**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC**

**CÔNG NGHỆ THÔNG TIN**

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**Độc Lập – Tự Do – Hạnh Phúc**

A blue logo with a flower and a planet

Description automatically generated

**BÁO CÁO ĐỒ ÁN MÔN HỌC**

**CÔNG NGHỆ LẬP TRÌNH ĐA NỀN TẢNG**

**CHO ỨNG DỤNG DI ĐỘNG**

**Đề tài:** Ứng dụng đặt chỗ nhà hàng

**MỤC LỤC**

[**1.** **GIỚI THIỆU ĐỀ TÀI** 5](#_Toc186376245)

[**1.1.** **Lý do chọn đề tài.** 5](#_Toc186376246)

[**1.2.** **Phạm vi đề tài.** 5](#_Toc186376247)

[**1.3.** **Đối tượng đề tài.** 5](#_Toc186376248)

[**2.** **TỔNG QUAN DỰ ÁN LIÊN QUAN.** 5](#_Toc186376249)

[**2.1.** **Pasgo.** 5](#_Toc186376250)

[**2.2.** **Golden Spoons.** 7](#_Toc186376251)

[**3.** **NỘI DUNG DỰ KIẾN THỰC HIỆN.** 8](#_Toc186376252)

[**3.1.** **Khảo sát nhu cầu người dùng.** 8](#_Toc186376253)

[**3.2.** **Liệt kê tính năng.** 8](#_Toc186376254)

[**3.2.1.** **Trang chủ.** 8](#_Toc186376255)

[**3.2.2.** **Hiển thị nhà hàng.** 9](#_Toc186376256)

[**3.2.3.** **Tìm kiếm.** 9](#_Toc186376257)

[**3.2.4.** **Đặt bàn và theo dõi tình trạng.** 10](#_Toc186376258)

[**3.2.5.** **Chỉ đường và gợi ý.** 10](#_Toc186376259)

[**3.2.6.** **Trò chuyện.** 11](#_Toc186376260)

[**3.2.7.** **Quản lý nhà hàng.** 11](#_Toc186376261)

[**3.2.8.** **Quản lý đặt chỗ và theo dõi tình trạng.** 12](#_Toc186376262)

[**3.2.9.** **Thống kê và báo cáo.** 12](#_Toc186376263)

[**3.2.10.** **Tạo tài khoản và xác nhận tài khoản.** 13](#_Toc186376264)

[**3.2.11.** **Đăng nhập.** 13](#_Toc186376265)

[**3.2.12.** **Đăng ký.** 14](#_Toc186376266)

[**3.3.** **Công nghệ và kỹ thuật áp dụng.** 14](#_Toc186376267)

[**3.3.1.** **Công nghệ sử dụng tại front-end.** 14](#_Toc186376268)

[**3.3.2.** **Công nghệ sử dụng tại back-end.** 15](#_Toc186376269)

[**3.4.** **Công nghệ xây dựng database.** 15](#_Toc186376270)

[**4.** **PHƯƠNG PHÁP THỰC HIỆN** 16](#_Toc186376271)

[**4.1.** **Khảo sát trải nghiệm người dùng (UX/UI):** 16](#_Toc186376272)

[**4.2.** **Phân tích yêu cầu hệ thống** 17](#_Toc186376273)

[**4.2.1.** **Ánh xạ dữ liệu từ mô hình hướng đối tượng (ERD).** 17](#_Toc186376274)

[**4.2.2.** **Danh sách các Use Cases.** 17](#_Toc186376275)

[**4.2.3.** **Sơ đồ lớp và đặc tả chức năng.** 17](#_Toc186376276)

[**4.3.** **Thiết kế** 18](#_Toc186376277)

[**4.3.1.** **Thiết kế giao diện người dùng (UI/UX)** 18](#_Toc186376278)

[**4.3.2.** **Thiết kế dữ liệu (Database)** 18](#_Toc186376279)

[**4.4.** **Xây dựng ứng dụng** 19](#_Toc186376280)

[**4.4.1.** **Xây dựng giao diện người dùng (Frontend)** 19](#_Toc186376281)

[**4.4.2.** **Xây dựng Backend:** 20](#_Toc186376282)

[**4.5.** **Kiểm thử Vận Hành và Bảo trì.** 21](#_Toc186376283)

[**5.** **TỔNG KẾT.** 21](#_Toc186376284)

[**5.1.** **Kết quả dự kiến đạt được.** 21](#_Toc186376285)

[**5.2.** **Ưu điểm.** 22](#_Toc186376286)

[**5.3.** **Nhược điểm.** 23](#_Toc186376287)

[**5.4.** **Hướng phát triển.** 23](#_Toc186376288)

[**TÀI LIỆU THAM KHẢO.** 2](#_Toc186376289)

**MỤC LỤC HÌNH ẢNH.**

[Hình 1: Giao diện ứng dụng Pasgo. 7](#_Toc186374597)

[Hình 2:Giao diện ứng dụng Golden SpoonS. 8](#_Toc186374598)

[Hình 3:Giao diện trang chủ. 9](#_Toc186374599)

[Hình 4: Hiển thị nhà hàng. 9](#_Toc186374600)

[Hình 5: Giao diện tìm kiếm. 10](#_Toc186374601)

[Hình 6: Giao diện đặt bàn và theo dõi tình trạng. 10](#_Toc186374602)

[Hình 7: Giao diện chỉ đường và gợi ý. 11](#_Toc186374603)

[Hình 8: Giao diện trò chuyện và threads. 11](#_Toc186374604)

[Hình 9: Giao diện admin quản lý nhà hàng. 12](#_Toc186374605)

[Hình 10: Giao diện admin quản lý đặt chỗ và theo dõi tình trạng. 12](#_Toc186374606)

[Hình 11: Giao diện admin xem thống kê và báo cáo. 13](#_Toc186374607)

[Hình 12: Giao diện admin quản lý các tài khoản. 13](#_Toc186374608)

[Hình 13: Giao diện đăng nhập. 14](#_Toc186374609)

[Hình 14: Giao diện đăng ký. 14](#_Toc186374610)

[Hình 15: ERD của dự án. 18](#_Toc186374611)

[Hình 16: Sitemap cho user và admin. 20](#_Toc186374612)

# **GIỚI THIỆU ĐỀ TÀI**

## **Lý do chọn đề tài.**

Trong xã hội hiện nay, nhu cầu tìm kiếm và thưởng thức ẩm thực tại các nhà hàng đang ngày càng tăng cao, không chỉ có người trẻ mà những người trung niên hay người cao tuổi cũng khá quan tâm đến những ứng dụng có thể đặt chỗ tại các nhà hàng.

