

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN CUỐI KỲ MÔN
NHẬP MÔN ỨNG DỤNG DI ĐỘNG
ỨNG DỤNG ĐẶT MÓN ĂN

Sinh viên thực hiện:

Nguyễn Đăng Phúc – 23521203

Bùi Lê Nhật Tri – 23521634

Trần Hữu Đức – 23520320

TP. HỒ CHÍ MINH, THÁNG 07 NĂM 2025

LỜI CẢM ƠN

Lời đầu tiên, nhóm chúng em không biết nói gì hơn ngoài bày tỏ sự biết ơn sâu sắc đến thầy và các bạn. Người đã đồng hành và tận tình giúp đỡ chúng em trong suốt quá trình học tập để hoàn thành đồ án này. Trong suốt quá trình đó, chúng em luôn nhận được sự hướng dẫn đầy tâm huyết, tận tình và tình yêu thương từ mọi người.

Đặc biệt, chúng em xin bày tỏ sự kính trọng và lòng biết ơn sâu sắc nhất đến thầy giáo đã hướng dẫn tụi em đó là ThS. Quan Chí Khánh An – Giảng Viên Khoa Công nghệ phần mềm, Trường ĐH Công nghệ thông tin - ĐHQGHCM, thầy là người đã dạy tụi em môn Nhập Môn Ứng Dụng Di Động và đã trực tiếp hướng dẫn, giúp đỡ cho chúng em để có thể hoàn thành đồ án này một cách tốt nhất.

Cuối cùng, chúng em xin chân thành cảm ơn gia đình và người thân, đã luôn tạo điều kiện, quan tâm, giúp đỡ, động viên chúng em trong suốt quá trình học tập và hoàn thành đồ án này.

Tuy nhiên, với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của một sinh viên và đây cũng là lần đầu chúng em tiếp xúc và làm việc với các ứng dụng di động, chúng em nhận thấy đồ án của mình vẫn còn chưa hoàn chỉnh, có nhiều thiếu sót, có nhiều điều cần chỉnh sửa, có nhiều điều cần học hỏi. Vì thế, chúng em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các thầy cô để chúng em có điều kiện bổ sung, nâng cao kiến thức của mình, phục vụ tốt hơn cho các đồ án kế tiếp cũng như công việc thực tế sau này.

Chúng em xin chân thành cảm ơn!

Mục lục

Chương 1. GIỚI THIỆU ĐỀ TÀI	1
1.1. Giới thiệu chung	1
1.2. Đối tượng và mục tiêu	1
1.3. Các chức năng chính của hệ thống	2
1.3.1. Tìm kiếm và đặt món ăn (client)	2
1.3.2. Quản lý thông tin cá nhân và lịch sử đặt món (client)	2
1.3.3. Quản lý tài khoản, phân quyền người dùng (admin)	2
1.3.4. Quản lý các thực thể trong hệ thống (admin)	2
1.3.5. Quản lý đặt món và món ăn	2
Chương 2. CÁC CÔNG NGHỆ SỬ DỤNG	2
2.1. Ngôn ngữ lập trình: TypeScript	3
2.1.1. Mô tả:	3
2.1.2. Phân tích & Vai trò trong đề tài:	3
2.2. Cơ sở dữ liệu: Firebase Firestore và Firebase Realtime Database	3
2.2.1. Mô tả:	3
2.2.2. Phân tích & Vai trò trong đề tài:	4
2.3. Thư viện/Framework phụ trợ	4
2.3.1. React Native:	4
2.3.2. Node.js:	5
2.3.3. Firebase Auth:	5
2.3.4. Cloudinary:	6
2.3.5. JWT (JSON Web Tokens):	6

2.3.6.	OAuth2:.....	6
2.4.	Công cụ kiểm thử API – Postman.....	7
2.4.1.	Mô tả:.....	7
2.4.2.	Phân tích & Vai trò trong đề tài:	7
2.5.	Công cụ thiết kế: Figma.....	8
2.5.1.	Mô tả:.....	8
2.5.2.	Phân tích & Vai trò trong đề tài:	8
Chương 3.	PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	9
3.1.	Kiến trúc hệ thống	9
3.1.1.	Sơ đồ tổng quan	9
3.1.2.	Thành phần chi tiết.....	9
3.2.	Sơ đồ luồng hoạt động.....	11
3.2.1.	Sơ đồ Use Case.	11
3.2.2.	Sơ đồ Use Case tổng quát.....	11
3.2.3.	Activity Diagram	16
3.2.4.	Danh sách sơ đồ hoạt động	19
3.3.	Thiết kế dữ liệu	23
3.3.1.	Mô tả Tân từ.....	23
3.3.2.	Sơ đồ dữ liệu	27
3.3.3.	Sơ đồ UML	28
3.3.4.	Cấu trúc các bảng dữ liệu trong SQL Bảng 1:Cấu trúc các bảng.....	29
Chương 4.	CÀI ĐẶT – TRIỂN KHAI.....	34
4.1.	Cách tổ chức mã nguồn.....	34
4.2.	Tính năng chính đã lập trình	34

4.3.	Hình ảnh giao diện thực tế.....	37
4.3.1.	Danh sách các màn hình.....	37
4.3.2.	Mô tả các màn hình:	39
4.4.	Mô tả logic xử lý chính.....	82
4.4.1.	Xác thực người dùng	82
4.4.2.	Thêm/Xóa/Sửa dữ liệu.....	82
4.4.3.	Hiển thị danh sách.....	83
4.5.	Khó khăn gặp phải và cách khắc phục	83
4.5.1.	Đồng bộ dữ liệu giữa Firebase Auth và Firestore	83
4.5.2.	Quản lý ảnh và upload file	83
4.5.3.	Quản lý state phức tạp trên mobile.....	83
4.5.4.	Cấu hình môi trường đa nền tảng.....	83
4.5.5.	Truy vấn dữ liệu phức tạp trên Firestore (NoSQL)	83
Chương 5.	TỔNG KẾT	84
5.1.	Kết quả sản phẩm.....	84
5.1.1.	Những chức năng đã hoàn thành.....	84
5.1.2.	Mức độ hoàn thiện của sản phẩm	85
5.2.	Đánh giá hiệu quả	85
5.3.	Hạn chế.....	85
5.4.	Định hướng phát triển.....	85
Chương 6.	ĐÁNH GIÁ KẾT QUẢ THỰC HIỆN.....	86
6.1.	Cấp độ tập thể.....	86
6.2.	Cấp độ cá nhân.....	86
6.3.	Thuận lợi và khó khăn trong quá trình thực hiện.....	87

6.3.1.	Thuận lợi.....	87
6.3.2.	Khó khăn	87

Danh sách thành viên		
Họ và tên	Mã số sinh viên	Vai trò
Nguyễn Đăng Phúc	23521203	Frontend Developer (Trưởng nhóm)
Bùi Lê Nhật Tri	23521634	Backend Developer
Trần Hữu Đức	23520320	Backend Developer

Bảng phân công công việc			
Công việc	Phúc	Tri	Đức
Viết báo cáo	X	X	X
Thiết kế cơ sở dữ liệu	X	X	
Viết sơ đồ tuần tự		X	X
Thiết kế Giao diện	X		
Server API		X	X
Kiểm tra đầu ra báo cáo, phần mềm	X		X
Mức độ hoàn thành các công việc được phân công (%)	100	100	100
Mức độ đóng góp cho kết quả của đề án (%)	33.3	33.3	33.3

Chương 1. GIỚI THIỆU ĐỀ TÀI

1.1. Giới thiệu chung

Trong bối cảnh xã hội hiện đại, nhu cầu đặt món ăn trực tuyến ngày càng gia tăng. Việc đặt món nhanh chóng, chính xác và thuận tiện là yếu tố then chốt để nâng cao trải nghiệm của khách hàng. Đồng thời, các nhà hàng, quán ăn và các dịch vụ giao đồ ăn cũng cần một hệ thống quản lý hiệu quả để xử lý thông tin món ăn, thực đơn, tài khoản người dùng, đối tác giao hàng, khuyến mãi,... một cách dễ dàng và chính xác.

Do đó, đề tài "Ứng dụng đặt món ăn" được thực hiện nhằm cung cấp một hệ thống đặt món online, hỗ trợ đầy đủ các chức năng từ phía người dùng (client) cho đến người quản trị hệ thống (admin), cho phép quản lý thông tin hiệu quả, phân quyền người dùng linh hoạt và tối ưu trải nghiệm sử dụng.

1.2. Đối tượng và mục tiêu

Đối tượng sử dụng:

- Người dùng phổ thông có nhu cầu tra đặt món ăn qua điện thoại (client).
- Quản trị viên hệ thống (admin) với quyền quản lý toàn bộ thông tin trong hệ thống.

Mục tiêu của đề tài:

- Xây dựng một ứng dụng di động đặt món ăn với giao diện trực quan, dễ sử dụng cho người dùng cuối.
- Phát triển giao diện quản trị dành riêng cho admin, hỗ trợ phân quyền người dùng, tạo role mới và quản lý toàn bộ dữ liệu của hệ thống.
- Đảm bảo tính bảo mật, hiệu năng và khả năng mở rộng của hệ thống trong quá trình sử dụng thực tế.

1.3. Các chức năng chính của hệ thống

1.3.1. Tìm kiếm và đặt món ăn (client)

Người dùng có thể tìm kiếm món ăn theo tên món. Hệ thống hiển thị danh sách món ăn/nhà hàng phù hợp để người dùng chọn và tiến hành đặt món.

1.3.2. Quản lý thông tin cá nhân và lịch sử đặt món (client)

Người dùng có thể đăng ký, đăng nhập, chỉnh sửa thông tin cá nhân, và xem lại lịch sử đặt món.

1.3.3. Quản lý tài khoản, phân quyền người dùng (admin)

Admin có thể tạo, chỉnh sửa, và xóa tài khoản. Đồng thời, có thể phân quyền cho tài khoản, tạo các role mới với quyền truy cập tương ứng.

1.3.4. Quản lý các thực thể trong hệ thống (admin)

Admin có thể quản lý các đối tượng quan trọng như:

- Người dùng(User)
- Nhà hàng (Restaurant)
- Món ăn (Food)
- Nhận xét (Review)
- Thông tin đơn đặt (Order)
- Danh mục món ăn (Category)

1.3.5. Quản lý đặt món và món ăn

Hệ thống tự động tính toán và cập nhật số lượng còn lại của món ăn khi có người đặt món, đồng thời thông báo cho người dùng hoặc chủ nhà hàng về tình trạng của đơn hàng.

Chương 2. CÁC CÔNG NGHỆ SỬ DỤNG

Phân tích Công nghệ và Công cụ sử dụng trong Ứng dụng Đặt món ăn

Trong quá trình phát triển ứng dụng đặt món ăn, nhóm đã lựa chọn và sử dụng các công nghệ, cơ sở dữ liệu, thư viện/framework phụ trợ và công cụ thiết kế sau:

2.1. Ngôn ngữ lập trình: TypeScript

2.1.1. Mô tả:

TypeScript là một ngôn ngữ lập trình mã nguồn mở được phát triển bởi Microsoft, là một "superset" của JavaScript, có nghĩa là mọi mã JavaScript hợp lệ đều là mã TypeScript hợp lệ. Điểm khác biệt chính là TypeScript bổ sung tính năng kiểu dữ liệu tĩnh (static typing).

2.1.2. Phân tích & Vai trò trong đề tài:

Tăng cường độ tin cậy và dễ bảo trì: Với tính năng kiểu dữ liệu tĩnh, TypeScript giúp phát hiện lỗi ngay trong quá trình phát triển (compile-time) thay vì khi chạy (runtime), giảm thiểu lỗi, đặc biệt quan trọng trong một ứng dụng có nhiều tương tác người dùng và quản lý dữ liệu phức tạp như đặt món ăn.

Nâng cao năng suất: Hỗ trợ các công cụ phát triển (IDE) cung cấp tính năng tự động hoàn thành mã, gợi ý kiểu, và tái cấu trúc mã hiệu quả hơn, giúp nhóm phát triển nhanh chóng và ít sai sót.

Phù hợp cho dự án lớn và nhóm làm việc: Khi dự án mở rộng với nhiều module và thành viên, TypeScript giúp duy trì cấu trúc mã rõ ràng, dễ hiểu và dễ hợp tác hơn.

Tính tương thích: Vì là một superset của JavaScript, TypeScript có thể tận dụng toàn bộ hệ sinh thái rộng lớn của JavaScript (bao gồm Node.js và React Native).

2.2. Cơ sở dữ liệu: Firebase Firestore và Firebase Realtime Database

2.2.1. Mô tả:

Cả Firebase Firestore và Firebase Realtime Database đều là các dịch vụ cơ sở dữ liệu NoSQL dựa trên đám mây, được cung cấp bởi Google Firebase. Chúng

được thiết kế để đồng bộ hóa dữ liệu thời gian thực và cung cấp SDK dễ dàng tích hợp vào ứng dụng di động và web.

2.2.2. Phân tích & Vai trò trong đề tài:

Đồng bộ hóa thời gian thực: Đây là lợi thế lớn nhất, cho phép cập nhật tức thì trạng thái đơn hàng (ví dụ: "đang chuẩn bị", "đang giao"), vị trí tài xế, hoặc sự thay đổi trong thực đơn. Điều này cực kỳ quan trọng để nâng cao trải nghiệm người dùng trong ứng dụng đặt món ăn.

Dễ dàng tích hợp với ứng dụng di động: Firebase cung cấp SDK mạnh mẽ cho React Native, giúp việc kết nối và quản lý dữ liệu từ phía client trở nên đơn giản, giảm thiểu lượng mã backend phải viết.

Khả năng mở rộng: Cả hai dịch vụ đều có khả năng mở rộng tốt để đáp ứng lượng dữ liệu và người dùng tăng trưởng.

Sự khác biệt và ứng dụng:

- **Firestore:** Phù hợp cho dữ liệu có cấu trúc hơn, cần truy vấn phức tạp (ví dụ: thông tin chi tiết món ăn, nhà hàng, đơn hàng phức tạp, dữ liệu người dùng). Firestore cung cấp các truy vấn mạnh mẽ hơn và cấu trúc dữ liệu theo bộ sưu tập và tài liệu.
- **Realtime Database:** Thích hợp cho dữ liệu đơn giản, đồng bộ hóa nhanh chóng và tần suất cập nhật cao (ví dụ: trạng thái online/offline của tài xế, chat giữa khách hàng và tài xế/nhà hàng).

Việc sử dụng kết hợp cả hai giúp tận dụng điểm mạnh của từng loại cho các kịch bản dữ liệu khác nhau trong ứng dụng đặt món.

2.3. Thư viện/Framework phụ trợ

2.3.1. React Native:

2.3.1.1. Mô tả:

React Native là một framework phát triển giao diện người dùng (UI) mã nguồn mở, được tạo bởi Facebook. Nó cho phép phát triển ứng dụng di động đa nền tảng (iOS và Android) bằng cách sử dụng JavaScript (và TypeScript).

2.3.1.2. Phân tích & Vai trò trong đề tài:

Cho phép nhóm viết một codebase duy nhất cho cả ứng dụng Android và iOS, tiết kiệm thời gian và nguồn lực. Giao diện người dùng được xây dựng bằng cách sử dụng các component, giúp phát triển nhanh chóng và dễ tái sử dụng.

2.3.2. Node.js:

2.3.2.1. Mô tả:

Node.js là một môi trường chạy JavaScript mã nguồn mở và đa nền tảng, cho phép chạy mã JavaScript ở phía máy chủ (server-side).

2.3.2.2. Phân tích & Vai trò trong đề tài:

Được sử dụng để xây dựng các API backend (nếu có các tác vụ phức tạp không thể xử lý trực tiếp trên Firebase Cloud Functions), quản lý logic nghiệp vụ phức tạp, tích hợp với các dịch vụ bên thứ ba (ví dụ: cổng thanh toán, API bản đồ). Node.js với mô hình bất đồng bộ, không chặn (non-blocking I/O) rất hiệu quả cho các ứng dụng I/O nặng như một hệ thống đặt món.

2.3.3. Firebase Auth:

2.3.3.1. Mô tả:

Là một dịch vụ xác thực được cung cấp bởi Firebase, hỗ trợ nhiều phương thức xác thực khác nhau (email/mật khẩu, Google, Facebook, số điện thoại, v.v.).

2.3.3.2. Phân tích & Vai trò trong đề tài:

Cung cấp giải pháp xác thực người dùng mạnh mẽ, an toàn và dễ triển khai, giúp đơn giản hóa quy trình đăng ký/đăng nhập và quản lý tài khoản người dùng trong ứng dụng đặt món.

2.3.4. Cloudinary:

2.3.4.1. Mô tả:

Cloudinary là một dịch vụ quản lý hình ảnh và video dựa trên đám mây, cung cấp các tính năng tải lên, lưu trữ, xử lý (thay đổi kích thước, cắt, thêm hiệu ứng) và phân phối nội dung đa phương tiện.

2.3.4.2. Phân tích & Vai trò trong đề tài:

Quan trọng để quản lý hình ảnh món ăn, nhà hàng, avatar người dùng. Cloudinary giúp tối ưu hóa hình ảnh cho các thiết bị di động khác nhau, cải thiện hiệu suất tải trang và giảm tải cho server backend, đồng thời đảm bảo hình ảnh chất lượng cao.

2.3.5. JWT (JSON Web Tokens):

2.3.5.1. Mô tả:

JWT là một tiêu chuẩn mở (RFC 7519) để tạo ra các token truy cập nhỏ gọn, an toàn và tự chứa thông tin giữa các bên.

2.3.5.2. Phân tích & Vai trò trong đề tài:

Được sử dụng cho cơ chế xác thực dựa trên token. Sau khi người dùng đăng nhập thành công (thông qua Firebase Auth hoặc cơ chế tùy chỉnh), một JWT sẽ được tạo và gửi về client. Client sẽ đính kèm token này trong mỗi yêu cầu tới API backend để chứng minh danh tính mà không cần gửi lại thông tin đăng nhập, đảm bảo tính bảo mật và hiệu quả của các giao tiếp API.

2.3.6. OAuth2:

2.3.6.1. Mô tả:

OAuth2 là một framework ủy quyền (authorization framework) cho phép ứng dụng của bên thứ ba truy cập tài nguyên của người dùng trên một dịch vụ khác (ví dụ: Google, Facebook) mà không cần biết mật khẩu của người dùng.

