xoá thông tin trong database

1. Mô tả chi tiết các bước:

* Tạo người dùng mới:
* Admin nhập các thông tin cần thiết như tên, email, số điện thoại, vai trò và avatar vào form.
* Hệ thống kiểm tra xem email đã tồn tại chưa.
* Nếu hợp lệ, hệ thống lưu thông tin người dùng mới vào database và hiển thị danh sách người dùng cập nhật.
* Cập nhật thông tin người dùng:
* Admin chọn một người dùng từ danh sách để cập nhật thông tin.
* Sau khi admin cập nhật, hệ thống kiểm tra tính hợp lệ của dữ liệu và lưu thay đổi vào database.
* Xóa người dùng:
* Admin chọn người dùng cần xóa.
* Hệ thống kiểm tra liên kết với các dữ liệu khác, yêu cầu xác nhận trước khi xóa người dùng khỏi database.
  + - 1. Thuật toán giải quyết:
* Tạo người dùng:
* Bước 1: Nhận thông tin người dùng từ form (tên, email, số điện thoại, mật khẩu, avatar, vai trò).
* Bước 2: Kiểm tra xem email có tồn tại trong database không.
* Bước 3: Nếu không tồn tại, lưu thông tin vào bảng users và thông báo tạo thành công.
* Cập nhật thông tin người dùng:
* Bước 1: Admin chọn người dùng cần cập nhật từ danh sách.
* Bước 2: Nhận thông tin cần cập nhật từ admin (tên, email, số điện thoại, vai trò).
* Bước 3: Kiểm tra quyền truy cập của admin trước khi lưu thông tin mới vào database.
* Xóa người dùng:
* Bước 1: Admin chọn người dùng cần xóa.
* Bước 2: Hệ thống kiểm tra liên kết với các dữ liệu khác, yêu cầu xác nhận xóa.
* Bước 3: Sau khi xác nhận, xóa thông tin người dùng khỏi bảng users.
  + 1. Chức năng Order
       1. Chi tiết chức năng

1. Thông tin đầu vào:

* Thông tin khách hàng:
* Tên khách hàng.
* Số điện thoại.
* Email.
* Ghi chú (nếu có).
* Thông tin về tour được đặt:
* ID của tour.
* Số lượng tour được đặt.
* Tổng giá trị đơn hàng.

1. Thông tin đầu ra:

* Thông tin chi tiết về đơn hàng: tên khách hàng, tour đặt, số lượng, tổng giá trị đơn hàng.
* Kết quả đặt hàng: thông báo thành công hoặc lỗi.

1. Cách thức xử lý:

* Người dùng chọn tour muốn đặt và số lượng.
* Hệ thống lấy thông tin tour và số lượng để tính tổng giá trị đơn hàng.
* Người dùng nhập thông tin cá nhân: tên, số điện thoại, email, ghi chú (nếu có).
* Hệ thống kiểm tra thông tin nhập vào, nếu hợp lệ sẽ lưu thông tin đơn hàng vào database.
* Hệ thống trả về kết quả đặt hàng và hiển thị thông báo đặt hàng thành công hoặc lỗi.

1. Dữ liệu cần lưu trữ:

Bảng orders\_tours: lưu thông tin đơn hàng.

Các trường dữ liệu cần lưu:

* id: ID của đơn hàng.
* user\_id: ID của người dùng (nếu có).
* tour\_id: ID của tour được đặt.
* name: Tên khách hàng.
* phone: Số điện thoại khách hàng.
* email: Địa chỉ email của khách hàng.
* quantity: Số lượng người tham gia tour.
* total: Tổng giá trị của đơn hàng.
* note: Ghi chú của khách hàng.
* created\_at: Ngày tạo đơn hàng.
* updated\_at: Ngày cập nhật đơn hàng.
  + - 1. Thiết kế giao diện (UI/UX)
* Giao diện đặt hàng:
* Form nhập thông tin: tên, số điện thoại, email, ghi chú.
* Hiển thị thông tin tour và giá trị đơn hàng.
* Nút "Đặt hàng" để gửi thông tin đơn hàng.
* Thông tin đơn hàng

Hiển thi các thông tin chi tiết để khách hàng kiểm tra lại đơn hàng, bao gồm: tên, số điện thoại, email, tour, số người, tổng giá trị đơn hàng, ghi chú

* + - 1. Sơ đồ hoạt động (Flowchart)