Trải nghiệm người dùng được nâng cao: Ứng dụng này có thể giúp người dùng chủ động và tiết kiệm thời gian hơn trong việc tìm kiếm và đặt chỗ.

Cơ hội tiếp thị và xây dựng thương hiệu: Việc các nhà hàng xuất hiện trên ứng dụng này cũng là cách giới thiệu nhà hàng của chủ sở hữu tới được người dùng, nâng cao độ nhận diện cho nhà hàng trong quá trình kinh doanh.

## **Phạm vi đề tài.**

Phạm vi giới hạn về địa bàn của đề tài này: Phạm vi là các nhà hàng nằm rải rác trên khu vực thành phố Hồ Chí Minh

Giới hạn thời gian nghiên cứu: từ tháng 9 năm 2024 đến tháng 11 năm 2024 nhằm có thời gian thu thập dữ liệu và hoàn thành giao diện cũng như các chức năng cho ứng dụng.

## **Đối tượng đề tài.**

Đối tượng là những người bận rộn không có thời gian để chủ động tìm kiếm trực tiếp từng nơi để ăn uống. Thay vào đó họ có thể sử dụng ứng dụng vừa tiết kiệm thời gian và có nhiều lựa chọn hơn phù hợp với quỷ thời gian của họ.

# **TỔNG QUAN DỰ ÁN LIÊN QUAN.**

Ngoài việc khảo sát người dùng thì nhóm cũng tiến hành khảo sát những ứng dụng tìm kiếm và đặt khách sạn nổi tiếng như Pasgo, Golden Spools …

## **Pasgo.**

Là một ứng dụng điện thoại chuyên về đặt bàn tại các nhà hàng ở Việt Nam. Ứng dụng giúp người dùng dễ dàng tìm kiếm nhà hàng, xem thực đơn, ưu đãi và đặt bàn trước để tránh tình trạng hết chỗ, đặc biệt là tại các nhà hàng nổi tiếng.

**Ưu điểm:**

* Bộ lọc hỗ trợ chi tiết với đa dạng tiêu chí cho người dùng lựa chọn ( Giá cả, Quốc gia, Vùng miền, Phong cách, Kiểu phục vụ, Số lượng người, Thời gian, …)
* Có các biểu tượng đại diện cho từng loại thức ăn như Lẩu, Nướng, Buffet, Hải sản….
* Tích hợp thông tin chi tiết: Thông tin rõ ràng về từng lựa chọn, từ hình ảnh, mô tả đến đánh giá từ người dùng khác, lượt người dùng ứng dụng đã ghé, giúp người dùng đưa ra quyết định nhanh chóng.
* Định vị và giới thiệu người dùng các nhà hàng có ưu đãi hot ở các khu vực gần đó.
* Dễ dàng tìm kiếm nhà hàng phù hợp khi người dùng chỉ cần tìm món ăn mình muốn và ứng dụng sẽ giới thiệu người dùng các nhà hàng gần người dùng có món ăn đó.
* Hỗ trợ khách hàng 24/7: Cung cấp dịch vụ hỗ trợ mọi lúc, giúp xử lý nhanh chóng các vấn đề phát sinh khi đặt vé hoặc khách sạn.

**Nhược điểm:**

* Giao diện không quá đầu tư: Mặc dù cung cấp nhiều tùy chọn, một số người dùng có thể cảm thấy giao diện hơi rối mắt với quá nhiều chữ, các tiêu đề và đề mục không có font chữ hay cỡ chữ khác với nội dung mà chỉ in đậm, không có viền bao quanh từng bài viết giới thiệu nhà hàng, gây khó khăn trong việc tiếp cận, thao tác và tìm kiếm thông tin chính xác.
* Quá tải thông tin quảng cáo và ưu đãi: Quá nhiều pop-up về khuyến mãi có thể làm gián đoạn trải nghiệm người dùng, đặc biệt khi chỉ muốn tìm kiếm thông tin đơn giản mà không cần các ưu đãi đi kèm..
* Thiếu tính năng so sánh dễ nhìn: Khi muốn so sánh nhiều tùy chọn như khách sạn hoặc chuyến bay, ứng dụng không có chế độ xem hoặc so sánh tiện lợi, khiến người dùng phải mở từng mục riêng lẻ để đối chiếu.

A screenshot of a menu

Description automatically generated

Hình : Giao diện ứng dụng Pasgo.

## **Golden Spoons.**

Là một ứng dụng điện thoại chuyên về đặt bàn và đi chợ online tại các nhà hàng ở Việt Nam. Ứng dụng giúp người dùng dễ dàng tìm kiếm nhà hàng, xem thực đơn, ưu đãi và đặt bàn trước để tránh tình trạng hết chỗ, đặc biệt là tại các nhà hàng nổi tiếng

**Ưu điểm:**

* Màu sắc hài hòa tươi mới.
* Giao diện tối giản dễ sử dụng.
* Nổi bật các ưu đãi flash sale ở trang đầu.
* Hỗ trợ tìm nhà hàng trong khu vực.