2.3.6.2. Phân tích & Vai trò trong đề tài:

Hỗ trợ tính năng "đăng nhập bằng Google/Facebook" trong ứng dụng. Điều này giúp tăng trải nghiệm người dùng, giảm thiểu rào cản đăng ký và tận dụng thông tin tài khoản đã có sẵn từ các dịch vụ phổ biến, làm cho quá trình xác thực nhanh chóng và tiện lợi hơn.

2.4. Công cụ kiểm thử API – Postman

2.4.1. Mô tả:

Postman là một nền tảng phổ biến dành cho phát triển API, cho phép các nhà phát triển thiết kế, xây dựng, kiểm thử và tài liệu hóa API. Nó cung cấp một giao diện đồ họa thân thiện để gửi các yêu cầu HTTP (GET, POST, PUT, DELETE, v.v.) tới API và xem phản hồi từ máy chủ.

2.4.2. Phân tích & Vai trò trong đề tài:

Kiểm thử chức năng API: Postman đóng vai trò thiết yếu trong việc kiểm thử các API backend được xây dựng bằng Node.js và tương tác với Firebase. Nhóm có thể dễ dàng gửi các yêu cầu để kiểm tra các chức năng như đăng nhập, đăng ký, thêm/sửa/xóa món ăn, quản lý đơn hàng, v.v., đảm bảo các API hoạt động đúng như mong đợi trước khi tích hợp vào ứng dụng di động.

Phát triển API nhanh chóng: Việc có thể gửi và nhận phản hồi API ngay lập tức giúp tăng tốc quá trình phát triển và gỡ lỗi cho các endpoint API.

Cộng tác nhóm: Postman hỗ trợ tạo các bộ sưu tập (collections) API, cho phép các thành viên trong nhóm chia sẻ và cộng tác trên cùng một tập hợp các yêu cầu API, đảm bảo tính nhất quán trong quá trình kiểm thử.

Xác minh dữ liệu: Sau khi gửi yêu cầu, Postman hiển thị phản hồi từ server, giúp nhóm xác minh xem dữ liệu được gửi đi và nhận về có đúng định dạng và nội dung mong muốn hay không.

Hỗ trợ JWT/OAuth2: Postman có tích hợp sẵn các tính năng để quản lý và gửi kèm token (như JWT) trong các yêu cầu API, rất tiện lợi khi kiểm thử các API được bảo vệ bằng xác thực và phân quyền.

2.5. Công cụ thiết kế: Figma

2.5.1. Mô tả:

Figma là một công cụ thiết kế giao diện người dùng (UI/UX) dựa trên nền tảng đám mây, cho phép làm việc cộng tác trong thời gian thực.

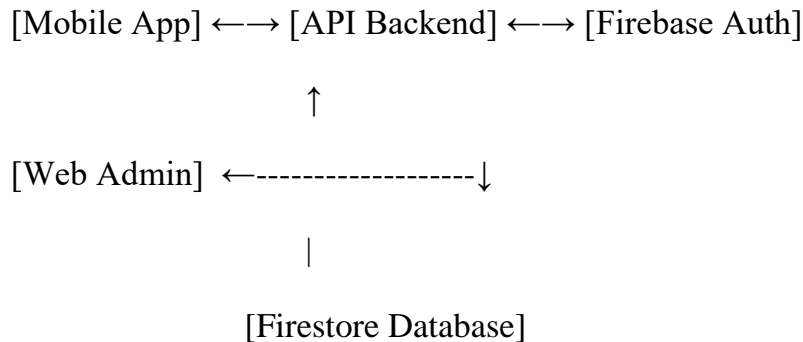
2.5.2. Phân tích & Vai trò trong đề tài:

- Thiết kế UI/UX trực quan: Giúp nhóm tạo ra các bản mock-up, wireframe và prototype chi tiết cho giao diện ứng dụng di động và website, đảm bảo tính thẩm mỹ và trải nghiệm người dùng tốt nhất.
- Cộng tác hiệu quả: Với khả năng làm việc trực tuyến và cộng tác theo thời gian thực, các thành viên trong nhóm có thể cùng nhau thiết kế, góp ý và chỉnh sửa, đồng thời dễ dàng chia sẻ thiết kế với những người liên quan.
- Hỗ trợ phát triển: Figma giúp xuất các thông số kỹ thuật (CSS, SVG, v.v.) và tài nguyên thiết kế, tạo điều kiện thuận lợi cho đội ngũ phát triển chuyển đổi thiết kế thành code.

Chương 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1. Kiến trúc hệ thống

3.1.1. Sơ đồ tổng quan



Mobile app: Ứng dụng React Native cho iOS/Android

Web admin: Ứng dụng React (Ant Design), phục vụ quản trị và nhà hàng

Backend: Node.js/Express, giữ vai trò trung gian logic, xác thực, truy xuất dữ liệu

Database: Firebase Firestore

Authentication: Firebase Auth

3.1.2. Thành phần chi tiết

3.1.2.1. Frontend

a. Mobile

- React Native + Expo
- Quản lý routing, navigation
- Sử dụng Formik, Yup cho form/validate
- State management: React Context, React Query

Chức năng chính:

- Đăng ký, đăng nhập, xác thực
- Tìm kiếm món ăn, nhà hàng, lọc/sắp xếp
- Xem chi tiết món ăn, nhà hàng
- Thêm món vào giỏ hàng, chỉnh sửa số lượng
- Đặt hàng, thanh toán (giả lập)

- Theo dõi đơn hàng, đánh giá nhà hàng/món ăn
- Quản lý tài khoản

b. Web Admin

- React + Ant Design
- Giao diện quản lý thân thiện cho nhà hàng/quản trị viên
- CRUD món ăn, danh mục, quản lý đơn hàng, tài khoản nhà hàng
- Upload hình ảnh (tích hợp Cloudinary)
- Biểu đồ, báo cáo đơn hàng, doanh thu
- Quản lý user

3.1.2.2. Backend

Node.js/Express

- Cấu trúc service-based: Tách từng module cho user, nhà hàng, món ăn, cart, order, category...
- Middlewares
- Xác thực JWT/Firebase token
- Xử lý quyền truy cập
- Tích hợp Firebase Admin SDK
- Đọc/ghi Firestore
- Tạo/xóa user Auth
- Phân quyền
- Xử lý nghiệp vụ:
 - Tạo user, nhà hàng, món ăn, đơn hàng
 - Liên kết dữ liệu Firestore đa collection (subcollection)
 - Xóa user, đồng bộ Auth/Firebase
 - Batch write, transaction khi cần đảm bảo toàn vẹn dữ liệu

3.1.2.3. Database

Firestore (NoSQL):

- Collection: Users, Restaurants, Dishes, Orders, Carts, Categories, Reviews, Notifications...
- Các sub-collection: Addresses (trong Users), Order_details (trong Orders)...
- Dữ liệu phân cấp, tối ưu cho truy vấn nhanh, realtime
- Firebase Authentication:
 - o Đăng ký, đăng nhập, reset password, xác thực hai lớp (nếu cần)
 - o Lưu role (CUSTOMER, RESTAURANT, ADMIN), đồng bộ thông tin với Firestore

3.2. Sơ đồ luồng hoạt động

3.2.1. Sơ đồ Use Case.

3.2.1.1. Chức năng

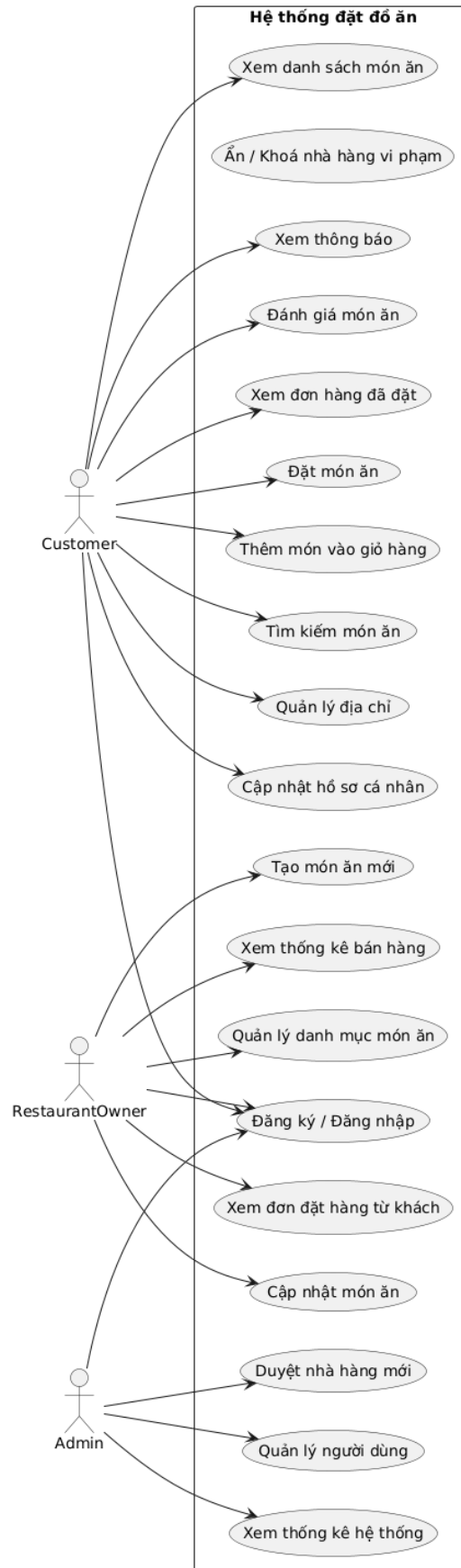
Use Case (Use Case Diagram) giúp hiểu rõ hệ thống sẽ làm gì, ai sẽ tương tác với nó và tương tác như thế nào. Đây là bước đầu tiên và quan trọng trong phân tích yêu cầu phần mềm, đặc biệt trong các hệ thống như cụ thể ở đây là trang web quản lý bán vé máy bay.

Các mục đích khi xây dựng sơ đồ Use Case:

Mục đích	Giải thích
Xác định chức năng hệ thống	Cho biết hệ thống có những chức năng gì (đặt vé, thanh toán, quản lý chuyến bay...)
Xác định các đối tượng sử dụng (Actor)	Ai sẽ tương tác với hệ thống (người dùng, khách chưa đăng nhập, quản trị viên...)
Mô tả mối quan hệ giữa actor và chức năng	Actor nào được phép sử dụng chức năng nào
Truyền đạt yêu cầu cho nhà phát triển	Là công cụ giao tiếp hiệu quả giữa nhà phân tích, khách hàng, lập trình viên
Làm cơ sở thiết kế và kiểm thử phần mềm	Dùng để viết tài liệu đặc tả, test case và thiết kế hệ thống backend/frontend
Giúp khách hàng dễ hiểu hệ thống hơn	Ngay cả người không rành kỹ thuật cũng hiểu được hệ thống có gì và dùng ra sao

3.2.2. Sơ đồ Use Case tổng quát

Sơ đồ Use Case tổng quát mô tả tổng quan các chức năng chính của hệ thống bán vé máy bay qua web, cùng với vai trò của các tác nhân (actors) tương tác với hệ thống. Đây là một công cụ quan trọng trong giai đoạn phân tích yêu cầu phần mềm, giúp hiểu rõ ai sử dụng hệ thống và sử dụng để làm gì.



Các tác nhân (Actors) trong hệ thống

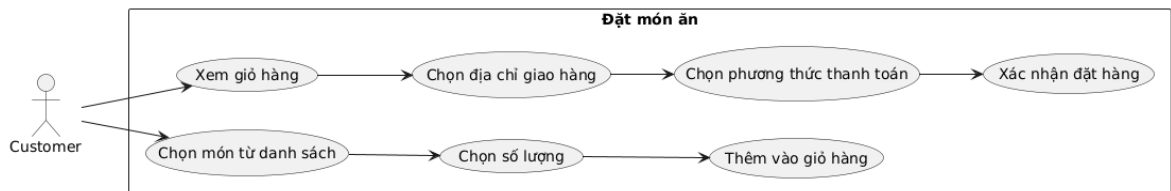
Khách: Người chưa đăng nhập, chỉ có thể sử dụng các chức năng xem và tìm kiếm thông tin.

Người dùng: Người đã đăng ký và đăng nhập, có thể sử dụng đầy đủ tính năng đặt vé.

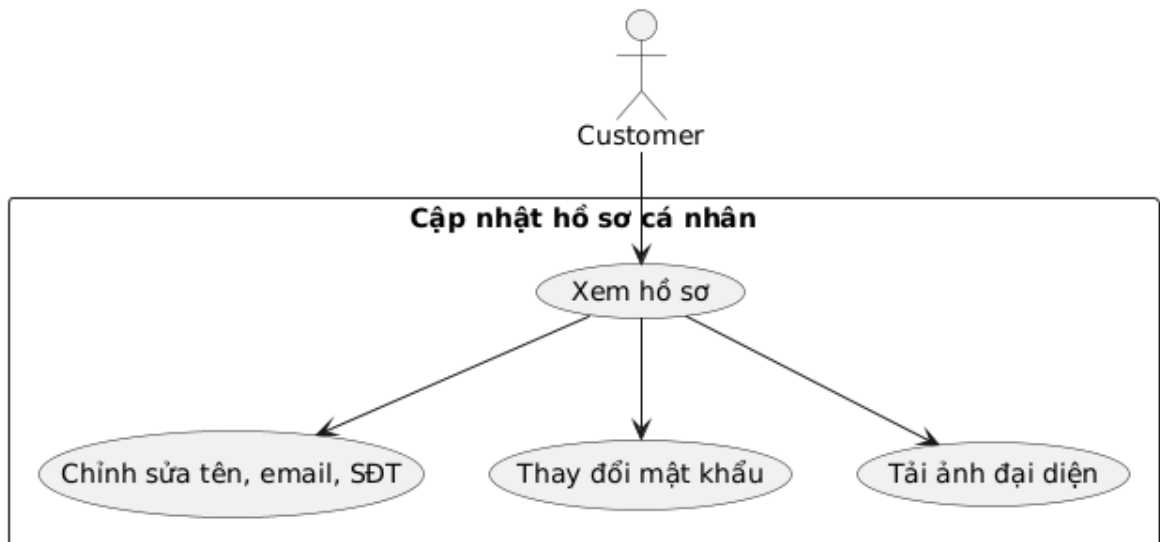
Quản trị viên: Người quản lý toàn bộ hệ thống, có quyền truy cập các chức năng quản lý chuyên sâu.