1. Lưu đồ tổng quan chức năng Order:

Hiển thị form order

Nhập thông tin khách hàng

Chọn tour và số lượng người tham gia

Thông tin hợp lệ

Thông tin không hợp lệ

Kiểm tra thông tin đặt hàng

Lưu order vào database

Hiển thị thông báo

1. Mô tả chi tiết các bước:
   * Nhập thông tin cá nhân: Người dùng nhập các thông tin cần thiết như tên, số điện thoại, email, và ghi chú (nếu có).
   * Chọn tour và số lượng: Người dùng chọn tour từ danh sách có sẵn và nhập số lượng người tham gia.
   * Kiểm tra thông tin: Hệ thống sẽ kiểm tra tính hợp lệ của thông tin đầu vào như số lượng tour, email, và số điện thoại.
   * Lưu đơn hàng: Nếu thông tin hợp lệ, hệ thống sẽ lưu đơn hàng vào database.
   * Thông báo kết quả: Hệ thống hiển thị thông báo thành công nếu đơn hàng được đặt thành công, hoặc báo lỗi nếu có vấn đề.
     + 1. Thuật toán giải quyết:
   * Bước 1: Nhận thông tin đầu vào (tour, số lượng, thông tin cá nhân).
   * Bước 2: Kiểm tra tính hợp lệ của các trường thông tin, như tên không rỗng, số điện thoại đúng định dạng, email hợp lệ.
   * Bước 3: Nếu thông tin hợp lệ, tính tổng giá trị đơn hàng dựa trên số lượng người tham gia và giá tour.
   * Bước 4: Lưu thông tin đơn hàng vào bảng orders\_tours.
   * Bước 5: Trả về kết quả đặt hàng cho người dùng (thành công hoặc lỗi).
     1. Chức năng quản lý order cho Admin
        1. Chi tiết chức năng
2. Thông tin đầu vào:

* Thông tin người đặt hàng: user\_id (ID người dùng), tên, số điện thoại, email, ghi chú.
* Thông tin tour được đặt: tour\_id (ID tour), số lượng người tham gia.
* Ghi chú từ khách hàng (nếu có).

1. Thông tin đầu ra:

* Danh sách các đơn hàng đã đặt: hiển thị toàn bộ đơn hàng với thông tin cơ bản (tên khách hàng, tour, số lượng, tổng giá trị).
* Chi tiết từng đơn hàng: hiển thị đầy đủ thông tin về người đặt, tour, số lượng người tham gia, tổng giá trị, và ghi chú.

1. Cách thức xử lý:

* Admin có thể xem danh sách các đơn hàng hiện có.
* Khi khách hàng đặt hàng, hệ thống sẽ tự động tính tổng số tiền dựa trên số lượng và giá tour.
* Đơn hàng sẽ được lưu lại trong hệ thống với thông tin khách hàng và tour đã đặt.
* Admin có thể cập nhật, hủy, hoặc xem chi tiết từng đơn hàng.

1. Dữ liệu cần lưu trữ:

Bảng orders:

* id: ID của đơn hàng.
* user\_id: ID của người dùng.
* tour\_id: ID của tour được đặt.
* quantity: Số lượng người tham gia.
* total: Tổng giá trị đơn hàng.
* note: Ghi chú của khách hàng.
* created\_at: Ngày tạo đơn hàng.
* updated\_at: Ngày cập nhật đơn hàng.
  + - 1. Thiết kế giao diện (UI/UX)

Giao diện danh sách đơn hàng:

* Hiển thị danh sách tất cả các đơn hàng với các thông tin: tên khách hàng, số điện thoại, email, tour, số lượng người, tổng giá trị.
* Các hành động: cập nhật thông tin đơn hàng, hủy đơn hàng.
  + - 1. Sơ đồ hoạt động (Flowchart)