**Nhược điểm:**

* Không có bộ lọc cho người dùng tìm kiếm theo nhu cầu.
* Không thể tìm nhà hàng dựa vào món ăn.
* Chỉ có thể tìm kiếm khi bạn biết nhà hàng bạn muốn tìm.

A screen shot of a cell phone

Description automatically generated

Hình :Giao diện ứng dụng Golden SpoonS.

# **NỘI DUNG DỰ KIẾN THỰC HIỆN.**

## **Khảo sát nhu cầu người dùng.**

Nhằm giúp ứng dụng tiếp cận người dùng một cách nhanh chóng, hiệu quả, thì việc khảo sát ý kiến người dùng về giao diện và tính năng mong muốn là một điều rất cần thiết. Việc khảo sát sẽ được tiến hành với nhóm người dùng là những người thường xuyên sử dụng các ứng dụng đặt chỗ nhà hàng thông qua khảo sát trực tiếp và khảo sát trực tuyến bằng Google Forms.

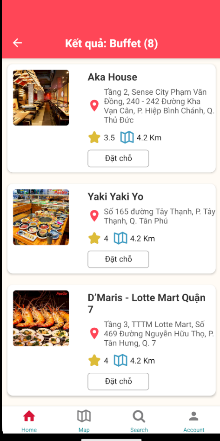
* 1. **Liệt kê tính năng.**
     1. **Trang chủ.**
* Nơi chứa các chức năng cơ bản của ứng dụng này cũng như thông tin của các nhà hàng.
* Từ trang chủ người dùng có thể khám phá nhiều tính năng hơn của ứng dụng này

**A screenshot of a food menu

Description automatically generated**

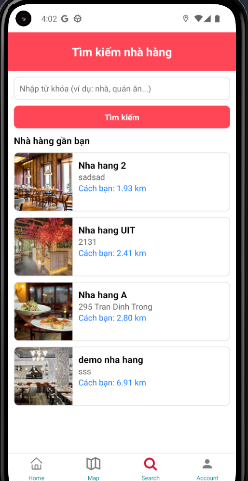
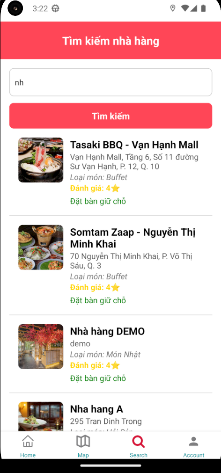
Hình :Giao diện trang chủ.

* + 1. **Hiển thị nhà hàng.**
* Ứng dụng hiển thị danh sách các nhà hàng với giao diện đẹp, dễ sử dụng và thân thiện.
* Trang chi tiết cung cấp thông tin đầy đủ như: tên nhà hàng, địa chỉ, giờ mở cửa, hình ảnh, đánh giá và thực đơn, đánh giá sao.

****

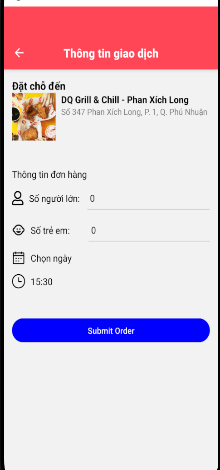
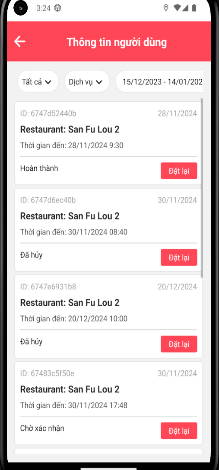
Hình : Hiển thị nhà hàng.

* + 1. **Tìm kiếm.**
* Ứng dụng đặt chỗ này có chức năng tìm kiếm các nhà hàng theo tên, địa chỉ mà bạn mong muốn.
* Có tích hợp thêm bộ lọc để hỗ trợ tìm kiếm (tìm nhà hàng theo khoảng cách gần xa, hay món ăn bạn thích).
* Có thể tìm kiếm các nhà hàng ở xung quanh mình.



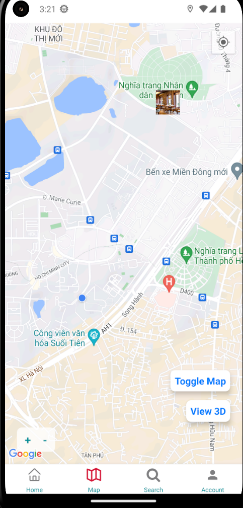
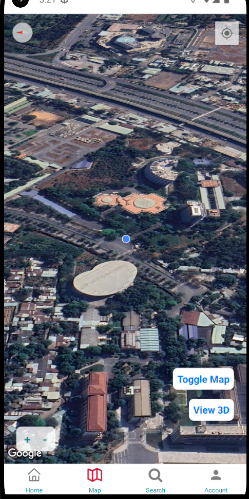
Hình : Giao diện tìm kiếm.

* + 1. **Đặt bàn và theo dõi tình trạng.**
* Ứng dụng hỗ trợ đặt chỗ trước với thông tin chi tiết về số lượng khách, thời gian và tình trạng chỗ trống.
* Ứng dụng còn cho người dùng xem các trạng thái đặt bàn (Hủy, Thành Công, Đang Duyệt).



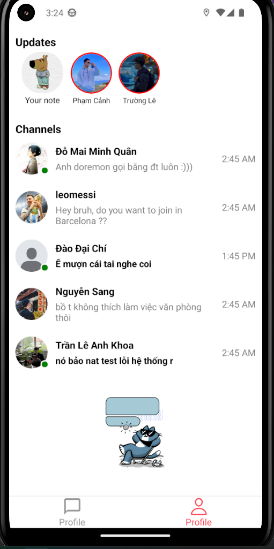
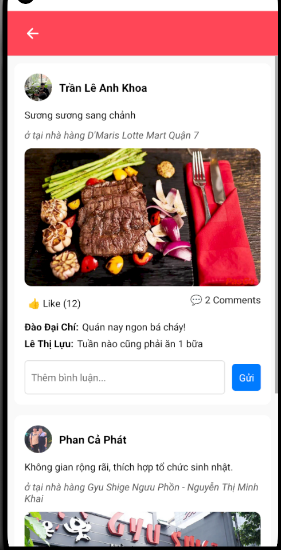
Hình : Giao diện đặt bàn và theo dõi tình trạng.