3.2.2.1. Sơ đồ Use Case chi tiết.

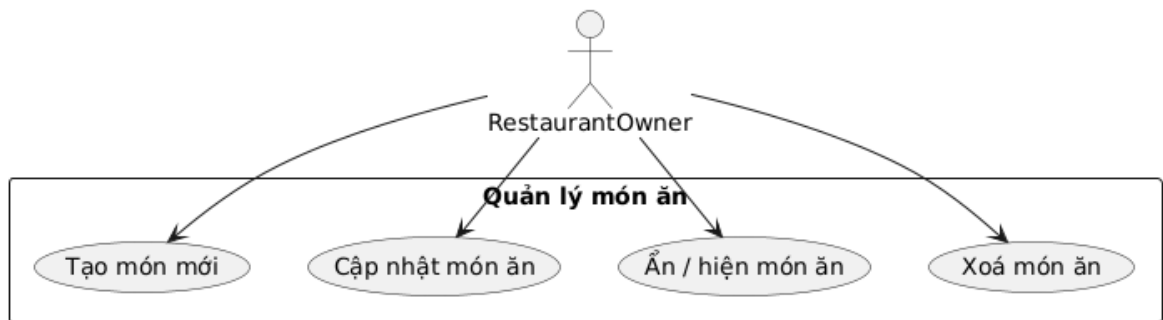
Đặt món ăn



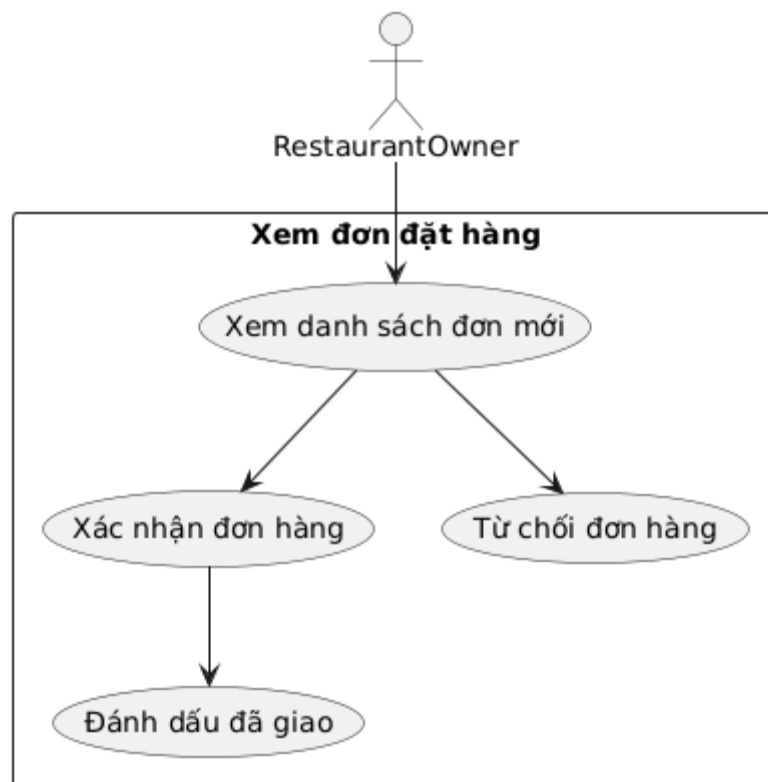
Cập nhật hồ sơ cá nhân



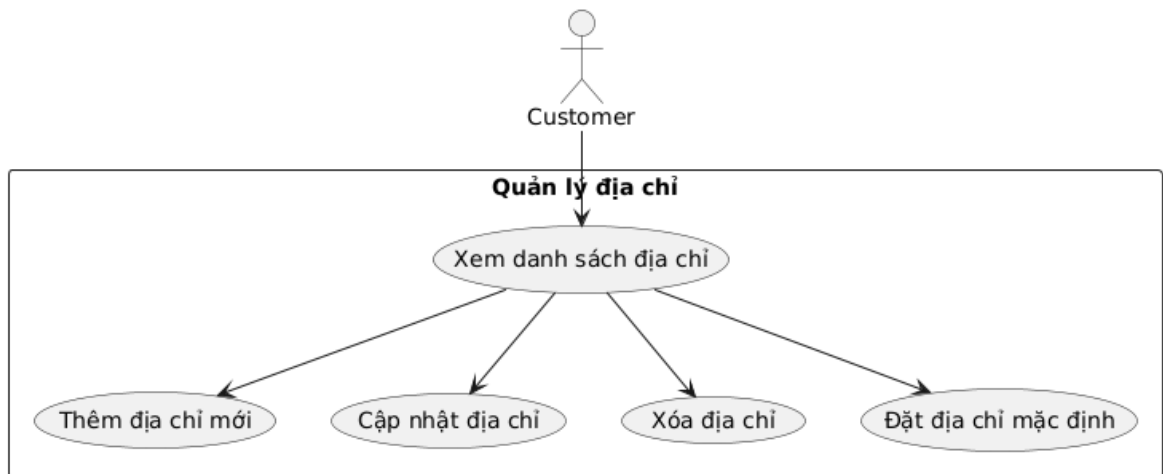
Quản lý món ăn



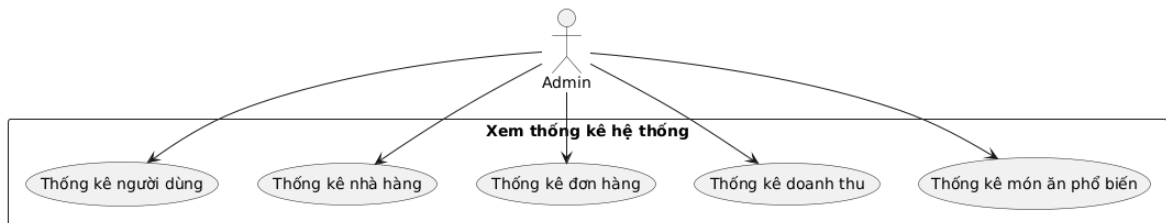
Xem đơn đặt hàng



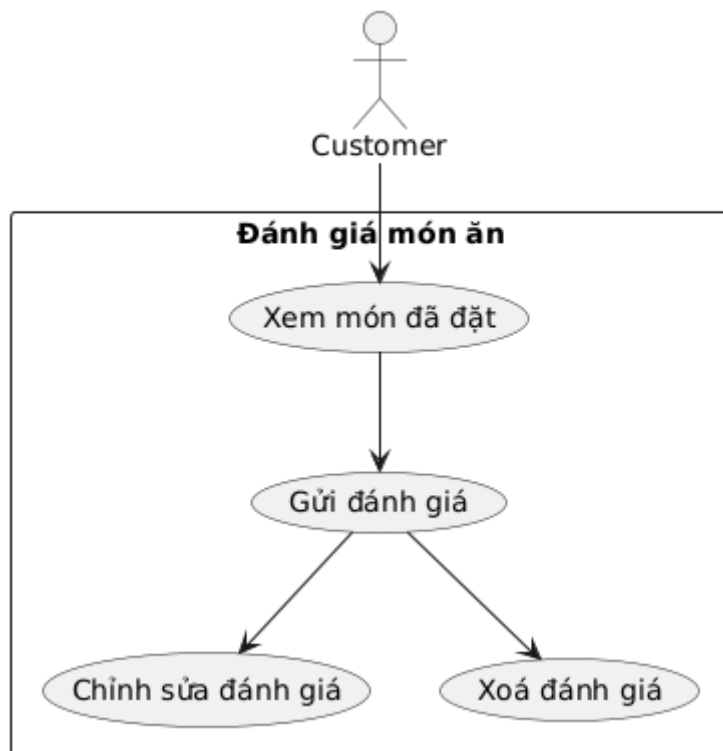
Quản lý địa chỉ



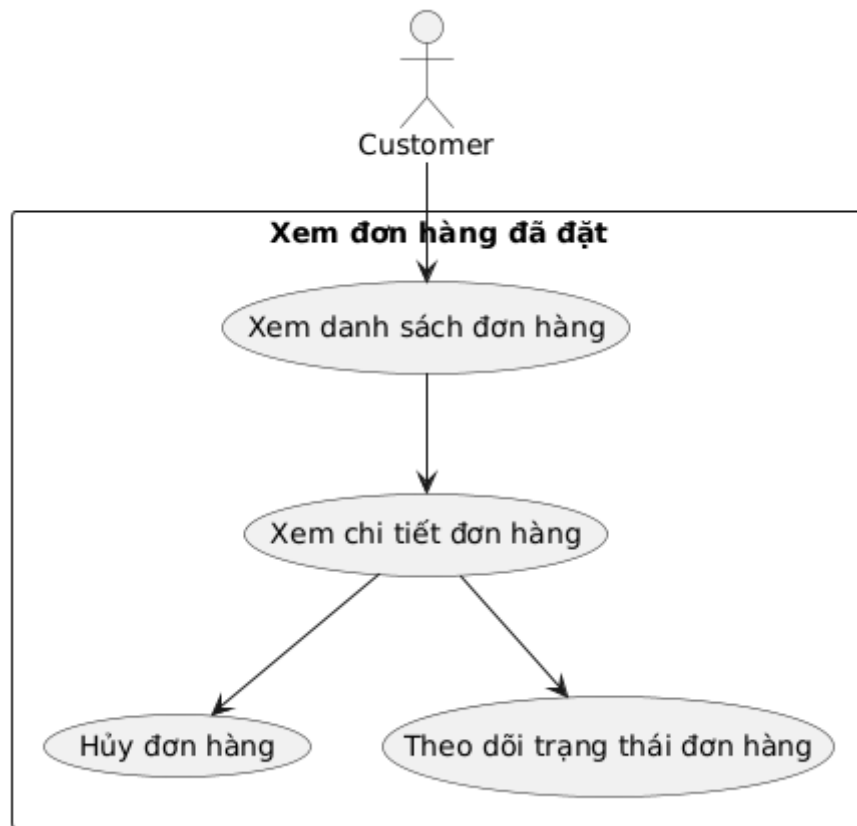
Xem thống kê hệ thống



Đánh giá món ăn



Xem đơn đặt hàng



3.2.3. Activity Diagram

3.2.3.1. Chức năng

Trong quá trình phân tích và thiết kế hệ thống, biểu đồ hoạt động (Activity Diagram) là một công cụ trực quan và quan trọng, được sử dụng để mô tả chi tiết luồng xử lý nghiệp vụ của các chức năng trong hệ thống. Đối với hệ thống web đặt đồ ăn trực tuyến, biểu đồ hoạt động có vai trò thiết yếu trong việc thể hiện trình tự các hành động giữa người dùng và hệ thống, đồng thời làm rõ các tình huống rẽ nhánh, vòng lặp và xử lý ngoại lệ.

a. Mô tả chi tiết luồng xử lý các chức năng chính

Biểu đồ hoạt động được sử dụng để mô tả quy trình nghiệp vụ của các chức năng cốt lõi trong hệ thống như:

- Đăng ký, đăng nhập người dùng.
- Tìm kiếm và đặt món ăn.

- Quản lý giỏ hàng và thanh toán.
- Theo dõi đơn hàng và gửi đánh giá.
- Quản lý thực đơn, danh mục, doanh thu (đối với nhà hàng).
- Duyệt nhà hàng, thống kê và kiểm soát hệ thống (đối với quản trị viên).

Thông qua biểu đồ, từng bước xử lý được trình bày rõ ràng theo thứ tự thực thi, giúp nhóm phát triển dễ dàng hiểu và triển khai đúng yêu cầu.

b. Thể hiện sự tương tác giữa người dùng và hệ thống

Biểu đồ hoạt động minh họa rõ sự luân chuyển giữa các hành động của người dùng và phản hồi từ hệ thống. Ví dụ:

- Người dùng chọn món → thêm vào giỏ → chọn địa chỉ → thanh toán → nhận xác nhận.
- Nhà hàng tiếp nhận đơn hàng → xác nhận → cập nhật trạng thái giao hàng.
- Admin duyệt nhà hàng → gửi thông báo kích hoạt hoặc từ chối.

Điều này giúp xác định ranh giới trách nhiệm giữa người dùng và hệ thống trong từng quy trình nghiệp vụ.

c. Làm rõ các điều kiện rẽ nhánh và xử lý ngoại lệ

Activity Diagram hỗ trợ thể hiện các điều kiện xử lý khác nhau như:

- Nếu thanh toán thành công → chuyển đến bước xác nhận đơn hàng.
 - Nếu thất bại → hiển thị lỗi và cho phép chọn lại phương thức thanh toán.
 - Nếu đơn hàng bị hủy → không cho đánh giá món ăn.
- Nhờ vậy, nhóm phát triển dễ dàng bao phủ đầy đủ các trường hợp thường và ngoại lệ khi lập trình hoặc kiểm thử.

d. Hỗ trợ xây dựng test case và kiểm thử hệ thống

Biểu đồ hoạt động đóng vai trò làm cơ sở để xây dựng các test case chức năng, đặc biệt trong:

- Kiểm tra quy trình chính xác theo từng nhánh điều kiện.
- Mô phỏng hành vi của người dùng theo các kịch bản thực tế.
- Phát hiện các lỗi logic hoặc thiếu sót trong luồng xử lý nghiệp vụ.

e. Tăng tính trực quan và dễ hiểu cho các bên liên quan

Biểu đồ hoạt động là công cụ trung gian giúp kết nối giữa:

- Người dùng / khách hàng (không chuyên kỹ thuật).
- Nhà phân tích nghiệp vụ (BA).

- Lập trình viên, kiểm thử viên và quản lý dự án.

Nhờ biểu đồ, tất cả các bên đều có thể hiểu được cách hệ thống vận hành mà không cần tiếp cận sâu vào mã nguồn hay cấu trúc kỹ thuật.

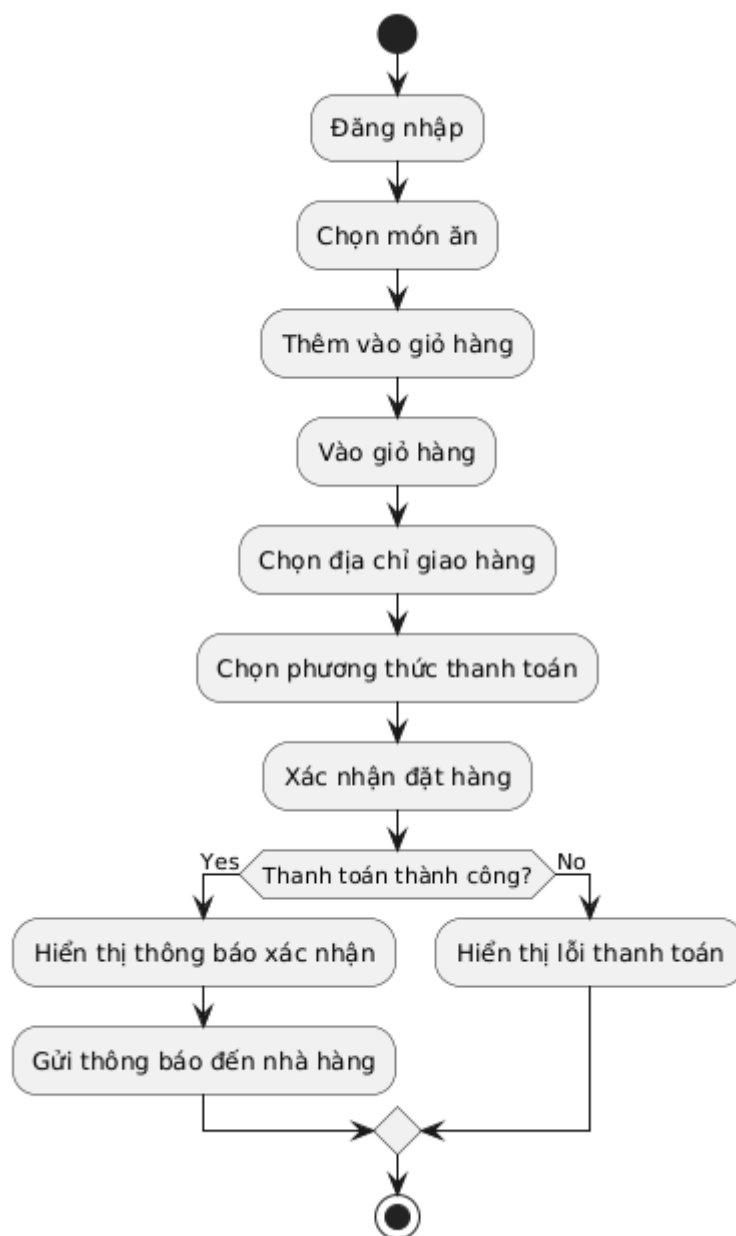
f. Đảm bảo hệ thống phù hợp với quy trình thực tế

Thông qua việc mô tả rõ ràng từng bước xử lý, biểu đồ hoạt động giúp nhóm phát triển:

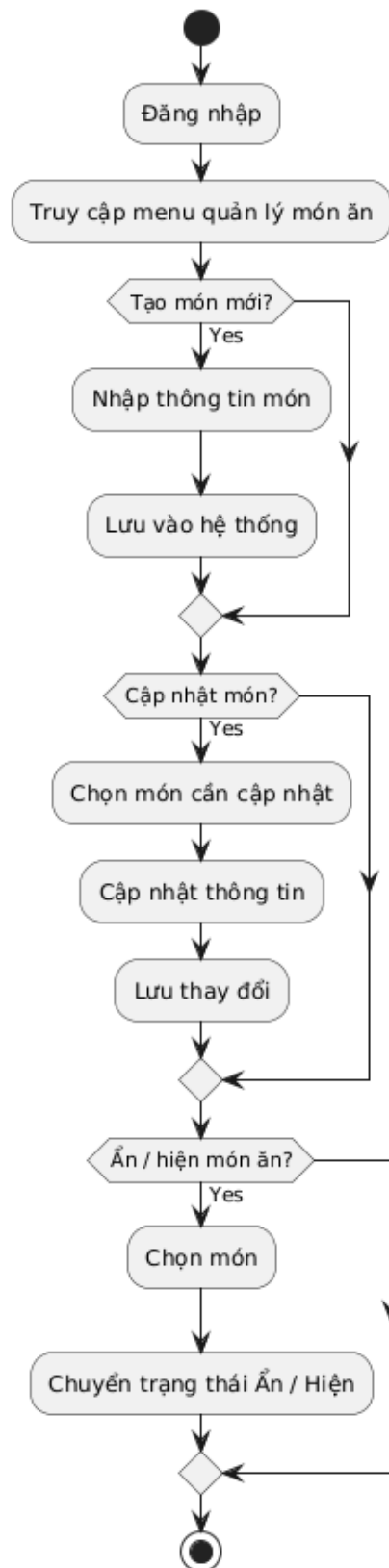
- Xác định đầy đủ các bước trong mỗi chức năng.
- Tránh bỏ sót các tình huống nghiệp vụ thực tế.
- Đảm bảo hệ thống được xây dựng đúng theo quy trình người dùng mong muốn.

3.2.4. Danh sách sơ đồ hoạt động

3.2.4.1. Đặt món ăn



3.2.4.2. Quản lý món ăn



3.2.4.3. Gửi đánh giá món ăn



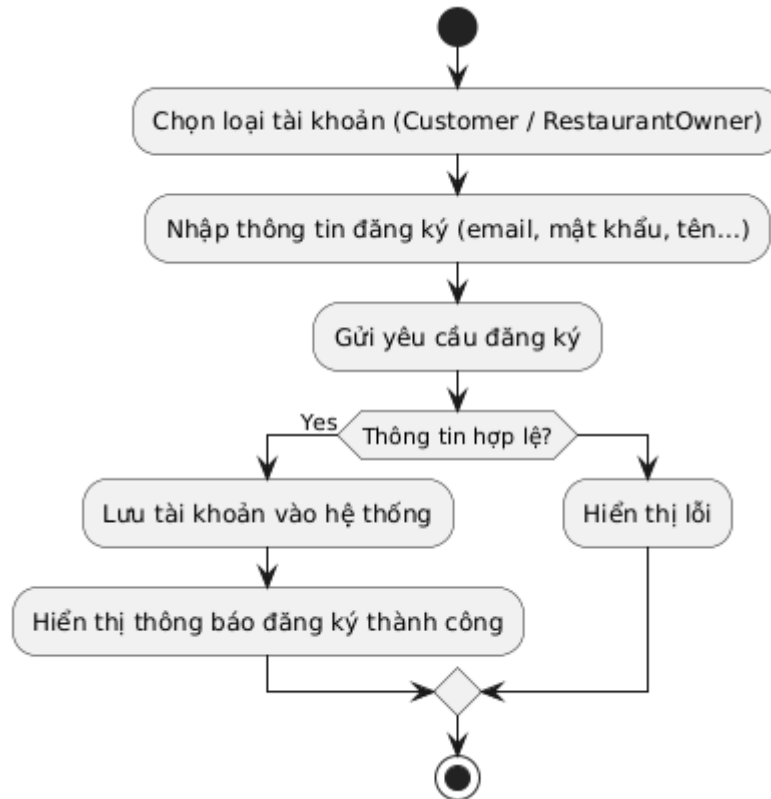
3.2.4.4. Xem thống kê



3.2.4.5. Quản lý địa chỉ



3.2.4.6. Đăng ký người dùng



3.3. Thiết kế dữ liệu

3.3.1. Mô tả Tân từ

1. Bảng "Tài khoản" Account

(Account ID, Email, Password, Role, Full Name, Created Date, Phone Number, Avatar)

Tân từ: Bảng "Tài khoản" lưu trữ thông tin tài khoản người dùng bao gồm: mã định danh (Account ID) là khóa chính, địa chỉ email (Email), mật khẩu (Password), vai trò của người dùng (Role) có thể là admin, restaurant, hoặc customer, họ tên đầy đủ (Full Name), ngày tạo tài khoản (Created Date), số điện thoại (Phone Number), và đường dẫn ảnh đại diện (Avatar).

2. Bảng "Địa chỉ" (Address)

(Address ID, Address Name, Account ID)

Tân từ: Bảng "Địa chỉ" lưu trữ thông tin về các địa chỉ của người dùng, bao gồm: mã định danh địa chỉ (Address ID) là khóa chính, tên địa chỉ (Address Name), và mã tài khoản (Account ID) là khóa ngoại liên kết với bảng "Tài khoản".