1. Lưu đồ tổng quan chức năng:

Khách đặt hàng

Tính tổng giá trị đơn hàng

Cập nhật thông tin đơn hàng

Huỷ đơn hàng

Kiểm tra tính hợp lệ dữ liệu đầu vào

Lưu thông tin đơn hàng vào database

Hiển thị danh sách đơn hàng

Cập nhật database

1. Mô tả chi tiết các bước:

* Khách đặt hàng: Khi khách hàng hoàn tất việc đặt tour, hệ thống sẽ tính tổng giá trị đơn hàng, kiểm tra tính hợp lệ dữ liệu đầu vào, ghi nhận đơn hàng và lưu vào database.
* Admin nhận đơn hàng: Admin có thể truy cập vào hệ thống và xem danh sách các đơn hàng mới.
* Cập nhật đơn hàng: Admin có thể thay đổi các thông tin của đơn hàng (tên khách hàng, số điện thoại, email, tour, số lượng người).
* Hủy đơn hàng: Nếu cần thiết, Admin có thể hủy đơn
  + - 1. Thuật toán giải quyết:
* Bước 1: Nhận thông tin đầu vào từ người dùng (tour được đặt, số lượng người, thông tin cá nhân).
* Bước 2: Tính tổng giá trị đơn hàng dựa trên số lượng và giá của tour.
* Bước 3: Kiểm tra tính hợp lệ dữ liệu đầu vào và lưu thông tin đơn hàng vào bảng orders.
* Bước 4: Admin xem danh sách đơn hàng từ database.
* Bước 6: Admin có thể thay đổi thông tin đơn hàng hoặc hủy đơn hàng khi cần.
* Bước 7: Hệ thống lưu lại các thay đổi của admin và hiển thị danh sách đơn hàng mới sau khi cập nhật.

1. Validation Checklists

|  |  |  |
| --- | --- | --- |
| **Table** | **Column** | **Validation** |
| **users** | name | * Required * Chuỗi (string), độ dài tối đa 255 ký tự. |
| phone | * Required * Unique * Chuỗi (string), định dạng số điện thoại hợp lệ. |
| email | * Required * Unique * Chuỗi (string), định dạng email hợp lệ. |
| password | * Required * Chuỗi (string), độ dài tối thiểu 6 ký tự |
| avatar | * Nullable * Kiểm tra nếu là file ảnh (image file). |
| role | * Required * Boolean, chỉ nhận giá trị true/false |
| verification\_code | * Required * Số nguyên (integer) |
| status | * Required * Boolean, chỉ nhận giá trị true/false |
| **tours** | name | * Required * Chuỗi (string), độ dài tối đa 255 ký tự. |
| price | * Required * Số nguyên (integer), phải là giá trị dương |
| image | * Nullable * Kiểm tra nếu là file ảnh (image file). |
| location | * Required * Kiểm tra định dạng chuỗi (string) |
| features | * Required * Kiểm tra định dạng chuỗi (string) |
| besttime | * Required * Kiểm tra định dạng chuỗi (string) |
| directions | * Required * Kiểm tra định dạng chuỗi (string) |
| trekkingroutes | * Required * Kiểm tra định dạng chuỗi (string) |
| items | * Required * Kiểm tra định dạng chuỗi (string) |
| cautions | * Required * Kiểm tra định dạng chuỗi (string) |
| requirements | * Required * Kiểm tra định dạng chuỗi (string) |
| **orders\_tours** | user\_id | * Nullable, kiểm tra nếu người dùng chưa đăng nhập * Khoá ngoại tham chiếu tới bảng users |
| tour\_id | * Required * Khoá ngoại tham chiếu tới bảng tours |
| name | * Required * Chuỗi (string), độ dài tối đa 255 ký tự |
| phone | * Required * Chuỗi (string), định dạng số điện thoại hợp lệ. |
| email | * Required * Chuỗi (string), định dạng email hợp lệ. |
| quantity | * Required * Số nguyên (integer), phải lớn hơn 0 |
| total | * Required * Số nguyên (integer), giá trị dương, được tính tự động dựa trên quantity và giá tour |
| note | * Nullable * Chuỗi (string), độ dài tối đa không giới hạn |
| **branches** | name | * Required * Chuỗi (string), độ dài tối đa 255 ký tự. |
| address | * Required * Chuỗi (string) |
| phone | * Required * Chuỗi (string), định dạng số điện thoại hợp lệ. |
| positioning | * Required * Định dạng toạ độ hợp lệ (lat, long) |
| **galleries** | tours\_id | * Required * Khoá ngoại tham chiếu tới bảng tours |
| images | * Nullable * Kiểm tra định dạng JSON hợp lệ * Nội dung JSON chứa URL ảnh hợp lệ |
| videos | * Nullable * Kiểm tra định dạng JSON hợp lệ * Nội dung JSON chứa URL video hợp lệ |