* + 1. **Chỉ đường và gợi ý.**
* Ứng dụng có 1 bản đồ với 2 chế độ giúp người dùng dễ dàng tìm kiếm nhà hàng lân cận
* Kết nối với Google map để chỉ đường và tính khoảng cách di chuyển.



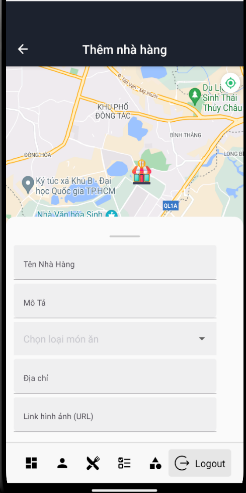
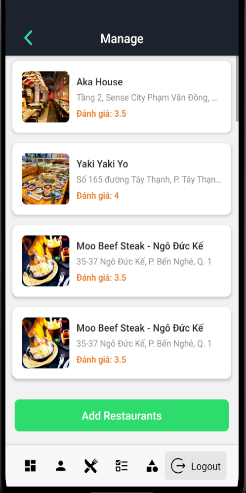
Hình : Giao diện chỉ đường và gợi ý.

* + 1. **Trò chuyện.**
* Ứng dụng giúp người dùng phản hồi về chất lượng của nhà hàng hoặc yêu cầu hỗ trợ với admin.
* Cho phép bạn bè đăng ảnh review nhà hàng.



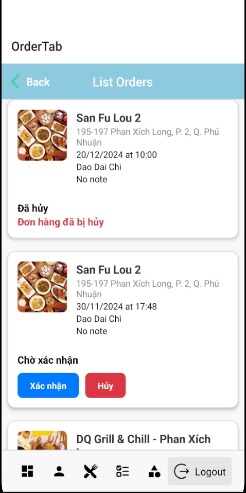
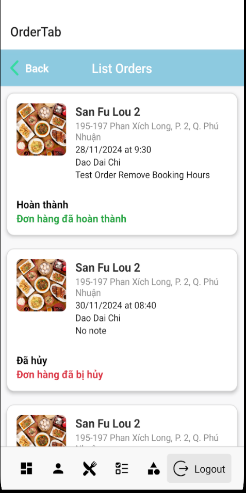
Hình : Giao diện trò chuyện và threads.

* + 1. **Quản lý nhà hàng.**
* Ứng dụng giúp admin có thể dễ dàng thêm, xóa hoặc thay đổi thông tin của nhà hàng như địa chỉ, giờ mở cửa, hình ảnh, thực đơn.
* Cập nhật trạng thái hoạt động (mở cửa/đóng cửa).



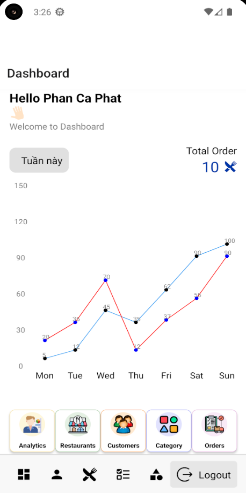
Hình : Giao diện admin quản lý nhà hàng.

* + 1. **Quản lý đặt chỗ và theo dõi tình trạng.**
* Theo dõi và quản lý các yêu cầu đặt chỗ của khách hàng.
* Xác nhận, từ chối hoặc thay đổi trạng thái đơn đặt bàn/món.



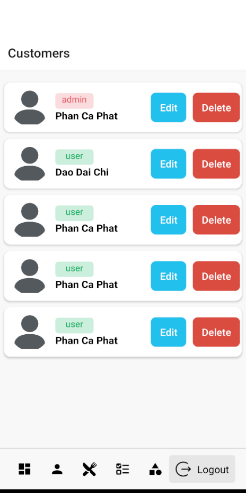
Hình : Giao diện admin quản lý đặt chỗ và theo dõi tình trạng.

* + 1. **Thống kê và báo cáo.**
* Có biểu đồ đường giúp admin dễ dàng theo dõi xem số liệu đặt bàn, đơn hàng, và doanh thu theo ngày, tuần, tháng.
* Theo dõi xu hướng khách hàng (món ăn yêu thích, thời gian cao điểm).



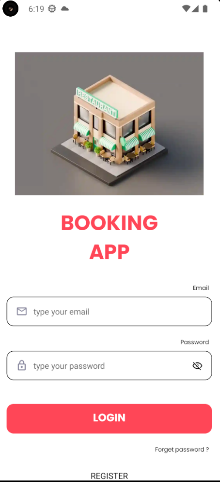
Hình : Giao diện admin xem thống kê và báo cáo.

* + 1. **Tạo tài khoản và xác nhận tài khoản.**
* Cho phép người dùng tạo tài khoản bằng cách nhập email, tên, mật khẩu.
* Mã xác nhận sẽ gửi về email để xác nhận tài khoản người dùng.



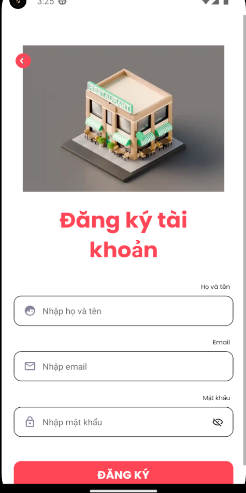
Hình : Giao diện admin quản lý các tài khoản.