3. Bảng "Nhà hàng" (Restaurant)

(Restaurant ID, Account ID, Restaurant Name, Address, Status, Restaurant Image, Phone Number, Rating, Created Date)

Tân từ: Bảng "Nhà hàng" lưu trữ thông tin chi tiết về các nhà hàng, bao gồm: mã định danh nhà hàng (Restaurant ID) là khóa chính, mã tài khoản (Account ID) là khóa ngoại liên kết với tài khoản của chủ nhà hàng, tên nhà hàng (Restaurant Name), địa chỉ chi tiết (Address), trạng thái hoạt động (Status) có thể là pending, active, hoặc closed, đường dẫn ảnh nhà hàng (Restaurant Image), số điện thoại liên hệ (Phone Number), điểm đánh giá (Rating), và ngày tạo hồ sơ nhà hàng (Created Date).

4. Bảng "Danh mục" (Category)

(Category ID, Category Name, Description, Created Date, Purchase Count, Illustration Image)

Tân từ: Bảng "Danh mục" lưu trữ thông tin về các loại danh mục món ăn, bao gồm: mã định danh danh mục (Category ID) là khóa chính, tên danh mục (Category Name), mô tả chi tiết (Description), ngày tạo danh mục (Created Date), số lượt mua của các món ăn trong danh mục này (Purchase Count), và đường dẫn ảnh minh họa (Illustration Image).

5. Bảng "Món ăn" (Dish)

(Dish ID, Restaurant ID, Category ID, Dish Name, Description, Price, Product Image, Purchase Count, Available for Sale, Quantity Remaining, Created Date, Rating)

Tân từ: Bảng "Món ăn" lưu trữ thông tin chi tiết về từng món ăn, bao gồm: mã định danh món ăn (Dish ID) là khóa chính, mã nhà hàng (Restaurant ID) là khóa ngoại liên kết với nhà hàng cung cấp món ăn, mã danh mục (Category ID) là khóa ngoại liên kết với danh mục của món ăn, tên món ăn (Dish Name), mô tả (Description), giá (Price), đường dẫn ảnh sản phẩm (Product Image), số lượt mua của món ăn (Purchase Count), trạng thái còn bán (Available for Sale) là một giá trị boolean, số lượng còn lại (Quantity Remaining), ngày tạo thông tin món ăn (Created Date), và điểm đánh giá (Rating).

6. Bảng "Giỏ hàng" (Cart)

(Cart ID, Account ID)

Tân từ: Bảng "Giỏ hàng" lưu trữ thông tin về giỏ hàng của người dùng, bao gồm: mã định danh giỏ hàng (Cart ID) là khóa chính, và mã tài khoản (Account ID) là khóa ngoại liên kết với tài khoản của khách hàng.

7. Bảng "Đơn hàng" (Order)

(Order ID, Account ID, Restaurant ID, Total Amount, Status, Order Time, Delivery Person Phone)

Tân từ: Bảng "Đơn hàng" lưu trữ thông tin tổng quan về các đơn đặt hàng, bao gồm: mã định danh đơn hàng (Order ID) là khóa chính, mã tài khoản (Account ID) là khóa ngoại của người đặt hàng, mã nhà hàng (Restaurant ID) là khóa ngoại của nhà hàng nhận đơn, tổng số tiền của đơn hàng (Total Amount), trạng thái của đơn

hàng (Status) có thể là pending, processing, delivered, hoặc cancelled, thời gian đặt hàng (Order Time), và số điện thoại của người giao hàng (Delivery Person Phone).

8. Bảng "Đơn hàng chi tiết" (Order Detail)

(Order ID, Dish ID, Quantity, Unit Price)

Tân từ: Bảng "Đơn hàng chi tiết" lưu trữ thông tin chi tiết về các món ăn trong từng đơn hàng, bao gồm: mã đơn hàng (Order ID) là khóa ngoại liên kết với bảng "Đơn hàng", mã món ăn (Dish ID) là khóa ngoại liên kết với bảng "Món ăn", số lượng của món ăn đó trong đơn hàng (Quantity), và đơn giá của món ăn tại thời điểm đặt hàng (Unit Price). Đây là một bảng trung gian cho mối quan hệ nhiều-nhiều giữa Đơn hàng và Món ăn.

9. Bảng "Đánh giá" (Review)

(Review ID, Account ID, Dish ID, Stars, Comment, Created Date)

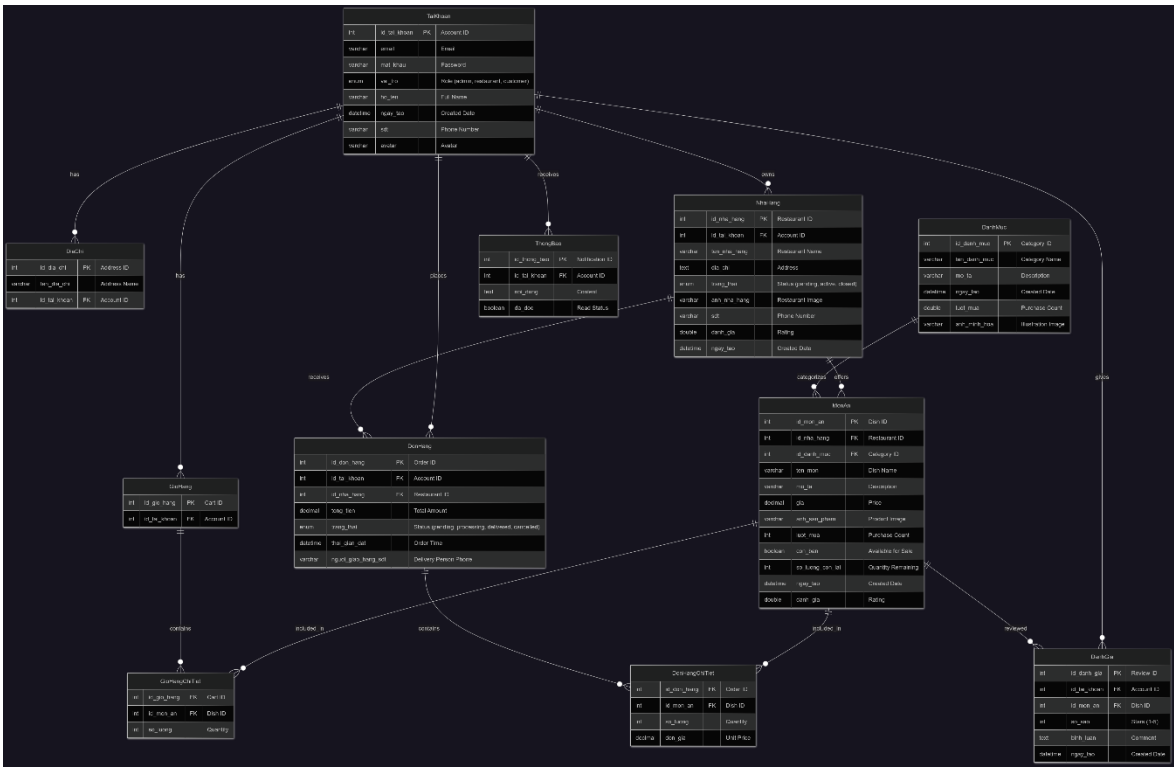
Tân từ: Bảng "Đánh giá" lưu trữ thông tin về các đánh giá của người dùng về món ăn, bao gồm: mã định danh đánh giá (Review ID) là khóa chính, mã tài khoản (Account ID) là khóa ngoại của người đánh giá, mã món ăn (Dish ID) là khóa ngoại của món ăn được đánh giá, số sao (Stars) từ 1 đến 5, bình luận (Comment), và ngày tạo đánh giá (Created Date).

10. Bảng "Thông báo" (Notification)

(Notification ID, Account ID, Content, Read Status)

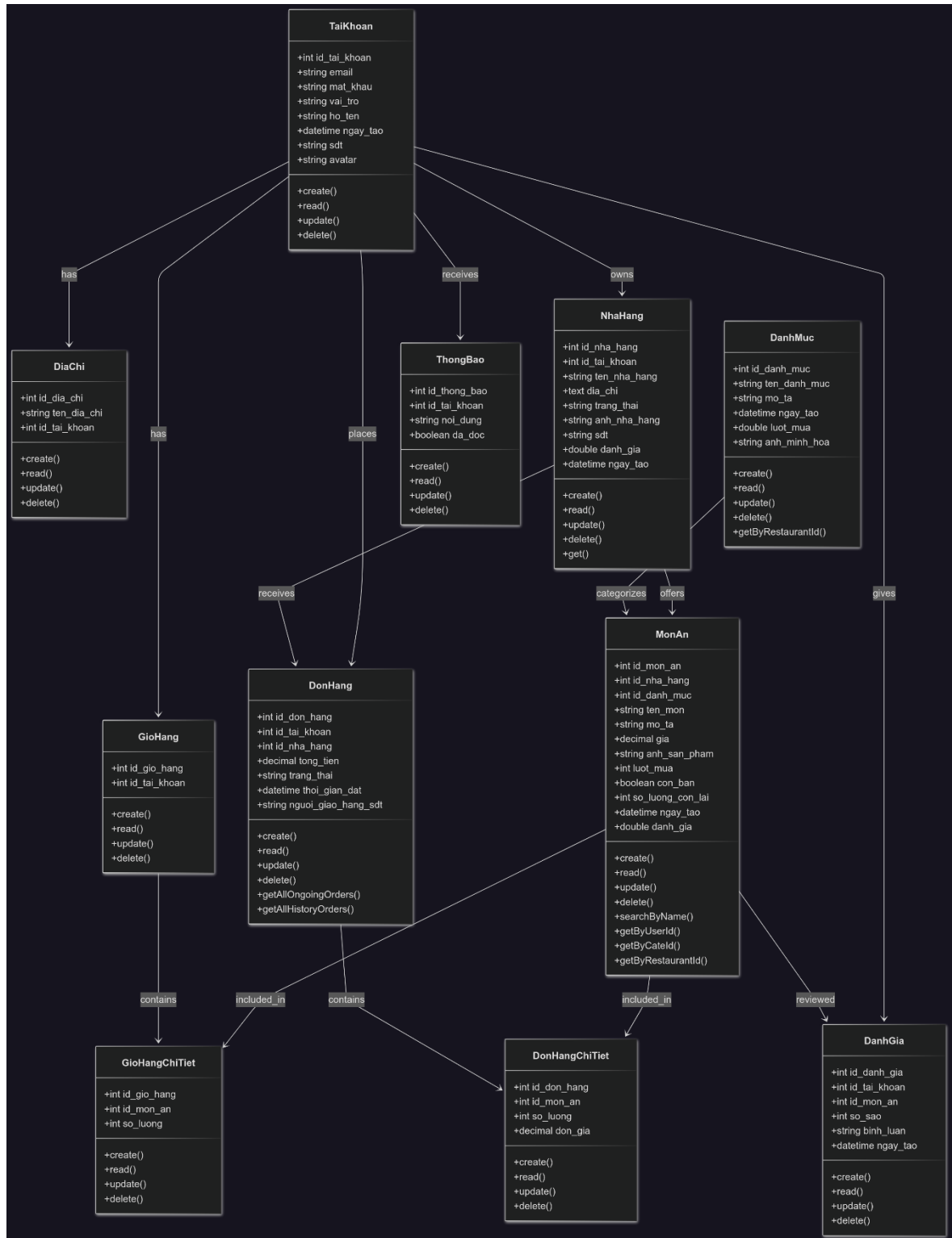
Tân từ: Bảng "Thông báo" lưu trữ thông tin về các thông báo gửi đến người dùng, bao gồm: mã định danh thông báo (Notification ID) là khóa chính, mã tài khoản (Account ID) là khóa ngoại của người nhận thông báo, nội dung thông báo (Content), và trạng thái đã đọc hay chưa (Read Status) là một giá trị boolean.

3.3.2. Sơ đồ dữ liệu



Hình 1: Sơ đồ cơ sở dữ liệu

3.3.3. Sơ đồ UML



Hình 2: Sơ đồ UML

3.3.4. Cấu trúc các bảng dữ liệu trong SQL

Bảng 1:Cấu trúc các bảng

Bảng	Tên cột	Kiểu dữ liệu	Thuộc tính	Mô tả
Account	account_id	UUID	Primary Key	Mã định danh tài khoản
	email	VARCHAR(255)	Not Null, Unique	Địa chỉ email
	password	VARCHAR(255)	Not Null	Mật khẩu
	role	VARCHAR(50)	Not Null	Vai trò (admin, restaurant, customer)
	full_name	VARCHAR(100)	Not Null	Họ tên đầy đủ
	created_date	DATE	Not Null	Ngày tạo tài khoản
	phone_number	VARCHAR(20)	Not Null	Số điện thoại
	avatar	VARCHAR(255)		Đường dẫn ảnh đại diện
	quan_he	VARCHAR(50)		Mối quan hệ (ví dụ: khách hàng, nhân viên)
Address	address_id	UUID	Primary Key	Mã định danh địa chỉ
	address_name	VARCHAR(255)	Not Null	Tên địa chỉ
	account_id	UUID	Foreign Key (Account)	Mã tài khoản (liên kết với Account)

Restaurant	restaurant_id	UUID	Primary Key	Mã định danh nhà hàng
	account_id	UUID	Foreign Key (Account)	Mã tài khoản của chủ nhà hàng
	restaurant_name	VARCHAR(100)	Not Null, Unique	Tên nhà hàng
	address	VARCHAR(255)	Not Null	Địa chỉ chi tiết
	status	VARCHAR(20)	Not Null	Trạng thái (pending, active, closed)
	restaurant_image	VARCHAR(255)		Đường dẫn ảnh nhà hàng
	phone_number	VARCHAR(20)	Not Null	Số điện thoại liên hệ
	rating	DECIMAL(3,2)		Điểm đánh giá (0.00-5.00)
	created_date	DATE	Not Null	Ngày tạo hồ sơ nhà hàng
Category	category_id	UUID	Primary Key	Mã định danh danh mục
	category_name	VARCHAR(100)	Not Null, Unique	Tên danh mục
	description	TEXT		Mô tả chi tiết
	created_date	DATE	Not Null	Ngày tạo danh mục
	purchase_count	INTEGER		Số lượt mua của các món ăn

	category_image	VARCHAR(255)		Đường dẫn ảnh minh họa
Dish	dish_id	UUID	Primary Key	Mã định danh món ăn
	restaurant_id	UUID	Foreign Key (Restaurant)	Mã nhà hàng cung cấp món ăn
	category_id	UUID	Foreign Key (Category)	Mã danh mục của món ăn
	dish_name	VARCHAR(100)	Not Null, Unique	Tên món ăn
	description	TEXT		Mô tả chi tiết
	price	DECIMAL(10,2)	Not Null	Giá của món ăn
	dish_image	VARCHAR(255)		Đường dẫn ảnh sản phẩm
	purchase_count	INTEGER		Số lượt mua của món ăn
	available_for_sale	BOOLEAN	Not Null	Trạng thái còn bán
	quantity_remaining	INTEGER	Not Null	Số lượng còn lại
	created_date	DATE	Not Null	Ngày tạo thông tin món ăn
	rating	DECIMAL(3,2)		Điểm đánh giá (0.00-5.00)
Cart	cart_id	UUID	Primary Key	Mã định danh giỏ hàng
	account_id	UUID	Foreign Key (Account)	Mã tài khoản của khách hàng

Cart Detail	cart_id	UUID	Foreign Key (Cart)	Mã giỏ hàng
	dish_id	UUID	Foreign Key (Dish)	Mã món ăn
	quantity	INTEGER	Not Null	Số lượng của món ăn trong giỏ
Order	order_id	UUID	Primary Key	Mã định danh đơn hàng
	account_id	UUID	Foreign Key (Account)	Mã tài khoản của người đặt hàng
	restaurant_id	UUID	Foreign Key (Restaurant)	Mã nhà hàng nhận đơn
	total_amount	DECIMAL(10,2)	Not Null	Tổng số tiền của đơn hàng
	status	VARCHAR(20)	Not Null	Trạng thái (pending, processing, delivered, cancelled)
	order_time	TIMESTAMP	Not Null	Thời gian đặt hàng
	delivery_person_phone	VARCHAR(20)		Số điện thoại người giao hàng
OrderDetail	order_id	UUID	Foreign Key (Order)	Mã đơn hàng
	dish_id	UUID	Foreign Key (Dish)	Mã món ăn
	quantity	INTEGER	Not Null	Số lượng của món ăn trong đơn

	unit_price	DECIMAL(10,2)	Not Null	Đơn giá tại thời điểm đặt hàng
Review	review_id	UUID	Primary Key	Mã định danh đánh giá
	account_id	UUID	Foreign Key (Account)	Mã tài khoản của người đánh giá
	dish_id	UUID	Foreign Key (Dish)	Mã món ăn được đánh giá
	stars	INTEGER	Not Null, CHECK (stars BETWEEN 1 AND 5)	Số sao (1-5)
	comment	TEXT		Bình luận
	created_date	DATE	Not Null	Ngày tạo đánh giá
Notification	notification_id	UUID	Primary Key	Mã định danh thông báo
	account_id	UUID	Foreign Key (Account)	Mã tài khoản nhận thông báo
	content	TEXT	Not Null	Nội dung thông báo
	read_status	BOOLEAN	Not Null	Trạng thái đã đọc (true/false)

Chương 4. CÀI ĐẶT – TRIỂN KHAI

4.1. Cách tổ chức mã nguồn

Front-end:

-Ứng dụng React Native (Mobile)

- app/: Các màn hình chính (login, cart, search, profile...)
- components/: Các thành phần UI dùng chung
- configs/: Các file cấu hình (axios, env)
- assets/: Ảnh, icon
- hooks/, utils/, contexts/: Tiện ích

-Fodibidy_web/: Ứng dụng React Web admin

- src/components/: Các component quản lý Restaurant, Food, Order, Profile...
- src/utils/, src/config/: Tiện ích, cấu hình
- src/pages/: Trang chính
- Backend:
- src/services/: Các service cho từng entity (user, dish, restaurant, cart, order...)
- src/middlewares/: Xác thực, phân quyền
- src/routes/: Định nghĩa các API endpoint
- src/controllers/: Xử lý business logic

4.2. Tính năng chính đã lập trình

1. Đăng nhập/Đăng ký (Firebase Auth)

1.1. Đăng nhập & xác thực người dùng

- Người dùng nhập email và password trên app/web và nhấn đăng nhập.
- Frontend gửi POST request tới API /api/v1/users/login.
- Backend kiểm tra thông tin bằng Firebase Auth:
- Nếu thành công, trả về Firebase IdToken.
- Nếu thất bại, trả về lỗi.
- Frontend lưu token vào localStorage (web) hoặc AsyncStorage (mobile) để sử dụng cho các request tiếp theo.
- Những request sau gửi token này lên header Authorization để xác thực.