1. Task sheet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Start time** | **Finish time** | **Implemented by** | **Evaluation (%)** |
| **Feasibility study phase** |  |  |  |  |
| Xác định mục tiêu tổng quan của dự án | 21/08/2024 | 23/08/2024 | Hieu Nguyen Van | 25% |
| Xác định mục tiêu tổng quan của dự án | 21/08/2024 | 23/08/2024 | Quan Luu Minh | 25% |
| Xác định mục tiêu tổng quan của dự án | 21/08/2024 | 23/08/2024 | Manh Nguyen Duc | 25% |
| Xác định mục tiêu tổng quan của dự án | 21/08/2024 | 23/08/2024 | Hau Nguyen Van | 25% |
| Đánh giá nguồn lực nhân sự | 21/08/2024 | 23/08/2024 | Hieu Nguyen Van | 100% |
| Xác định các công nghệ và framwork có thể sử dụng | 21/08/2024 | 23/08/2024 | Hieu Nguyen Van | 25% |
| Xác định các công nghệ và framwork có thể sử dụng | 21/08/2024 | 23/08/2024 | Quan Luu Minh | 75% |
| Phân tích về việc tích hợp với các công nghệ bên ngoài | 21/08/2024 | 23/08/2024 | Hieu Nguyen Van | 100% |
| **Requirement Analysis phase** |  |  |  |  |
| Xác định rõ các chức năng, tính năng cần có trong hệ thống | 21/08/2024 | 23/08/2024 | Hieu Nguyen Van | 50% |
| Xác định rõ các chức năng, tính năng cần có trong hệ thống | 21/08/2024 | 23/08/2024 | Quan Luu Minh | 50% |
| Phân tích yêu cầu về cơ sở dữ liệu, các bảng cần thiết | 21/08/2024 | 23/08/2024 | Hieu Nguyen Van | 50% |
| Phân tích yêu cầu về cơ sở dữ liệu, các bảng cần thiết | 21/08/2024 | 23/08/2024 | Quan Luu Minh | 50% |
| **Design phase** |  |  |  |  |
| Thiết kế kiến trúc cơ sở dữ liệu | 24/08/2024 | 30/08/2024 | Hieu Nguyen Van | 50% |
| Thiết kế kiến trúc cơ sở dữ liệu | 24/08/2024 | 30/08/2024 | Quan Luu Minh | 50% |
| Thiết kế kiến trúc hệ thống | 24/08/2024 | 30/08/2024 | Manh Nguyen Duc | 50% |
| Thiết kế kiến trúc hệ thống | 24/08/2024 | 30/08/2024 | Hau Nguyen Van | 50% |
| Tạo sơ đồ ERD | 21/08/2024 | 23/08/2024 | Hieu Nguyen Van | 50% |
| Tạo sơ đồ ERD | 21/08/2024 | 23/08/2024 | Quan Luu Minh | 50% |
| Thiết kế sơ đồ hoạt động cho các chức năng chính | 24/08/2024 | 30/08/2024 | Manh Nguyen Duc | 50% |
| Thiết kế sơ đồ hoạt động cho các chức năng chính | 24/08/2024 | 30/08/2024 | Hau Nguyen Van | 50% |
| Thiết kế UI/UX mockup cho các trang chính | 24/08/2024 | 30/08/2024 | Hieu Nguyen Van | 25% |
| Thiết kế UI/UX mockup cho các trang chính | 24/08/2024 | 30/08/2024 | Quan Luu Minh | 25% |
| Thiết kế UI/UX mockup cho các trang chính | 24/08/2024 | 30/08/2024 | Manh Nguyen Duc | 25% |
| Thiết kế UI/UX mockup cho các trang chính | 24/08/2024 | 30/08/2024 | Hau Nguyen Van | 25% |
| Thiết kế API nếu có tích hợp với các dịch vụ bên ngoài | 24/08/2024 | 30/08/2024 | Hieu Nguyen Van | 100% |
| **Development phase** |  |  |  |  |
| Xây dựng cơ sở dữ liệu và các bảng dựa trên migrations đã thiết kế | 31/08/2024 | 05/09/2024 | Hieu Nguyen Van | 25% |
| Xây dựng cơ sở dữ liệu và các bảng dựa trên migrations đã thiết kế | 31/08/2024 | 05/09/2024 | Quan Luu Minh | 25% |