* + 1. **Đăng nhập.**
* Cho phép người dùng hoặc admin đăng nhập bằng email và mật khẩu
* Tài khoản của admin khi đăng nhập vào sẽ có 1 màn hình quản trị các thông số của các nhà hàng, đơn đặt hàng, tin nhắn, quản trị tài khoản khác.



Hình : Giao diện đăng nhập.

* + 1. **Đăng ký.**
* Cho phép người dùng đăng ký bằng email



Hình : Giao diện đăng ký.

* 1. **Công nghệ và kỹ thuật áp dụng.**
     1. **Công nghệ sử dụng tại front-end.**

Các công nghệ nhóm sử dụng để thiết kế front-end của ứng dụng bao gồm Expo, ReactJS, React Native, React Navigation, React Router, React Strap, và Amplitude. Trong đó, ReactJS đóng vai trò là thư viện chính để xây dựng giao diện người dùng, cung cấp khả năng phát triển nhanh và linh hoạt thông qua các component có thể tái sử dụng. React Native giúp chuyển giao diện từ React lên ứng dụng di động, còn Expo hỗ trợ quá trình phát triển và triển khai nhanh chóng, đặc biệt với các tính năng đặc thù cho di động. Để quản lý điều hướng và trải nghiệm người dùng mượt mà hơn, nhóm đã tích hợp React Navigation trong ứng dụng di động và React Router cho ứng dụng web. Bên cạnh đó, React Strap cung cấp các component Bootstrap, giúp tối ưu hóa giao diện người dùng. Cuối cùng, Amplitude được sử dụng để theo dõi hành vi người dùng, hỗ trợ phân tích và cải thiện trải nghiệm dựa trên dữ liệu. Sự kết hợp của các công nghệ này giúp nhóm xây dựng một giao diện trực quan, hiện đại, đáp ứng tốt cho cả người dùng web và di động.

* + 1. **Công nghệ sử dụng tại back-end.**

Về phần back-end, nhóm đã lựa chọn NodeJS và ExpressJS làm nền tảng chính để xây dựng server và các API, giúp quản lý các tác vụ phía máy chủ một cách hiệu quả và nhanh chóng. ExpressJS cung cấp một framework đơn giản nhưng mạnh mẽ cho việc định tuyến và xử lý yêu cầu HTTP, tạo cơ sở để phát triển API RESTful một cách dễ dàng. Để hỗ trợ giao tiếp dữ liệu giữa client và server, nhóm sử dụng Apollo GraphQL và GraphQL, cung cấp ngôn ngữ truy vấn linh hoạt và tối ưu, cho phép client lấy đúng dữ liệu cần thiết. MongoDB đóng vai trò là hệ quản trị cơ sở dữ liệu NoSQL, phù hợp để lưu trữ và quản lý dữ liệu một cách linh hoạt và mở rộng dễ dàng khi ứng dụng phát triển. Ngoài ra, Firebase được tích hợp để hỗ trợ các chức năng xác thực người dùng và lưu trữ dữ liệu đám mây, đảm bảo an toàn và tăng tính tiện dụng cho ứng dụng. Sự kết hợp của các công nghệ này giúp back-end của ứng dụng trở nên mạnh mẽ, đáp ứng nhanh chóng và đảm bảo khả năng mở rộng trong tương lai.

* 1. **Công nghệ xây dựng database.**

MongoDB là một hệ thống quản trị cơ sở dữ liệu NoSQL mã nguồn mở, hiện đại và mạnh mẽ, được thiết kế để xử lý dữ liệu linh hoạt và tối ưu cho các ứng dụng hiện đại. Khác biệt so với các cơ sở dữ liệu quan hệ truyền thống như PostgreSQL hay MySQL, MongoDB lưu trữ dữ liệu dưới dạng tài liệu JSON mở rộng (BSON). Điều này mang lại sự linh hoạt vượt trội, cho phép bạn quản lý dữ liệu một cách hiệu quả mà không cần tuân theo các ràng buộc phức tạp của mô hình bảng truyền thống.

MongoDB cung cấp một loạt tính năng mạnh mẽ để hỗ trợ phát triển ứng dụng và quản lý dữ liệu, bao gồm:

* Lưu trữ linh hoạt: Không giống như các cơ sở dữ liệu quan hệ phải tuân theo các cấu trúc bảng cố định, MongoDB cho phép bạn lưu trữ dữ liệu mà không cần định nghĩa schema trước. Điều này rất phù hợp với các ứng dụng yêu cầu thay đổi cấu trúc dữ liệu thường xuyên, cho phép dễ dàng mở rộng và điều chỉnh.
* Khả năng mở rộng cao: MongoDB được xây dựng để hỗ trợ lưu trữ và xử lý lượng dữ liệu lớn thông qua cơ chế sharding (phân tán dữ liệu trên nhiều máy chủ). Tính năng này không chỉ đảm bảo hiệu năng tối ưu mà còn hỗ trợ khả năng mở rộng ngang, rất hữu ích trong các hệ thống phân tán toàn cầu.
* Tích hợp nhanh chóng và dễ dàng: Với tính chất không cần schema cố định, MongoDB đặc biệt phù hợp với các dự án phát triển linh hoạt theo phương pháp Agile. Các nhà phát triển có thể dễ dàng cập nhật và thay đổi cấu trúc dữ liệu mà không làm gián đoạn ứng dụng.
* Khả năng xử lý dữ liệu thời gian thực: MongoDB hỗ trợ các thao tác truy vấn mạnh mẽ, bao gồm các phép lọc, tìm kiếm toàn văn (full-text search), và tổng hợp dữ liệu. Các tính năng này được tối ưu để xử lý dữ liệu với tốc độ cao, đáp ứng nhu cầu phân tích thời gian thực.