1.2. Đăng ký

- Người dùng nhập thông tin cá nhân → Nhấn "Đăng ký".

- Frontend gửi request lên Backend (/api/v1/users/register).
- Backend đăng ký user trên Firebase Auth, lưu thông tin vào Firestore.
- Trả về token + thông tin user cho frontend.
- Người dùng tự động chuyển sang trạng thái đã đăng nhập.

2. Thêm/Xóa/Sửa dữ liệu cho các entity (user, món ăn, nhà hàng, giỏ hàng, đơn hàng)

2.1. Quản lý người dùng

- Người dùng đăng nhập thành công → Xem/chỉnh sửa thông tin cá nhân.
- Sửa thông tin → gửi request PUT lên backend.
- Backend cập nhật dữ liệu Firestore, trả về kết quả.

2.2. Quản lý nhà hàng

- Nhà hàng đăng nhập web admin.
- Xem, sửa thông tin nhà hàng, quản lý category, menu, đơn hàng.
- Tạo/sửa/xóa món ăn → gửi request đến backend, backend cập nhật Firestore.

2.3. Quản lý món ăn

- Nhân viên/nhà hàng vào dashboard → Thêm món ăn mới (điền tên, giá, ảnh...).
- Gửi request POST lên backend.
- Backend lưu thông tin vào collection Dishes.
- Khi người dùng tìm kiếm/duyet menu, frontend gọi API lấy dữ liệu món ăn.

3. Hiện thị danh sách:

Hiện thị danh sách nhà hàng, món ăn

- Người dùng mở app/web → Frontend gọi API lấy danh sách nhà hàng/món ăn.
- Backend truy vấn Firestore, trả về dữ liệu.
- Người dùng có thể tìm kiếm/lọc, frontend gửi query lên backend để lấy kết quả tương ứng.

4. Quản lý giỏ hàng, đặt hàng, cập nhật trạng thái đơn

4.1. Xử lý giỏ hàng

- Người dùng chọn món → Nhấn "Thêm vào giỏ".

- Frontend gửi request lên backend để thêm vào collection Cart.
- Người dùng vào giỏ hàng (cart), xem danh sách món đã chọn.
- Có thể thay đổi số lượng/xóa món → frontend gửi request cập nhật/xóa.

4.2. Đặt hàng

- Người dùng xác nhận đặt hàng → Nhấn "Đặt hàng".
- Frontend gửi thông tin đơn hàng lên backend.
- Backend tạo mới document Order, Order_details trong Firestore.
- Backend gửi thông báo tới nhà hàng (nếu có).
- Người dùng theo dõi trạng thái đơn hàng (API trả về trạng thái cập nhật).

5. Đánh giá, bình luận .

5.1. Đánh giá món ăn/nhà hàng

- Sau khi hoàn tất đơn, người dùng vào mục đánh giá.
- Frontend gửi review (rating, comment) lên backend.
- Backend lưu vào collection Reviews.
- Nhà hàng nhìn thấy đánh giá trên dashboard.

6. Xác thực & phân quyền các API

- Mọi request đến API cần xác thực đều phải gửi token.
- Middleware backend kiểm tra token, xác định role (CUSTOMER, RESTAURANT, ADMIN).
- Chỉ user đúng role mới thực hiện được các thao tác tương ứng.

4.3. Hình ảnh giao diện thực tế

4.3.1. Danh sách các màn hình

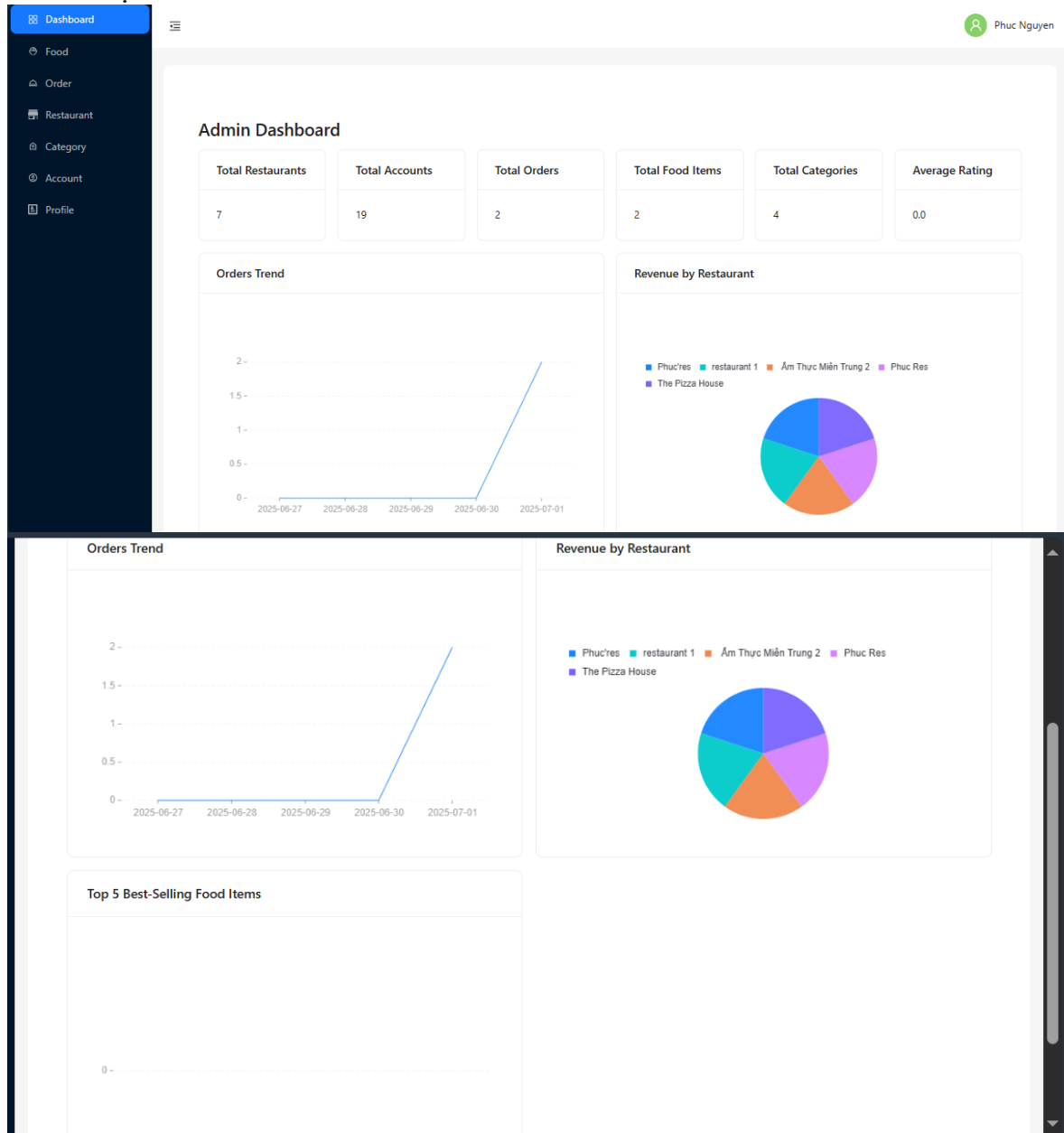
STT	Màn hình	Loại màn hình	Chức năng
1	Màn hình đăng nhập	Nhập liệu	Đăng nhập vào phần mềm
2	Màn hình Trang chủ admin	Màn hình chính	Hiển thị các thông tin chính của trang quản lí
3	Màn hình Trang chủ nhà hàng	Màn hình chính	Hiển thị các thông tin chính của trang nhà hàng
4	Màn hình Trang chủ người dùng	Màn hình chính	Hiển thị các thông tin chính của trang người dùng
5	Màn hình Quản lý món ăn	Nhập liệu/tra cứu	Hiển thị/Nhập các thông tin chính liên quan món ăn
6	Màn hình Thống kê	Tra cứu	Hiển thị thông tin thống kê cho quản lý
7	Màn hình Quản lý các đơn đặt	Tra cứu/Nhập liệu	Hiển thị/Nhập các thông tin liên quan đơn đặt
8	Màn hình Quản lý nhà hàng	Tra cứu/Nhập liệu	Hiển thị/Nhập các thông tin liên quan nhà hàng
9	Màn hình Quản lý danh mục	Tra cứu/Nhập liệu	Hiển thị/Nhập các thông tin liên quan danh mục
10	Màn hình Quản lý tài khoản	Tra cứu/Nhập liệu	Hiển thị/Nhập các thông tin liên quan tài khoản
11	Màn hình Cập nhập thông tin đơn đặt	Nhập liệu	Cập nhập các thông tin của đơn đặt
12	Màn hình Thông tin tài khoản đang đăng nhập	Tra cứu/Nhập liệu	Hiển thị/Cập nhập thông tin tài khoản đang đăng nhập
13	Màn hình Thông tin nhà hàng đang đăng nhập	Tra cứu/Nhập liệu	Hiển thị/Cập nhập thông tin nhà hàng đang đăng nhập
14	Màn hình Thông tin chi tiết món ăn	Tra cứu	Hiển thị thông tin món ăn được chọn
15	Màn hình Thông tin chi tiết nhà hàng	Tra cứu	Hiển thị thông tin nhà hàng được chọn

16	Màn hình Thông tin chi tiết tài khoản	Tra cứu	Hiển thị thông tin tài khoản được chọn
17	Màn hình Thêm mới nhà hàng	Nhập liệu	Nhập vào thông tin nhà hàng để thêm mới
18	Màn hình Thêm mới danh mục	Nhập liệu	Nhập vào thông tin danh mục để thêm mới
19	Màn hình Thêm mới tài khoản	Nhập liệu	Nhập vào thông tin tài khoản để thêm mới
20	Màn hình Cập nhật nhà hàng	Nhập liệu	Nhập vào thông tin nhà hàng để cập nhật
21	Màn hình Cập nhật danh mục	Nhập liệu	Nhập vào thông tin danh mục để cập nhật
22	Màn hình Cập nhật tài khoản	Tra cứu/Nhập liệu	Hiển thị/Nhập vào thông tin tài khoản để cập nhật
23	Màn hình Cập nhật nhà hàng của tài khoản đang đăng nhập	Tra cứu/Nhập liệu	Hiển thị/Nhập vào thông tin nhà hàng để cập nhật
24	Màn hình Tìm kiếm món ăn	Tra cứu	Tìm kiếm các món ăn theo tên
25	Màn hình Đặt món	Tra cứu/Nhập liệu	Chọn các món từ giỏ hàng và số lượng để bắt đầu đặt món
26	Màn hình Hồ sơ	Tra cứu/Nhập liệu	Cập nhật/ hiển thị thông tin người dùng
27	Màn hình đăng ký/đăng nhập	Nhập liệu	Đăng nhập hoặc tạo tài khoản mới
28	Màn hình Trang chủ ứng dụng	Tra cứu	Màn hình chính của ứng dụng cho người dùng
29	Màn hình Các đơn đang đặt	Tra cứu	Hiển thị các đơn đã/đang đặt của người dùng
30	Màn hình Quản lý đơn đặt	Tra cứu	Hiển thị các đơn đặt của nhà hàng đang đăng nhập quản lý
31	Màn hình Cập nhật đơn đặt	Tra cứu/Nhập liệu	Cập nhật các thông tin đơn hàng

4.3.2. Mô tả các màn hình:

4.3.2.1. Màn hình trang chủ admin và thống kê

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	tabDashboard	TabPage	Truy cập màn hình tổng quan (dashboard)

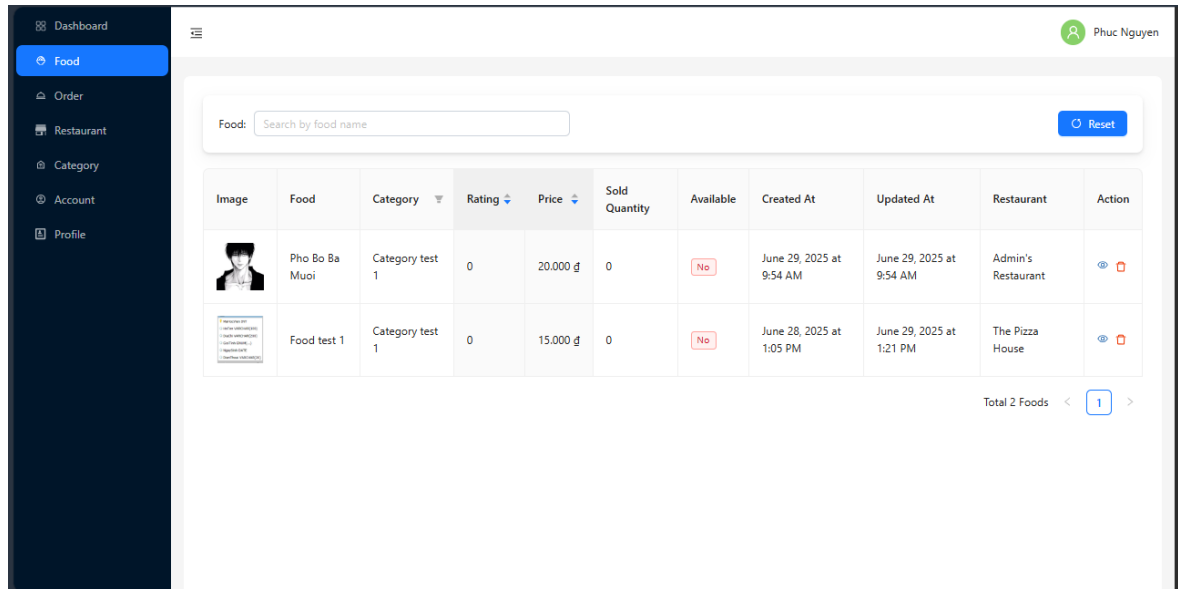
2	tabFood	TabPage	Hiển thị và quản lý danh sách món ăn
3	tabOrder	TabPage	Hiển thị và quản lý danh sách đơn hàng
4	tabRestaurant	TabPage	Hiển thị và quản lý danh sách nhà hàng
5	tabCategory	TabPage	Hiển thị và quản lý các danh mục món ăn
6	tabAccount	TabPage	Hiển thị và quản lý tài khoản người dùng
7	tabProfile	TabPage	Hiển thị thông tin và chỉnh sửa tài khoản admin
8	txtTotalRestaurants	TextBox/Label	Hiển thị tổng số nhà hàng trong hệ thống
9	txtTotalAccounts	TextBox/Label	Hiển thị tổng số tài khoản người dùng
10	txtTotalOrders	TextBox/Label	Hiển thị tổng số đơn hàng đã tạo
11	txtTotalFoodItems	TextBox/Label	Hiển thị tổng số món ăn
12	txtTotalCategories	TextBox/Label	Hiển thị tổng số danh mục món ăn
13	txtAverageRating	TextBox/Label	Hiển thị điểm đánh giá trung bình toàn hệ thống
14	chartOrdersTrend	LineChart	Biểu đồ xu hướng số lượng đơn hàng theo thời gian
15	chartRevenueByRes	PieChart	Biểu đồ tròn thể hiện doanh thu theo từng nhà hàng
16	userAvatar	Icon	Hiển thị icon avatar người dùng đang đăng nhập (góc phải phía trên cùng)
17	lblUserName	Label	Hiển thị tên người dùng đang đăng nhập

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Khi bấm vào tabDashboard	Hiển thị màn hình tổng quan (dashboard), cập nhật các chỉ số thống kê và biểu đồ
2	Khi bấm vào tabFood	Hiển thị và quản lý danh sách món ăn trong hệ thống
3	Khi bấm vào tabOrder	Hiển thị và quản lý danh sách đơn hàng
4	Khi bấm vào tabRestaurant	Hiển thị và quản lý danh sách nhà hàng
5	Khi bấm vào tabCategory	Hiển thị và quản lý các danh mục món ăn
6	Khi bấm vào tabAccount	Hiển thị và quản lý tài khoản người dùng
7	Khi bấm vào tabProfile	Hiển thị thông tin và cho phép chỉnh sửa tài khoản admin
8	Khi tải lại trang dashboard	Các chỉ số thống kê (nhà hàng, tài khoản, đơn hàng, món ăn, danh mục, rating) tự động cập nhật dữ liệu mới nhất
9	Khi có đơn hàng mới được tạo	Biểu đồ Orders Trend và các chỉ số sẽ cập nhật realtime số lượng đơn hàng
10	Khi có doanh thu mới theo nhà hàng	Biểu đồ Revenue by Restaurant cập nhật phân bổ doanh thu cho từng nhà hàng
11	Khi admin bấm vào tên người dùng	Hiển thị menu hoặc chuyển đến trang thông tin tài khoản admin

4.3.2.2. Màn hình Quản lý món ăn

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtSearchFood	TextBox	Nhập tên món ăn để tìm kiếm nhanh trong danh sách
2	btnReset	Button	Đặt lại bộ lọc/tìm kiếm, hiển thị toàn bộ danh sách
3	tblFoodList	DataTable	Bảng hiển thị danh sách các món ăn với nhiều cột thông tin
4	colImage	Image	Hiển thị hình ảnh minh họa của món ăn
5	colFoodName	Text	Tên món ăn
6	colCategory	Text	Danh mục món ăn
7	colRating	Number	Điểm đánh giá trung bình của món
8	colPrice	Number/Text	Giá bán món ăn
9	colSoldQuantity	Number	Số lượng đã bán
10	colAvailable	Label	Trạng thái còn hàng ("No" nếu hết hàng)