| Xây dựng cơ sở dữ liệu và các bảng dựa trên migrations đã thiết kế | 31/08/2024 | 05/09/2024 | Manh Nguyen Duc | 25% |
| Xây dựng cơ sở dữ liệu và các bảng dựa trên migrations đã thiết kế | 31/08/2024 | 05/09/2024 | Hau Nguyen Van | 25% |
| Phát triển các chức năng cơ bản | 31/08/2024 | 23/09/2024 | Hieu Nguyen Van | 25% |
| Phát triển các chức năng cơ bản | 31/08/2024 | 23/09/2024 | Quan Luu Minh | 25% |
| Phát triển các chức năng cơ bản | 31/08/2024 | 23/09/2024 | Manh Nguyen Duc | 25% |
| Phát triển các chức năng cơ bản | 31/08/2024 | 23/09/2024 | Hau Nguyen Van | 25% |
| Phát triển hệ thống xác thực người dùng | 31/08/2024 | 23/09/2024 | Manh Nguyen Duc | 90% |
| Phát triển hệ thống xác thực người dùng | 31/08/2024 | 23/09/2024 | Quan Luu Minh | 10% |
| Xây dựng chức năng đặt hàng, tính toán tổng số tiền và lưu vào cơ sở dữ liệu | 31/08/2024 | 23/09/2024 | Hau Nguyen Van | 100% |
| Phát triển phần quản lý gallery, branches | 31/08/2024 | 15/09/2024 | Hieu Nguyen Van | 100% |
| Tích hợp Maps API cho việc hiển thị chi nhánh | 31/08/2024 | 15/09/2024 | Hieu Nguyen Van | 100% |
| Viết các seeder cho database để thêm dữ liệu mẫu | 31/08/2024 | 05/09/2024 | Hieu Nguyen Van | 25% |
| Viết các seeder cho database để thêm dữ liệu mẫu | 31/08/2024 | 05/09/2024 | Quan Luu Minh | 25% |
| Viết các seeder cho database để thêm dữ liệu mẫu | 31/08/2024 | 05/09/2024 | Manh Nguyen Duc | 25% |
| Viết các seeder cho database để thêm dữ liệu mẫu | 31/08/2024 | 05/09/2024 | Hau Nguyen Van | 25% |
| Tối ưu hóa frontend với Tailwind CSS cho giao diện người dùng | 31/08/2024 | 24/09/2024 | Quan Luu Minh | 25% |
| Tối ưu hóa frontend với Tailwind CSS cho giao diện người dùng | 31/08/2024 | 24/09/2024 | Manh Nguyen Duc | 75% |
| **Testing phase** |  |  |  |  |
| Kiểm thử chức năng cho các tính năng chính: quản lý tours, orders, users | 06/09/2024 | 25/09/2024 | Hieu Nguyen Van | 10% |
| Kiểm thử chức năng cho các tính năng chính: quản lý tours, orders, users | 06/09/2024 | 25/09/2024 | Quan Luu Minh | 30% |
| Kiểm thử chức năng cho các tính năng chính: quản lý tours, orders, users | 06/09/2024 | 25/09/2024 | Manh Nguyen Duc | 30% |
| Kiểm thử chức năng cho các tính năng chính: quản lý tours, orders, users | 06/09/2024 | 25/09/2024 | Hau Nguyen Van | 30% |
| Kiểm thử quy trình đặt hàng | 06/09/2024 | 25/09/2024 | Hau Nguyen Van | 100% |
| Kiểm thử các API tích hợp | 06/09/2024 | 25/09/2024 | Hieu Nguyen Van | 100% |
| Kiểm thử giao diện người dùng trên nhiều trình duyệt và thiết bị | 06/09/2024 | 25/09/2024 | Manh Nguyen Duc | 100% |
| Fix các lỗi phát hiện trong quá trình kiểm thử | 06/09/2024 | 25/09/2024 | Hieu Nguyen Van | 25% |
| Fix các lỗi phát hiện trong quá trình kiểm thử | 06/09/2024 | 25/09/2024 | Quan Luu Minh | 25% |
| Fix các lỗi phát hiện trong quá trình kiểm thử | 06/09/2024 | 25/09/2024 | Manh Nguyen Duc | 25% |
| Fix các lỗi phát hiện trong quá trình kiểm thử | 06/09/2024 | 25/09/2024 | Hau Nguyen Van | 25% |