MongoDB hiện được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau như:

* Quản lý nội dung: Hỗ trợ các hệ thống CMS (Content Management System) với khả năng tổ chức nội dung động và linh hoạt.
* Mạng xã hội: Lưu trữ các loại dữ liệu đa dạng như bài viết, bình luận, và dữ liệu người dùng với quy mô lớn.
* Phân tích dữ liệu: Xử lý khối lượng dữ liệu khổng lồ để phân tích và tạo báo cáo.
* Internet of Things (IoT): Quản lý và lưu trữ luồng dữ liệu từ các thiết bị IoT theo thời gian thực.

Với ưu điểm về khả năng chịu lỗi cao, tốc độ xử lý vượt trội và khả năng mở rộng linh hoạt, MongoDB không chỉ phù hợp với các tập dữ liệu nhỏ mà còn là giải pháp lý tưởng cho những dự án đòi hỏi xử lý và quản lý khối lượng dữ liệu lớn. Dù là một startup nhỏ hay một tổ chức lớn, MongoDB luôn là một công cụ hữu ích giúp tăng tốc phát triển ứng dụng và đảm bảo tính cạnh tranh trong kỷ nguyên số hóa ngày nay.

1. **PHƯƠNG PHÁP THỰC HIỆN**

Để tiến hành xây dựng và phát triển ứng dụng di động, nhóm em sẽ thực hiện các công việc tuần tự theo quy trình phát triển phần mềm thác nước (Waterfall Model). Cụ thể:

* 1. **Khảo sát trải nghiệm người dùng (UX/UI):**

Thông qua việc khảo sát người dùng và nghiên cứu các ứng dụng di động uy tín cùng lĩnh vực, nhóm đã xác định được giao diện và trải nghiệm mà người dùng mong muốn, cũng như những tính năng thiết yếu cần có.

Dựa trên kết quả khảo sát, nhóm sẽ:

* + - Phác thảo ý tưởng về giao diện ứng dụng, đảm bảo sự tiện lợi và thân thiện với người dùng.
    - Đề xuất và xây dựng danh sách các chức năng chính, tập trung vào việc tối ưu hóa trải nghiệm người dùng trên thiết bị di động.
    - Lên kế hoạch cụ thể để triển khai thiết kế giao diện và luồng hoạt động của ứng dụng, đảm bảo tính đồng bộ và nhất quán.

Quy trình khảo sát sẽ giúp nhóm hiểu rõ hơn nhu cầu thực tế của người dùng, làm cơ sở để phát triển một ứng dụng di động đáp ứng tốt các yêu cầu sử dụng.

* 1. **Phân tích yêu cầu hệ thống**
     1. **Ánh xạ dữ liệu từ mô hình hướng đối tượng (ERD).**

Nhóm sẽ mô hình hóa dữ liệu bằng ERD để mô tả tổng quát các thực thể và mối quan hệ giữa chúng. Các thực thể sẽ được ánh xạ trực tiếp sang các tài liệu trong cơ sở dữ liệu MongoDB, với các mối quan hệ được thể hiện qua referencing (tham chiếu) hoặc embedding (lồng dữ liệu).

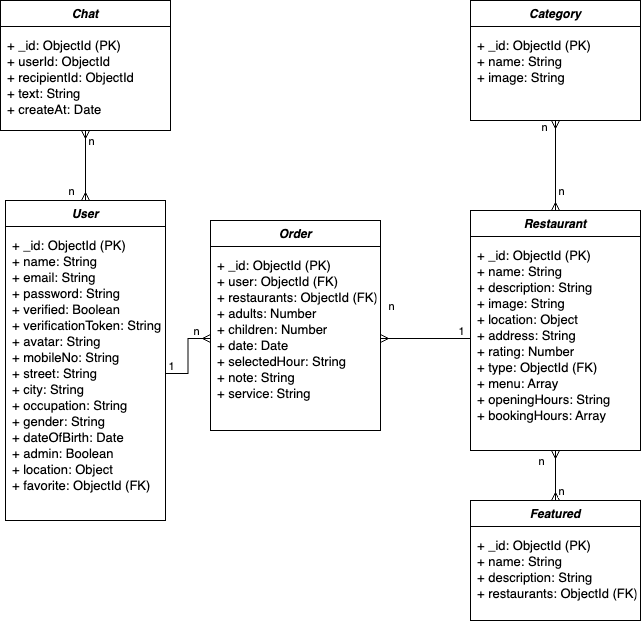
* + 1. **Danh sách các Use Cases.**

Nhóm sẽ liệt kê và phân tích các trường hợp sử dụng (Use Cases), bao gồm các tính năng chính của ứng dụng di động như:

* Tìm kiếm và đặt chỗ.
* Quản lý tài khoản người dùng.
* Xem và đánh giá địa điểm.
* Quản lý danh sách yêu thích.
  + 1. **Sơ đồ lớp và đặc tả chức năng.**

Sơ đồ lớp: Cho thấy mối liên hệ giữa các thực thể và cấu trúc dữ liệu trong ứng dụng.

Đặc tả chức năng: Mô tả chi tiết từng tính năng của ứng dụng, từ giao diện người dùng đến cách các tài liệu dữ liệu tương tác với nhau thông qua MongoDB.



Hình : ERD của dự án.

* 1. **Thiết kế**
     1. **Thiết kế giao diện người dùng (UI/UX)**

Dựa trên kết quả khảo sát, nhóm sẽ:

Tiến hành thiết kế giao diện ứng dụng bằng các công cụ như Figma và tập trung vào trải nghiệm người dùng trên thiết bị di động. Sau đó dùng React Native để thiết kế theo bản vẽ của Figma đã vẽ.

Thiết kế các màn hình chính như trang chủ, tìm kiếm, chi tiết địa điểm, quản lý tài khoản, danh sách yêu thích và màn hình admin.

Tối ưu hóa giao diện để phù hợp với các kích thước màn hình khác nhau, đảm bảo ứng dụng hoạt động tốt trên cả hệ điều hành Android và iOS.