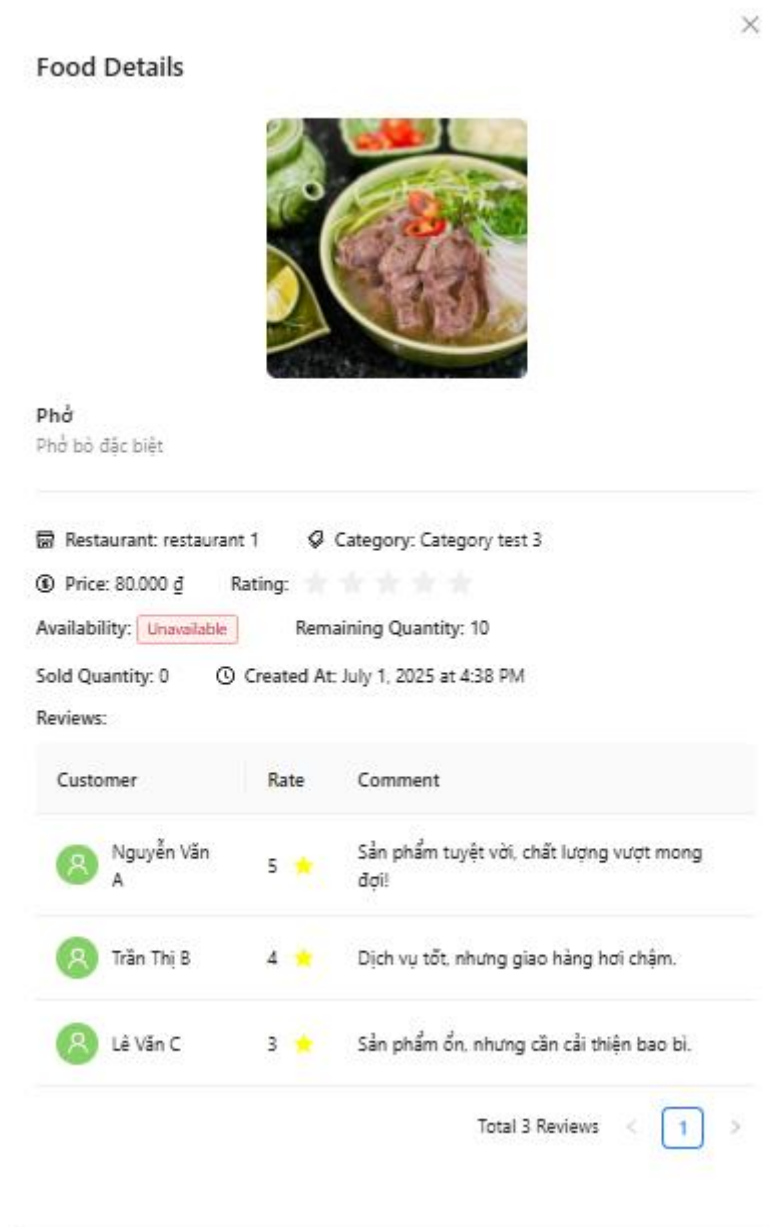
11	colCreatedAt	DateTime	Thời điểm tạo món ăn
12	colUpdatedAt	DateTime	Thời điểm cập nhật gần nhất
13	colRestaurant	Text	Tên nhà hàng sở hữu món ăn
14	colActionView	Icon Button	Nút xem chi tiết món ăn (icon con mắt)
15	colActionDelete	Icon Button	Nút xóa món ăn (icon thùng rác)
16	lblTotalFoods	Label	Tổng số món ăn đang hiển thị
17	pagination	Pagination	Điều hướng trang khi có nhiều món ăn

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập tên món vào txtSearchFood	Lọc danh sách món ăn theo tên đã nhập
2	Bấm nút btnReset	Xóa bộ lọc tìm kiếm, hiển thị lại toàn bộ danh sách món ăn
3	Bấm icon con mắt (colActionView)	Hiển thị chi tiết thông tin món ăn
4	Bấm icon thùng rác (colActionDelete)	Hiện hộp thoại xác nhận xóa, nếu đồng ý thì xóa món ăn khỏi hệ thống
5	Bấm vào tiêu đề các cột bảng	Sắp xếp danh sách món ăn theo thứ tự tăng/giảm của cột đó (nếu có hỗ trợ)
6	Chuyển trang trên pagination	Hiển thị món ăn ở trang tiếp theo/trước đó

4.3.2.3. Màn hình Thông tin chi tiết món ăn

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	imgFood	Image	Hiển thị hình ảnh của món ăn
2	lblFoodName	Label	Tên món ăn

3	lblFoodDescription	Label	Mô tả món ăn (trong ví dụ là “nothing”)
4	lblRestaurant	Label	Tên nhà hàng sở hữu món ăn
5	lblCategory	Label	Danh mục món ăn
6	lblPrice	Label	Giá món ăn
7	ratingFood	RatingStars	Điểm đánh giá trung bình (hiện tại chưa có rating tổng hợp)
8	lblAvailability	Label	Trạng thái còn hàng/hết hàng ("Unavailable" màu đỏ nếu hết hàng)
9	lblRemainingQty	Label	Số lượng còn lại của món ăn
10	lblSoldQty	Label	Số lượng đã bán
11	lblCreatedAt	Label	Thời điểm tạo món ăn
12	tblReviews	DataTable	Bảng hiển thị các đánh giá từ khách hàng
13	colCustomer	Text	Tên khách hàng đánh giá (có avatar hình tròn)
14	colRate	Number/Stars	Số sao đánh giá của khách hàng
15	colComment	Text	Nội dung bình luận của khách hàng
16	lblTotalReviews	Label	Tổng số lượng đánh giá
17	paginationReviews	Pagination	Điều hướng trang khi có nhiều đánh giá

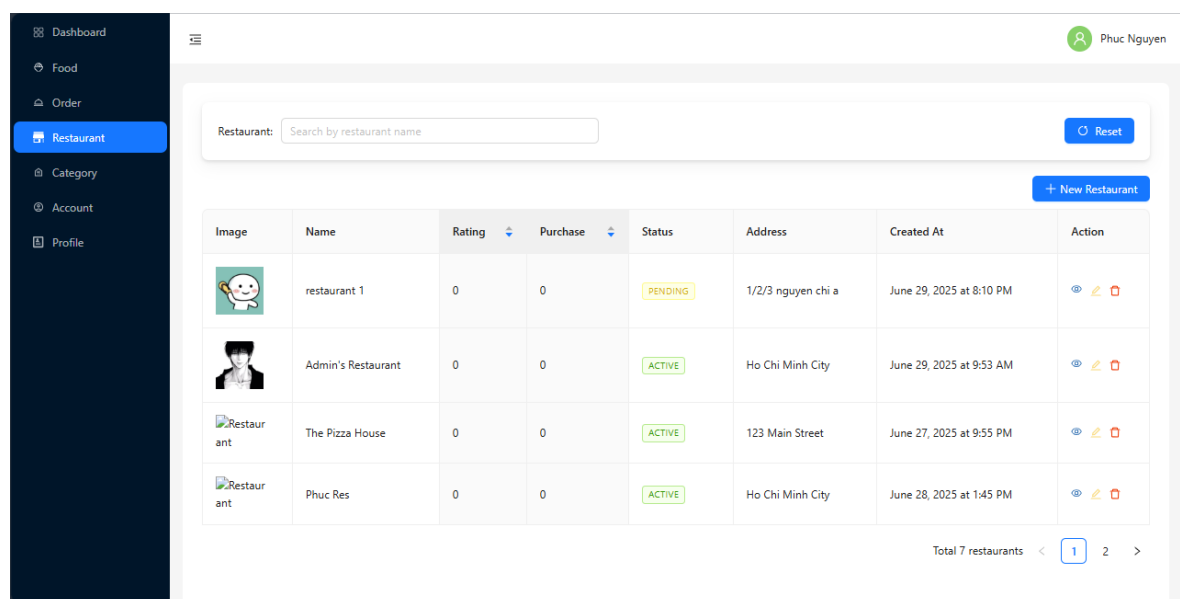
c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Bấm nút đóng (X góc trên bên phải)	Đóng cửa sổ chi tiết món ăn, quay lại màn hình trước đó

2	Chuyển trang trên pagination	Hiện thị các đánh giá ở trang tiếp theo/trước đó của bảng đánh giá
3	Khi dữ liệu món ăn cập nhật	Thông tin về trạng thái, giá, số lượng, đánh giá sẽ tự động cập nhật trên màn hình
4	Khi có đánh giá mới	Bảng reviews cập nhật thêm dòng mới về khách hàng, số sao và nội dung bình luận

4.3.2.4. Màn hình Quản lý nhà hàng

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Ràng buộc	Chức năng
1	txtSearchRestaurant	TextBox		Nhập tên nhà hàng để tìm kiếm nhanh trong danh sách
2	btnReset	Button		Đặt lại bộ lọc/tìm kiếm, hiển thị toàn bộ danh sách
3	btnNewRestaurant	Button		Thêm mới một nhà hàng
4	tblRestaurantList	DataTable		Bảng hiển thị danh sách các nhà hàng với nhiều cột thông tin

5	colImage	Image		Hiển thị hình ảnh đại diện của nhà hàng
6	colName	Text		Tên nhà hàng
7	colRating	Number		Điểm đánh giá trung bình của nhà hàng
8	colPurchase	Number		Số đơn hàng đã bán của nhà hàng
9	colStatus	Label		Trạng thái hoạt động: PENDING (chờ duyệt, màu vàng), ACTIVE (đang hoạt động, màu xanh lá cây)
10	colAddress	Text		Địa chỉ nhà hàng
11	colCreatedAt	DateTime		Thời điểm tạo nhà hàng
12	colActionView	Icon Button		Nút xem chi tiết nhà hàng (icon con mắt)
13	colActionEdit	Icon Button		Nút chỉnh sửa thông tin nhà hàng (icon cây bút)
14	colActionDelete	Icon Button		Nút xóa nhà hàng (icon thùng rác)
15	lblTotalRestaurants	Label		Hiển thị tổng số nhà hàng đang có trong hệ thống
16	pagination	Pagination		Điều hướng trang khi có nhiều nhà hàng

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập tên nhà hàng vào txtSearchRestaurant	Lọc danh sách nhà hàng theo tên đã nhập

2	Bấm nút btnReset	Xóa bộ lọc tìm kiếm, hiển thị lại toàn bộ danh sách nhà hàng
3	Bấm nút btnNewRestaurant	Mở form thêm mới nhà hàng
4	Bấm icon con mắt (colActionView)	Hiển thị chi tiết thông tin nhà hàng
5	Bấm icon cây bút (colActionEdit)	Mở form chỉnh sửa thông tin nhà hàng
6	Bấm icon thùng rác (colActionDelete)	Hiện hộp thoại xác nhận xóa, nếu đồng ý thì xóa nhà hàng khỏi hệ thống
7	Bấm vào tiêu đề các cột bảng	Sắp xếp danh sách nhà hàng theo thứ tự tăng/giảm của cột đó (nếu có hỗ trợ)
8	Chuyển trang trên pagination	Hiển thị nhà hàng ở trang tiếp theo/trước đó

4.3.2.5. Màn hình Thêm mới nhà hàng

a. Giao diện:

Create Restaurant

✕

* Email

* Restaurant name

* Full name

* Address

* Password

* Restaurant Phone

* Bio

Cancel

Create

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Ràng buộc	Chức năng
-----	-----	------	-----------	-----------

1	txtEmail	TextBox	Bắt buộc	Nhập email tài khoản nhà hàng
2	txtRestaurantName	TextBox	Bắt buộc	Nhập tên nhà hàng
3	txtFullName	TextBox	Bắt buộc	Nhập họ tên chủ nhà hàng hoặc người đại diện
4	txtAddress	TextBox	Bắt buộc	Nhập địa chỉ nhà hàng
5	txtPassword	PasswordBox	Bắt buộc, có ẩn/hiện	Nhập mật khẩu cho tài khoản nhà hàng
6	txtRestaurantPhone	TextBox	Bắt buộc	Nhập số điện thoại liên hệ của nhà hàng
7	txtBio	TextArea	Tùy chọn (có thể bỏ trống)	Nhập mô tả ngắn về nhà hàng
8	btnCancel	Button		Hủy thao tác, đóng cửa sổ tạo mới
9	btnCreate	Button		Tạo mới nhà hàng với các thông tin đã nhập
10	btnClose	Icon Button		Đóng cửa sổ tạo mới (góc trên bên phải)

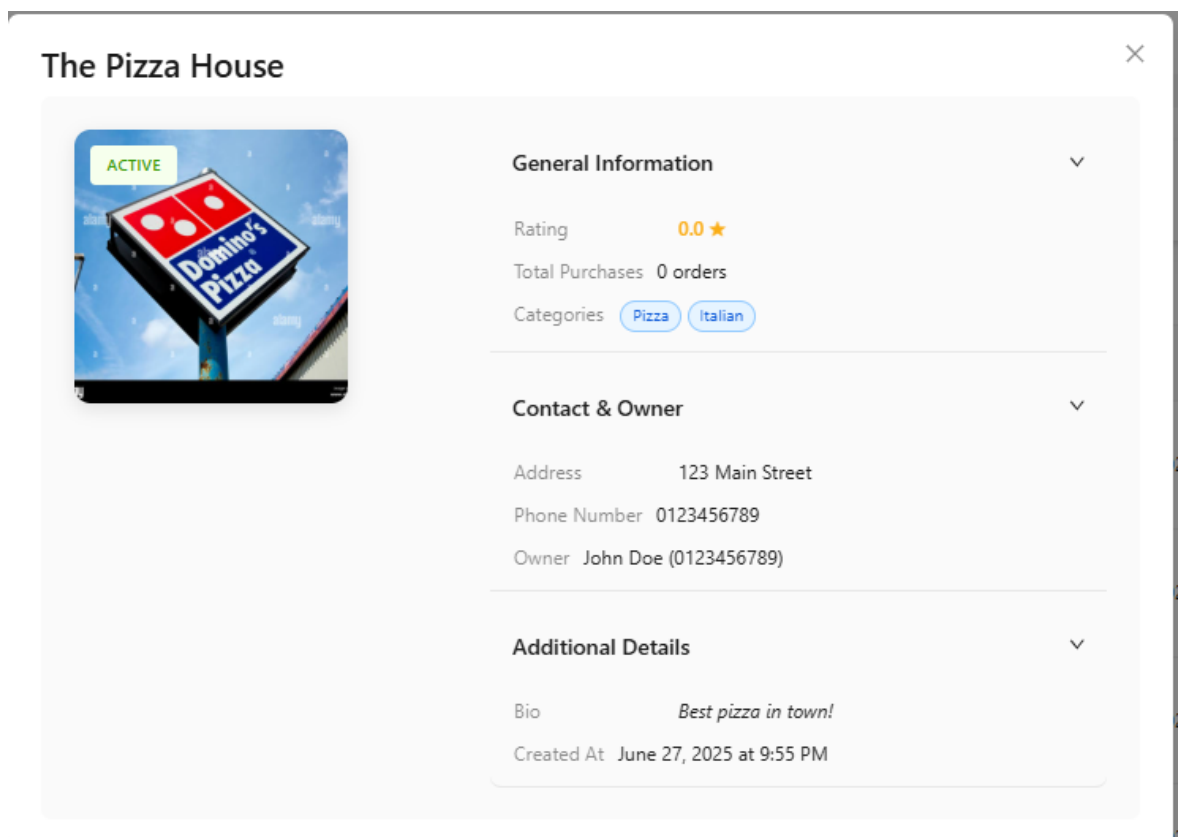
c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập thông tin vào các ô TextBox	Cập nhật dữ liệu cho từng trường tương ứng trong form
2	Bấm nút btnCreate	Kiểm tra hợp lệ các trường bắt buộc, nếu hợp lệ thì gửi yêu cầu tạo mới nhà hàng
3	Bấm nút btnCancel	Đóng form tạo mới nhà hàng, không lưu dữ liệu

4	Bấm nút btnClose (X)	Đóng form tạo mới nhà hàng, không lưu dữ liệu
5	Bấm icon ẩn/hiện mật khẩu	Chuyển đổi giữa trạng thái hiển thị hoặc ẩn ký tự trong trường mật khẩu
6	Nhập thiếu trường bắt buộc	Hiển thị cảnh báo yêu cầu nhập đủ các thông tin cần thiết trước khi tạo mới

4.3.2.6. Màn hình Thông tin chi tiết nhà hàng

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Ràng buộc	Chức năng
1	lblRestaurantName	Label	Bắt buộc	Hiển thị tên nhà hàng
2	imgRestaurantImage	Image	Có hoặc không (ảnh mặc định)	Hiển thị ảnh đại diện nhà hàng

3	lblStatus	Label/Badge	Bắt buộc	Hiển thị trạng thái nhà hàng (ví dụ: ACTIVE)
4	lblRating	Label	Bắt buộc	Hiển thị đánh giá trung bình
5	lblTotalOrders	Label	Bắt buộc	Hiển thị tổng số đơn hàng đã nhận
6	lblCategories	Tag/List	Tùy chọn	Hiển thị danh sách danh mục món ăn
7	lblAddress	Label	Bắt buộc	Hiển thị địa chỉ nhà hàng
8	lblPhoneNumber	Label	Bắt buộc	Hiển thị số điện thoại
9	lblOwner	Label	Bắt buộc	Hiển thị tên chủ nhà hàng và số điện thoại
10	lblBio	Label/Italic	Tùy chọn	Hiển thị mô tả ngắn gọn về nhà hàng
11	lblCreatedAt	Label	Bắt buộc	Hiển thị ngày tạo nhà hàng
12	btnClose	Icon Button	Bắt buộc	Đóng cửa sổ xem thông tin

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Click vào nút btnClose (X)	Đóng cửa sổ popup hiển thị thông tin nhà hàng
2	Hover vào rating hoặc orders	(Tùy chọn) Hiển thị tooltip mô tả thêm nếu được hỗ trợ
3	Hiển thị dữ liệu	Hệ thống tải và hiển thị thông tin chi tiết của nhà hàng từ cơ sở dữ liệu
4	Trạng thái không phải ACTIVE	Hiển thị màu trạng thái khác (ví dụ: màu xám với CLOSED)
5	Không có bio hoặc orders	Hiển thị nội dung mặc định như <i>No description available</i> hoặc <i>0 orders</i>

4.3.2.7. Màn hình Cập nhật thông tin nhà hàng

a. Giao diện:


Update Restaurant

X

ID

VH20h780T4g25U95N3jF

Restaurant image



* Restaurant name

The Pizza House

* Address

123 Main Street

* Restaurant Phone

0123456789

* Bio

Best pizza in town!

Cancel

Update

b. Mô tả các đối tượng trên màn hình:

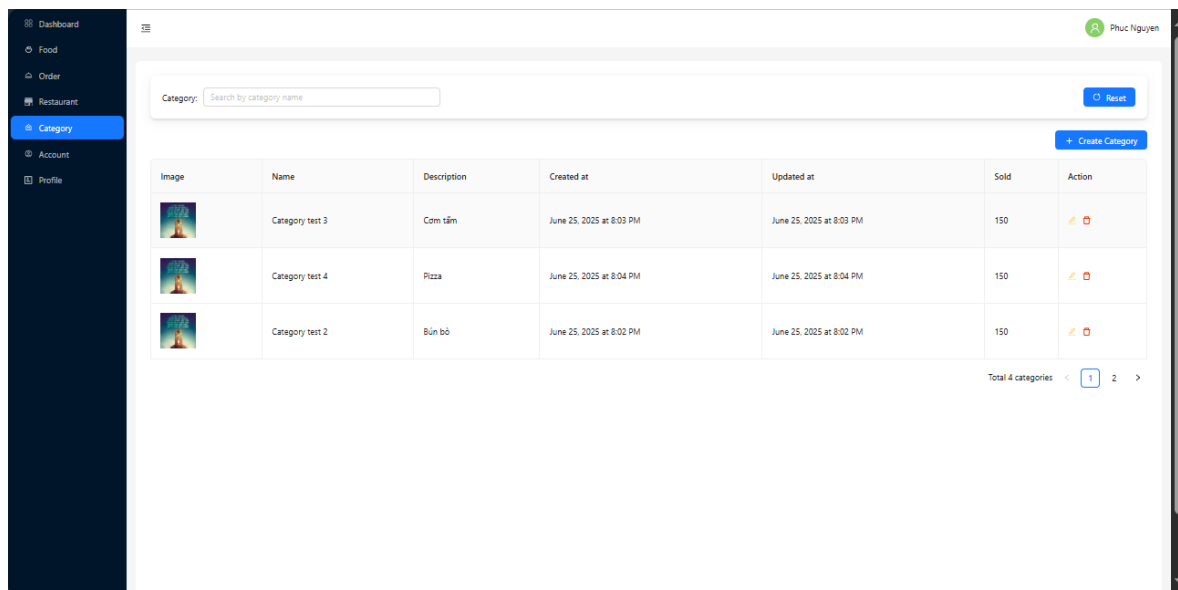
STT	Tên	Kiểu	Chức năng
1	txtId	TextBox	Hiển thị ID định danh của nhà hàng
2	imgRestaurantImage	Image Upload	Cho phép thay đổi hình ảnh đại diện nhà hàng
3	txtRestaurantName	TextBox	Nhập tên nhà hàng
4	txtAddress	TextBox	Nhập địa chỉ của nhà hàng
5	txtRestaurantPhone	TextBox	Nhập số điện thoại liên hệ của nhà hàng
6	txtBio	TextArea	Nhập mô tả ngắn về nhà hàng
7	btnCancel	Button	Hủy thao tác cập nhật và đóng form
8	btnUpdate	Button	Gửi thông tin cập nhật lên server
9	btnClose	Icon Button	Đóng popup cập nhật (góc trên phải)

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Bấm btnUpdate	Kiểm tra các trường bắt buộc. Nếu hợp lệ thì gửi yêu cầu cập nhật lên backend
2	Bấm btnCancel	Đóng form cập nhật mà không lưu thay đổi
3	Bấm btnClose (dấu "x" góc phải)	Đóng form cập nhật mà không lưu thay đổi
4	Thay ảnh bằng imgRestaurantImage	Cho phép người dùng tải ảnh mới lên để cập nhật hình đại diện
5	Nhập thiếu các trường bắt buộc	Hiện thị thông báo lỗi (có thể bôi đỏ và báo "This field is required")
6	Thay đổi nội dung ở các TextBox/TextArea	Cập nhật giá trị vào state nội bộ hoặc formData

4.3.2.8. Màn hình Quản lý danh mục

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtSearchCategory	TextBox	Nhập từ khóa để tìm kiếm danh mục theo tên
2	btnReset	Button	Xóa nội dung tìm kiếm và hiển thị toàn bộ danh mục
3	btnCreateCategory	Button	Mở form để tạo danh mục mới
4	tblCategory	Table	Hiện thị danh sách các danh mục với các cột thông tin

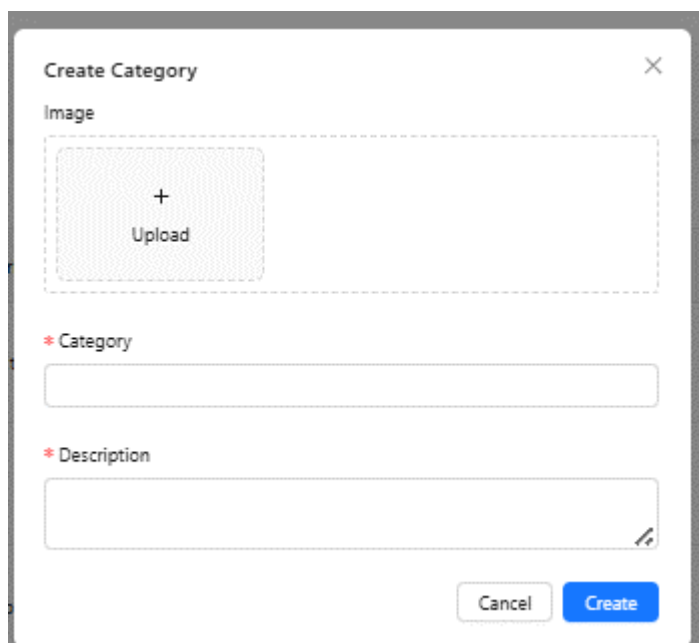
5	btnEdit	Icon Button	Chỉnh sửa danh mục tương ứng
6	btnDelete	Icon Button	Xóa danh mục tương ứng
7	paginationControl	Pagination	Điều hướng phân trang giữa các danh mục

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập từ khóa trong txtSearchCategory	Lọc danh sách danh mục theo tên phù hợp
2	Bấm btnReset	Xóa từ khóa tìm kiếm, tải lại toàn bộ danh mục
3	Bấm btnCreateCategory	Mở popup form để tạo danh mục mới
4	Bấm btnEdit trên từng dòng	Mở popup để chỉnh sửa thông tin danh mục
5	Bấm btnDelete trên từng dòng	Hiện thị xác nhận, nếu đồng ý thì xóa danh mục
6	Điều hướng qua paginationControl	Chuyển trang và hiển thị danh sách danh mục tương ứng
7	Dữ liệu thay đổi sau thêm/sửa/xóa	Tự động làm mới bảng dữ liệu (reload)

4.3.2.9. Màn hình Thêm mới danh mục

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

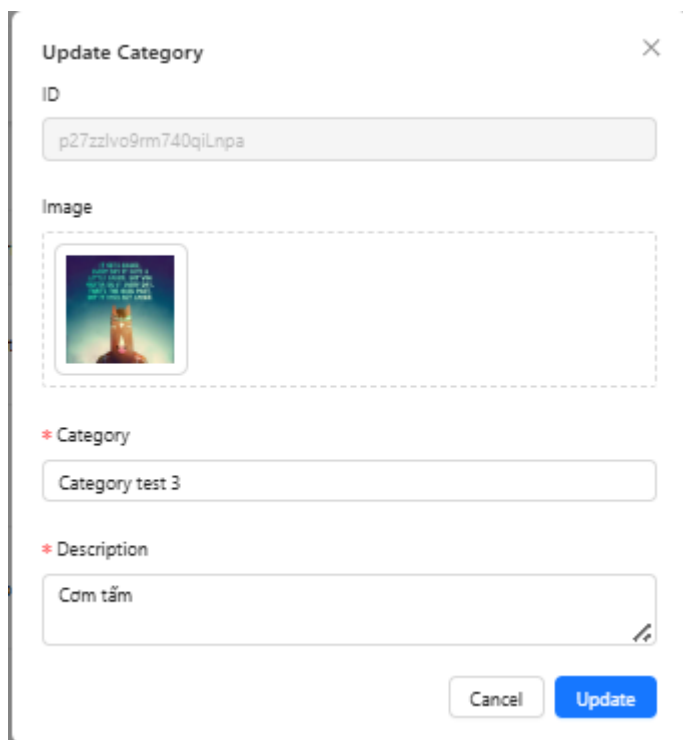
STT	Tên	Kiểu	Chức năng/Chú thích
1	txtCustomerEmail	TextBox	Ô nhập email khách hàng
2	txtCustomerName	TextBox	Ô nhập tên khách hàng
3	txtCustomerPhone	TextBox	Ô nhập số điện thoại khách hàng
4	txtCustomerAddress	TextBox	Ô nhập địa chỉ khách hàng
5	txtCustomerPassword	PasswordBox	Ô nhập mật khẩu khách hàng
6	txtNote	TextArea	Ô nhập ghi chú về khách hàng
7	btnCancel	Button	Nút hủy thao tác, đóng cửa sổ tạo mới.
8	btnCreate	Button	Nút tạo mới khách hàng với thông tin đã nhập.
9	btnClose	Icon Button	Nút đóng (hình "X") ở góc trên bên phải để đóng cửa sổ.

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập thông tin vào các ô TextBox	Cập nhật dữ liệu ngay lập tức vào trường tương ứng trong form (ví dụ: nhập " <u>khach@example.com</u> " vào txtCustomerEmail).
2	Bấm nút btnCreate	Kiểm tra hợp lệ các trường bắt buộc (txtCustomerEmail, txtCustomerName, txtCustomerPhone, txtCustomerAddress, txtCustomerPassword). Nếu hợp lệ (không rỗng, email hợp lệ, số điện thoại 10 chữ số), gửi yêu cầu tạo mới khách hàng đến server. Nếu không hợp lệ, hiển thị thông báo lỗi (ví dụ: "Vui lòng nhập đầy đủ thông tin").
3	Bấm nút btnCancel	Đóng form tạo mới ngay lập tức, không lưu dữ liệu đã nhập, quay về màn hình trước đó.
4	Bấm nút btnClose (X)	Đóng form tạo mới ngay lập tức, không lưu dữ liệu, tương tự btnCancel.
5	Bấm icon ẩn/hiện mật khẩu	Chuyển đổi trạng thái hiển thị của txtCustomerPassword:

4.3.2.10. Màn hình Cập nhập danh mục

a. Giao diện:



The screenshot shows a modal window titled "Update Category" with a close button (X) in the top right corner. Inside the window, there are several input fields and buttons:

- ID:** A text box containing the value "p27zzlvo9rm740qiLnpa".
- Image:** A section with a dashed border containing a small image preview of a lighthouse.
- * Category:** A text box containing the value "Category test 3".
- * Description:** A text area containing the value "Cơm tấm".
- Buttons:** "Cancel" and "Update" buttons are located at the bottom right.

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtCategoryId	TextBox (disabled)	Hiển thị ID của danh mục (không chỉnh sửa được)
2	imgPreview	Image Box	Hiển thị hình ảnh minh họa hiện tại của danh mục
3	uploadImageArea	Drag & Drop	Cho phép thay đổi ảnh danh mục bằng cách kéo thả hoặc chọn file
4	txtCategoryName	TextBox	Nhập tên mới hoặc chỉnh sửa tên danh mục
5	txtDescription	TextArea	Nhập mô tả cho danh mục
6	btnCancel	Button	Hủy thao tác cập nhật và đóng cửa sổ
7	btnUpdate	Button	Cập nhật danh mục với thông tin mới

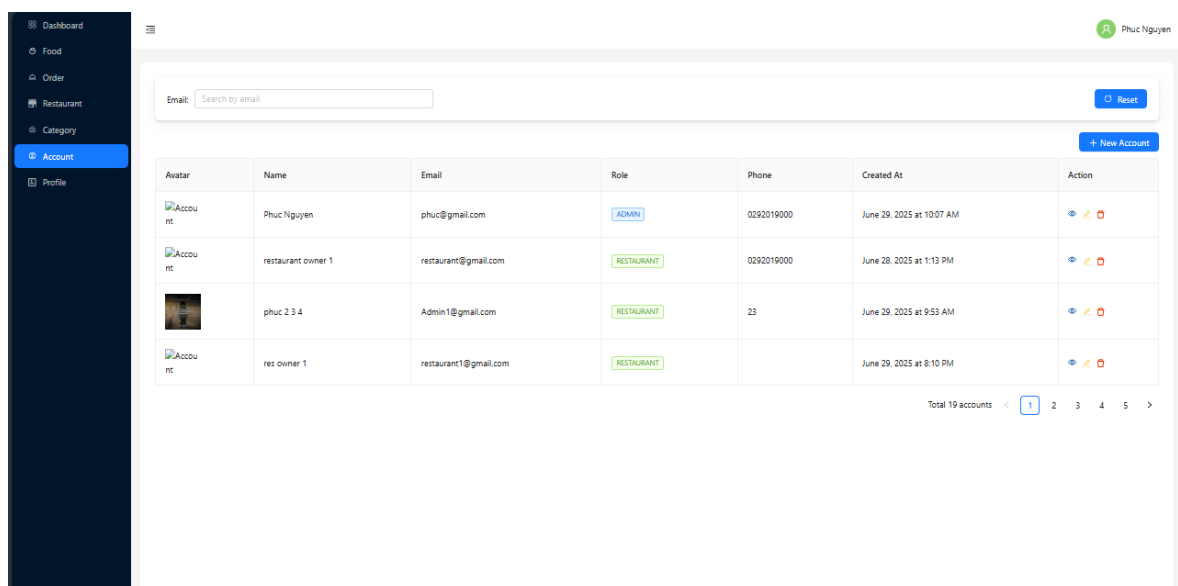
c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
-----	---------	-------

1	Thay đổi ảnh trong uploadImageArea	Cập nhật ảnh xem trước (imgPreview) và lưu ảnh mới
2	Nhập dữ liệu vào txtCategoryName hoặc txtDescription	Cập nhật thông tin danh mục trong form
3	Bấm btnUpdate	Kiểm tra hợp lệ các trường bắt buộc, nếu hợp lệ thì gửi yêu cầu cập nhật danh mục
4	Bấm btnCancel	Đóng popup mà không lưu thay đổi
5	Thiếu trường bắt buộc	Hiển thị cảnh báo yêu cầu nhập đầy đủ thông tin

4.3.2.11. Màn hình Quản lý tài khoản người dùng

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtSearchEmail	TextBox	Nhập email để tìm kiếm tài khoản
2	btnReset	Button	Xóa nội dung tìm kiếm và tải lại danh sách
3	btnCreateAccount	Button	Mở form tạo mới tài khoản
4	tblAccount	Table	Hiển thị danh sách tài khoản (avatar, tên, email, vai trò, sdt, ngày tạo, hành động)
5	btnView	Icon Button	Xem chi tiết tài khoản

6	btnEdit	Icon Button	Sửa thông tin tài khoản
7	btnDelete	Icon Button	Xóa tài khoản
8	paginationControl	Pagination	Phân trang danh sách tài khoản

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập email vào txtSearchEmail	Lọc danh sách tài khoản theo email phù hợp
2	Bấm btnReset	Xóa dữ liệu tìm kiếm và hiển thị toàn bộ danh sách tài khoản
3	Bấm btnCreateAccount	Mở popup tạo mới tài khoản
4	Bấm icon btnView	Hiển thị chi tiết thông tin tài khoản đã chọn
5	Bấm icon btnEdit	Mở popup chỉnh sửa thông tin tài khoản
6	Bấm icon btnDelete	Hiển thị xác nhận xóa và thực hiện xóa tài khoản nếu đồng ý
7	Bấm vào phân trang	Chuyển sang trang kế tiếp/trước để xem thêm tài khoản

4.3.2.12. Màn hình Thêm mới tài khoản người dùng

a. Giao diện:

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtEmail	TextBox	Nhập địa chỉ email cho tài khoản
2	txtFullName	TextBox	Nhập họ và tên người dùng
3	txtPhone	TextBox	Nhập số điện thoại người dùng
4	cmbRole	Dropdown	Chọn vai trò người dùng (ADMIN / RESTAURANT / ...)
5	txtPassword	PasswordBox	Nhập mật khẩu tài khoản mới
6	uploadAvatar	Upload Box	Tải lên ảnh đại diện cho tài khoản
7	btnCreate	Button	Gửi thông tin để tạo tài khoản
8	btnCancel	Button	Hủy thao tác tạo mới và đóng cửa sổ
9	btnTogglePwd	Icon Button	Hiện/ẩn mật khẩu đã nhập trong txtPassword

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập thông tin vào các ô nhập	Cập nhật dữ liệu tương ứng cho từng trường trong form
2	Chọn ảnh đại diện	Hiển thị ảnh preview và lưu ảnh đã chọn tạm thời
3	Bấm Create	Kiểm tra hợp lệ các trường, nếu hợp lệ thì gọi API tạo mới tài khoản
4	Bấm Cancel	Đóng form tạo tài khoản, không lưu dữ liệu
5	Bấm icon ẩn/hiện mật khẩu	Chuyển đổi giữa hiển thị và ẩn ký tự của ô Password
6	Bỏ trống trường bắt buộc	Hiển thị cảnh báo yêu cầu nhập đủ thông tin cần thiết

4.3.2.13. Màn hình Thông tin tài khoản người dùng

a. Giao diện:

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng/Chú thích
1	txtEmail	TextBox	Ô nhập email tài khoản nhà hàng (ví dụ: <u>nhahang@example.com</u>).
2	txtRestaurantName	TextBox	Ô nhập tên nhà hàng (ví dụ: "Nhà hàng Phở 24").
3	txtFullName	TextBox	Ô nhập họ tên chủ nhà hàng (ví dụ: "Nguyễn Văn A").
4	txtAddress	TextBox	Ô nhập địa chỉ nhà hàng (ví dụ: "123 Đường Lê Lợi").
5	txtPassword	PasswordBox	Ô nhập mật khẩu (hiển thị dấu chấm khi ẩn).
6	txtRestaurantPhone	TextBox	Ô nhập số điện thoại (ví dụ: "0905123456").
7	txtBio	TextArea	Ô nhập mô tả nhà hàng (ví dụ: "Chuyên phở bò ngon").
8	btnCancel	Button	Nút hủy thao tác, đóng cửa sổ cập nhật.
9	btnUpdate	Button	Nút cập nhật thông tin nhà hàng.
10	btnClose	Icon Button	Nút đóng (hình "X") ở góc trên bên phải.