* + 1. **Thiết kế dữ liệu (Database)**

Dữ liệu của ứng dụng sẽ được thiết kế và triển khai sử dụng MongoDB, với các thư viện hỗ trợ trong môi trường Node.js:

* Mongoose: Một thư viện mạnh mẽ để định nghĩa schema, quản lý tài liệu trong MongoDB, và thực hiện các thao tác như tìm kiếm, thêm, sửa đổi, và xóa dữ liệu.
* MongoDB Atlas: Sử dụng nền tảng đám mây để lưu trữ dữ liệu, đảm bảo khả năng mở rộng, bảo mật và hiệu năng.

Cụ thể:

* Thiết kế cơ sở dữ liệu: Tạo các schema với Mongoose cho các thực thể chính như:
* User (Người dùng): Lưu thông tin tài khoản, danh sách yêu thích.
* Restaurant (Địa điểm): Lưu thông tin chi tiết của nhà hàng, hình ảnh, đánh giá.
* Booking (Đặt chỗ): Lưu thông tin lịch sử đặt chỗ của người dùng.
* Orders (Đặt chỗ): Lưu thông tin đặt bàn của người dùng
* Categories (Thể loại): Lưu các loại món ăn mà nhà hàng đó kinh doanh
* Messages (Tin nhắn): Lưu các tin nhắn phản hồi của khách hàng với admin

Công cụ và thư viện hỗ trợ:

* Express.js: Framework để xây dựng API phía server.
* Mongoose: Quản lý và thao tác dữ liệu MongoDB.
* Node.js: Làm backend xử lý logic ứng dụng.
  1. **Xây dựng ứng dụng**
     1. **Xây dựng giao diện người dùng (Frontend)**

Sử dụng React Native để xây dựng giao diện ứng dụng, dựa trên thiết kế đã được phác thảo trong Figma.

Giao diện người dùng (User Interface):

* Người dùng có thể tìm kiếm nhà hàng, xem chi tiết nhà hàng, đặt chỗ, và lưu lại các địa điểm yêu thích.
* Màn hình chính sẽ hiển thị danh sách các nhà hàng, lọc theo các tiêu chí như loại hình, vị trí, đánh giá.
* Người dùng có thể xem chi tiết thông tin nhà hàng, lịch sử đặt chỗ, nhắn tin với admin và thực hiện đặt bàn trực tuyến.

Giao diện quản lý (Admin Interface):

* Các quản trị viên có thể thêm, sửa, và xóa thông tin nhà hàng, kiểm tra thông tin đặt chỗ của khách hàng, và quản lý các yêu cầu hỗ trợ từ người dùng.
* Quản trị viên có thể xem báo cáo tổng hợp về doanh thu, số lượng đặt chỗ, và các đánh giá của khách hàng.

A diagram of a company

Description automatically generatedA diagram of a company

Description automatically generated

Hình : Sitemap cho user và admin.

* + 1. **Xây dựng Backend:**

Chuẩn bị môi trường phát triển: Cài đặt Node.js và các công cụ hỗ trợ. Nhóm chọn sử dụng Android Studio cho việc phát triển ứng dụng di động và Express.js cho backend với Node.js.

Tạo dự án Node.js và thiết lập Dependency Management: Sử dụng npm để quản lý các dependency. Các thư viện cần thiết sẽ được thêm vào file package.json, ví dụ như express, mongoose, cors, v.v.

Xây dựng Model và Repository: Tạo các model đại diện cho dữ liệu trong hệ thống như nhà hàng, đặt bàn, người dùng, v.v. Định nghĩa các thuộc tính và mối quan hệ giữa các đối tượng. Đối với cơ sở dữ liệu, sử dụng MongoDB và kết nối thông qua các thư viện như mongoose (cho MongoDB).

Xây dựng API Controllers và xử lý Business Logic: Tạo các API Controllers trong Express.js để xử lý các yêu cầu HTTP từ phía client (ứng dụng Android). Các API sẽ bao gồm việc tạo mới nhà hàng, đặt bàn, tìm kiếm nhà hàng, v.v. Xử lý các logic nghiệp vụ trong các lớp controller.

Xác thực và phân quyền người dùng: Cấu hình xác thực và phân quyền người dùng bằng cách sử dụng JWT (JSON Web Tokens) hoặc các thư viện xác thực như passport.js để bảo mật các API và chỉ cho phép người dùng hợp lệ truy cập vào các tính năng tương ứng.

Xử lý lỗi và ngoại lệ: Sử dụng middleware trong Express.js để xử lý lỗi và các tình huống ngoại lệ. Đảm bảo người dùng nhận được thông báo lỗi chính xác khi có sự cố xảy ra trong quá trình tương tác với backend.

## **Kiểm thử Vận Hành và Bảo trì.**

* **Kiểm thử:** Nhóm sẽ sử dụng công cụ **Postman** để kiểm thử các API đã được xây dựng. Việc kiểm thử bằng công cụ này giúp phát hiện nhanh chóng các lỗi API và khắc phục chúng hiệu quả hơn. Các API sẽ được kiểm thử liên tục ngay khi hoàn thành để đảm bảo chất lượng. Ngoài ra, nhóm cũng thực hiện kiểm thử chéo trong quá trình đẩy và tải code lên GitHub của từng thành viên.
* **Vận Hành:** Sau khi quá trình kiểm thử hoàn tất, website sẽ được đưa vào hoạt động chính thức. Để website tiếp cận được người dùng nhanh chóng, nhóm sẽ quảng bá thông qua các kênh khác nhau, chẳng hạn như mạng xã hội và tối ưu hóa công cụ tìm kiếm (SEO). Bên cạnh đó, việc theo dõi lưu lượng truy cập và hành vi người dùng sẽ giúp nhóm điều chỉnh website sao cho phù hợp với nhu cầu của người dùng.
* **Bảo trì:** Nhóm sẽ theo dõi thường xuyên quá trình vận hành của website để phát hiện và khắc phục sự cố kịp thời. Đồng thời, việc kiểm tra sao lưu dữ liệu định kỳ sẽ được thực hiện để tránh mất mát dữ liệu quan trọng.

# **TỔNG KẾT.**

## **Kết quả dự kiến đạt được.**

Với các kế hoạch đã đề ra và kiến thức đã tích lũy trong quá trình học, nhóm em tin rằng sẽ phát triển một ứng dụng đặt bàn nhà hàng thật sự thân thiện và tiện ích cho người dùng. Dự kiến, ứng dụng sẽ là cầu nối giữa người dùng và các nhà hàng, giúp người dùng dễ dàng tìm kiếm, đặt bàn, xem thực đơn và đánh giá các nhà hàng.

Đối với người dùng, ứng dụng sẽ cung cấp giao diện với các chức năng như: tìm kiếm nhà hàng, xem thông tin chi tiết, đặt bàn, lưu lại yêu thích, chia sẻ nhà hàng, và xem thực đơn. Điều này giúp đáp ứng nhu cầu tìm kiếm và đặt chỗ của người dùng một cách nhanh chóng và tiện lợi.

Đối với nhà hàng, ứng dụng sẽ cung cấp giao diện quản lý đơn giản nhưng đầy đủ, với các chức năng như: quản lý thông tin nhà hàng, cập nhật thực đơn, quản lý đơn hàng và phản hồi khách hàng. Điều này giúp các nhà hàng dễ dàng duy trì và phát triển dịch vụ.

## **Ưu điểm.**

* **Đáp ứng nhu cầu thị trường:** Ứng dụng đáp ứng nhu cầu ngày càng tăng của người tiêu dùng trong việc tìm kiếm và đặt bàn tại các nhà hàng một cách thuận tiện và nhanh chóng.
* **Sử dụng công nghệ hiện đại:** Ứng dụng được phát triển bằng các công nghệ tiên tiến như **React Native** cho mobile, **Node.js** với **Express.js** cho backend, giúp tối ưu hóa hiệu suất và giao diện.
* **Chức năng hữu ích:** Ứng dụng cung cấp nhiều tính năng như tìm kiếm nhà hàng, đặt bàn trực tuyến, lưu lại các nhà hàng yêu thích, chia sẻ trên mạng xã hội, và quản lý thông tin cá nhân, giúp nâng cao trải nghiệm người dùng.
* **Khả năng mở rộng:** Cấu trúc của ứng dụng cho phép dễ dàng bổ sung các tính năng mới trong tương lai, đáp ứng nhu cầu thay đổi của người dùng và thị trường.
* **Cạnh tranh tốt trên thị trường:** Với giao diện thân thiện và các tính năng tiên tiến, ứng dụng có thể cạnh tranh hiệu quả với các ứng dụng tương tự trên thị trường.

## **Nhược điểm.**

* **Yêu cầu kỹ năng kỹ thuật:** Việc phát triển và duy trì ứng dụng đòi hỏi kiến thức vững về lập trình và quản lý hệ thống, điều này có thể là một thách thức đối với nhóm nếu thiếu kinh nghiệm.
* **Vấn đề bảo mật và hiệu suất:** Các vấn đề về bảo mật và hiệu suất của ứng dụng phụ thuộc vào các công nghệ và framework được sử dụng, cần phải chú trọng cải thiện liên tục.
* **Chi phí đầu tư:** Dù là một ứng dụng di động, nhưng việc duy trì và phát triển ứng dụng vẫn đòi hỏi chi phí ban đầu cho việc phát triển, lưu trữ và bảo trì hệ thống.
* **Trải nghiệm người dùng:** Nếu không tối ưu hóa kỹ lưỡng, ứng dụng có thể gặp vấn đề về hiệu suất và trải nghiệm người dùng, ảnh hưởng đến tỷ lệ chuyển đổi.
* **Cạnh tranh cao:** Thị trường ứng dụng đặt bàn nhà hàng đang có sự cạnh tranh mạnh mẽ, đòi hỏi ứng dụng phải liên tục cải tiến và có chiến lược marketing hiệu quả.

## **Hướng phát triển.**

Trong tương lai, nhóm sẽ tiếp tục phát triển ứng dụng với các tính năng và giao diện như đã đề ra, đồng thời nghiên cứu và áp dụng những công nghệ mới để nâng cao hiệu quả và trải nghiệm người dùng. Nhóm cũng sẽ tập trung vào việc cải thiện bảo mật, tối ưu hóa hiệu suất và mở rộng các tính năng để phù hợp với nhu cầu phát triển của thị trường

# **TÀI LIỆU THAM KHẢO.**

[1]. OpenTable, "Make online reservations at restaurants," OpenTable. [Online]. Available: https://www.opentable.com/. [Accessed: Dec. 15, 2024].

[2]. W3Schools, "MongoDB Tutorial," W3Schools. [Online]. Available: https://www.w3schools.com/mongodb/. [Accessed: Dec. 15, 2024].

[3]. W3Schools, "Node.js Tutorial," W3Schools. [Online]. Available: https://www.w3schools.com/nodejs/. [Accessed: Dec. 15, 2024]

[4]. GeeksforGeeks, "Express.js," GeeksforGeeks. [Online]. Available: https://www.geeksforgeeks.org/express-js/. [Accessed: Dec. 15, 2024].

[5]. JWT.io, "Introduction to JSON Web Tokens," JWT.io. [Online]. Available: https://jwt.io/introduction/. [Accessed: Dec. 15, 2024].

[6]. RapidAPI, "Axios with React: The Complete Guide," RapidAPI. [Online]. Available: https://rapidapi.com/guides/axios-react-complete-guide. [Accessed: Dec. 15, 2024].