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập thông tin vào các ô TextBox	Cập nhật dữ liệu ngay lập tức vào trường tương ứng trong form (ví dụ: sửa "Nhà hàng Phở 24" thành "Phở 24 Deluxe").
2	Bấm nút btnUpdate	Kiểm tra hợp lệ các trường bắt buộc (txtEmail, txtRestaurantName, txtFullName, txtAddress, txtPassword, txtRestaurantPhone). Nếu hợp lệ (không rỗng, email hợp lệ, số điện thoại 10 chữ số), gửi yêu cầu cập nhật thông tin nhà hàng đến server. Nếu không hợp lệ, hiển thị thông báo lỗi (ví dụ: "Vui lòng nhập đầy đủ thông tin").
3	Bấm nút btnCancel	Đóng form cập nhật ngay lập tức, không lưu dữ liệu đã chỉnh sửa, quay về màn hình trước đó.
4	Bấm nút btnClose (X)	Đóng form cập nhật ngay lập tức, không lưu dữ liệu, tương tự btnCancel.
5	Bấm icon ẩn/hiện mật khẩu	Chuyển đổi trạng thái hiển thị của txtPassword:

4.3.2.14. Màn hình Cập nhập tài khoản người dùng

a. Giao diện:

Update Account

ID

0YmfNihISIQ1AJSjTezuul7s0jz1

* Email

phuc@gmail.com

* Full Name

Phuc Nguyen

* Phone Number

0292019000

* Role

Admin

Avatar

avatar.jpg

Cancel Update

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtId	TextBox	Hiển thị ID tài khoản (khóa chính, không được chỉnh sửa)
2	txtEmail	TextBox	Hiển thị email, không cho phép thay đổi
3	txtFullName	TextBox	Nhập/chỉnh sửa họ tên người dùng
4	txtPhone	TextBox	Nhập/chỉnh sửa số điện thoại người dùng
5	cmbRole	Dropdown	Hiển thị vai trò hiện tại, không cho phép chỉnh sửa
6	uploadAvatar	Upload Box	Cho phép cập nhật lại ảnh đại diện mới
7	btnUpdate	Button	Gửi yêu cầu cập nhật thông tin tài khoản
8	btnCancel	Button	Hủy cập nhật, đóng form

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Chỉnh sửa Full Name hoặc Phone	Cập nhật giá trị vào state của form
2	Bấm Update	Kiểm tra các trường hợp lệ, gửi API cập nhật nếu đúng
3	Bấm Cancel	Đóng form mà không lưu thay đổi
4	Upload avatar mới	Hiển thị ảnh preview và gắn vào payload cập nhật
5	Bỏ trống trường bắt buộc	Hiển thị cảnh báo yêu cầu nhập đủ thông tin cần thiết

4.3.2.15. Màn hình Thông tin tài khoản đang đăng nhập

a. Giao diện:

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	imgAvatar	Upload Box	Cập nhật ảnh đại diện cá nhân
2	txtEmail	Label	Hiển thị email của người dùng

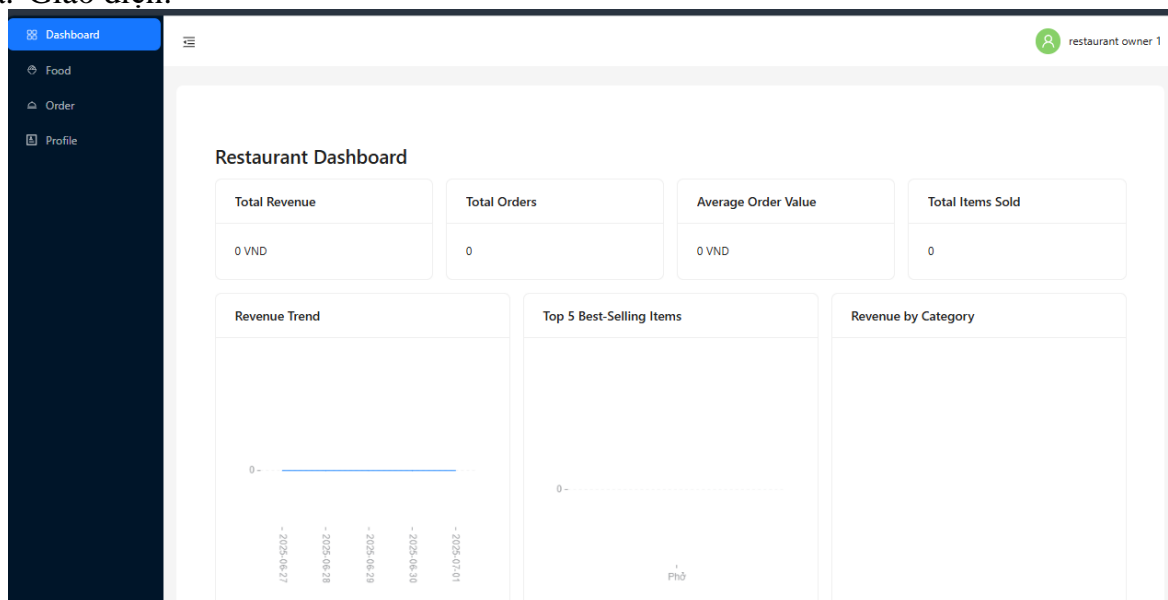
3	txtRole	Label	Hiển thị vai trò (admin / restaurant / user)
4	txtCreatedAt	Label	Hiển thị thời gian tạo tài khoản
5	txtFullName	TextBox	Cho phép người dùng chỉnh sửa họ tên
6	txtPhone	TextBox	Cho phép chỉnh sửa số điện thoại
7	btnSave	Button	Gửi dữ liệu cập nhật thông tin người dùng
8	btnCancel	Button	Hủy bỏ thay đổi, giữ nguyên thông tin trước đó

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Chỉnh sửa Full Name hoặc Phone	Cập nhật state/form values
2	Bấm nút Save	Kiểm tra dữ liệu, gọi API cập nhật nếu hợp lệ
3	Bấm nút Cancel	Hủy thao tác, không cập nhật dữ liệu mới
4	Upload ảnh đại diện mới	Preview hình ảnh mới, lưu vào dữ liệu cập nhật
5	Thiếu thông tin bắt buộc	Hiển thị thông báo lỗi yêu cầu nhập đầy đủ trước khi lưu

4.3.2.16. Màn hình Trang chủ nhà hàng và thống kê

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

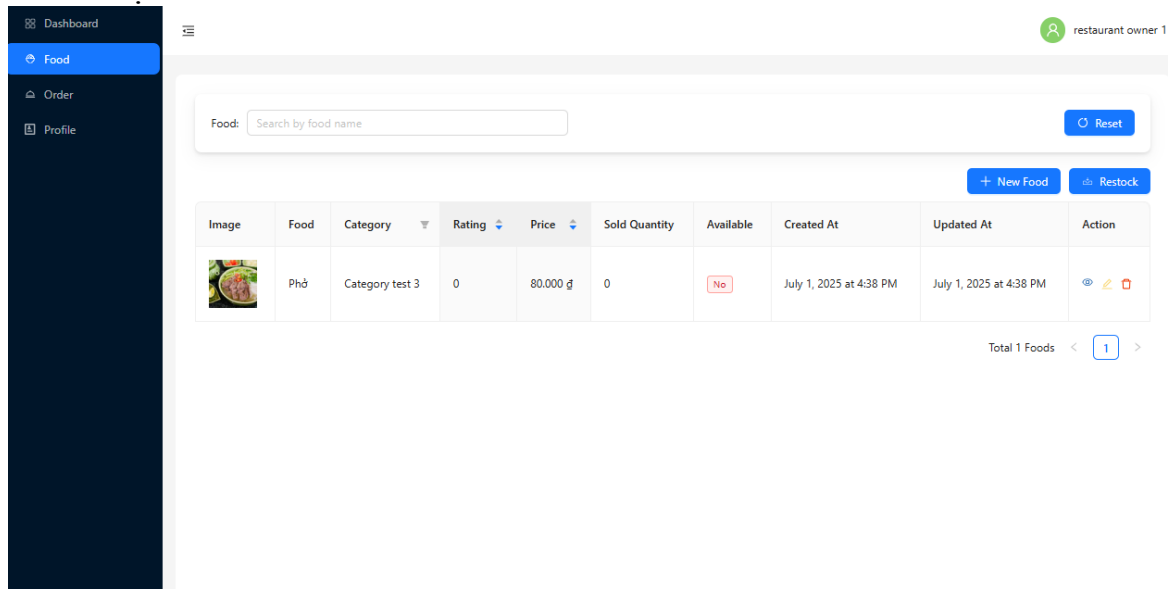
STT	Tên	Kiểu	Chức năng
1	lblTotalRevenue	Label	Hiển thị tổng doanh thu của nhà hàng
2	lblTotalOrders	Label	Hiển thị tổng số đơn hàng đã được đặt
3	lblAvgOrderValue	Label	Hiển thị giá trị trung bình mỗi đơn hàng
4	lblTotalItemsSold	Label	Hiển thị tổng số món ăn đã bán
5	chartRevenueTrend	Line Chart	Biểu đồ đường thể hiện xu hướng doanh thu theo ngày
6	chartBestSellingItems	Bar Chart	Biểu đồ cột hiển thị top 5 món ăn bán chạy nhất
7	chartRevenueByCategory	Pie Chart	Biểu đồ tròn thể hiện doanh thu theo từng danh mục món ăn

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Truy cập vào màn hình dashboard	Gửi request lấy dữ liệu thống kê từ server
2	Nhận dữ liệu từ API	Hiển thị các số liệu trong các thẻ thống kê (revenue, orders, etc.)
3	Render biểu đồ doanh thu	Hiển thị biểu đồ đường với trục thời gian (ngày) và giá trị doanh thu
4	Render biểu đồ món bán chạy	Hiển thị biểu đồ top 5 món có số lượng bán nhiều nhất
5	Render biểu đồ theo danh mục món ăn	Hiển thị tỷ trọng doanh thu theo từng danh mục

4.3.2.17. Màn hình Quản lý món ăn

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtSearchFood	TextBox	Tìm kiếm món ăn theo tên
2	btnReset	Button	Đặt lại bộ lọc tìm kiếm
3	btnNewFood	Button	Mở form tạo món ăn mới
4	btnRestock	Button	Mở form cập nhật lại số lượng món ăn
5	tblFood	Table	Hiển thị danh sách món ăn với các cột dữ liệu
6	colImage	Image	Hiển thị ảnh món ăn
7	colFoodName	Text	Tên món ăn
8	colCategory	Text	Danh mục của món ăn
9	colRating	Number	Điểm đánh giá
10	colPrice	Number	Giá món ăn
11	colSoldQuantity	Number	Số lượng đã bán
12	colAvailable	Badge/Text	Tình trạng còn bán hay không (Yes/No)
13	colCreatedAt	DateTime	Ngày tạo
14	colUpdatedAt	DateTime	Ngày cập nhật
15	colAction	Action Icons	Gồm các nút xem, sửa, xóa

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập từ khóa vào ô tìm kiếm	Lọc danh sách món ăn theo tên
2	Nhấn nút Reset	Xóa bộ lọc, hiển thị lại toàn bộ danh sách món ăn
3	Nhấn nút “New Food”	Mở form tạo mới món ăn
4	Nhấn nút “Restock”	Mở form cập nhật số lượng tồn món ăn
5	Nhấn icon xem	Xem chi tiết món ăn
6	Nhấn icon sửa	Mở form cập nhật món ăn
7	Nhấn icon xóa	Hiển thị xác nhận xóa, nếu đồng ý thì thực hiện xóa khỏi cơ sở dữ liệu
8	Nhấn các biểu tượng sắp xếp	Sắp xếp dữ liệu theo giá, đánh giá, v.v.

4.3.2.18. Màn hình Thêm mới món ăn

a. Giao diện:

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	imgImage	Upload Box	Tải lên ảnh cho món ăn

2	txtCategory	Dropdown	Chọn danh mục cho món ăn
3	txtDescription	TextBox	Nhập mô tả cho món ăn
4	txtFood	TextBox	Nhập tên món ăn
5	txtPrice	TextBox	Nhập giá của món ăn
6	txtQuantity	TextBox	Nhập số lượng của món ăn
7	btnCancel	Button	Hủy bỏ thao tác tạo món ăn
8	btnCreate	Button	Gửi dữ liệu để tạo mới món ăn

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Chỉnh sửa bất kỳ trường nào	Cập nhật state/form values
2	Bấm nút Create	Kiểm tra dữ liệu, gọi API tạo nếu hợp lệ
3	Bấm nút Cancel	Hủy thao tác, đóng form
4	Upload ảnh mới	Preview hình ảnh mới, lưu vào dữ liệu
5	Thiếu thông tin bắt buộc	Hiện thị thông báo lỗi yêu cầu nhập đầy đủ

4.3.2.19. Màn hình Thay đổi số lượng món ăn

a. Giao diện:

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtFood	TextBox	Nhập tên món ăn cần nhập lại
2	txtQuantity	TextBox	Nhập số lượng cần nhập lại
3	btnCancel	Button	Hủy bỏ thao tác nhập lại món ăn
4	btnRestock	Button	Gửi dữ liệu để nhập lại món ăn

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Chỉnh sửa Food hoặc Quantity	Cập nhật state/form values
2	Bấm nút Restock	Kiểm tra dữ liệu, gọi API nhập lại nếu hợp lệ
3	Bấm nút Cancel	Hủy thao tác, đóng form
4	Thiếu thông tin bắt buộc	Hiển thị thông báo lỗi yêu cầu nhập đầy đủ

4.3.2.20. Màn hình Cập nhật món ăn

a. Giao diện:

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	imgImage	Upload Box	Hiển thị và cập nhật ảnh món ăn
2	txtCategory	Dropdown	Chọn danh mục cho món ăn
3	txtDescription	TextBox	Nhập mô tả cho món ăn
4	txtFood	TextBox	Nhập tên món ăn
5	txtPrice	TextBox	Nhập giá của món ăn
6	txtQuantity	TextBox	Nhập số lượng của món ăn
7	btnCancel	Button	Hủy bỏ thao tác cập nhật món ăn

8	btnUpdate	Button	Gửi dữ liệu để cập nhật món ăn
---	-----------	--------	--------------------------------

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Chỉnh sửa bất kỳ trường nào	Cập nhật state/form values
2	Bấm nút Update	Kiểm tra dữ liệu, gọi API cập nhật nếu hợp lệ
3	Bấm nút Cancel	Hủy thao tác, đóng form
4	Upload ảnh mới	Preview hình ảnh mới, lưu vào dữ liệu
5	Thiếu thông tin bắt buộc	Hiện thị thông báo lỗi yêu cầu nhập đầy đủ

4.3.2.21. Màn hình Quản lý thông tin nhà hàng

a. Giao diện:

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	imgAvatar	Upload Box	Cập nhật ảnh đại diện cá nhân
2	txtEmail	Label	Hiện thị email của người dùng
3	txtRole	Label	Hiện thị vai trò (restaurant)
4	txtCreatedAt	Label	Hiện thị thời gian tạo tài khoản
5	txtFullName	TextBox	Cho phép chỉnh sửa họ tên
6	txtPhone	TextBox	Cho phép chỉnh sửa số điện thoại

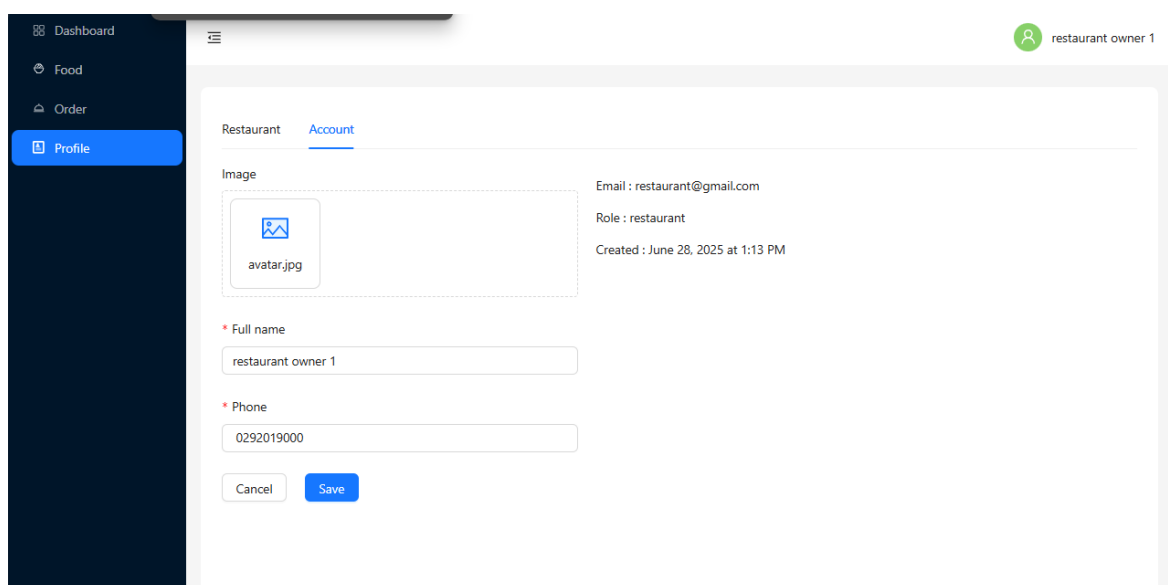
7	btnCancel	Button	Hủy bỏ thay đổi, giữ nguyên thông tin
8	btnSave	Button	Gửi dữ liệu cập nhật thông tin

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Chỉnh sửa Full Name hoặc Phone	Cập nhật state/form values
2	Bấm nút Save	Kiểm tra dữ liệu, gọi API cập nhật nếu hợp lệ
3	Bấm nút Cancel	Hủy thao tác, không cập nhật dữ liệu
4	Upload ảnh đại diện mới	Preview hình ảnh mới, lưu vào dữ liệu
5	Thiếu thông tin bắt buộc	Hiển thị thông báo lỗi yêu cầu nhập đầy đủ

4.3.2.22. Màn hình Quản lý thông tin tài khoản nhà hàng

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	imgImage	Upload Box	Hiển thị và cập nhật ảnh món ăn
2	txtCategory	Dropdown	Chọn danh mục cho món ăn
3	txtDescription	TextBox	Nhập mô tả cho món ăn
4	txtFood	TextBox	Nhập tên món ăn
5	txtPrice	TextBox	Nhập giá của món ăn

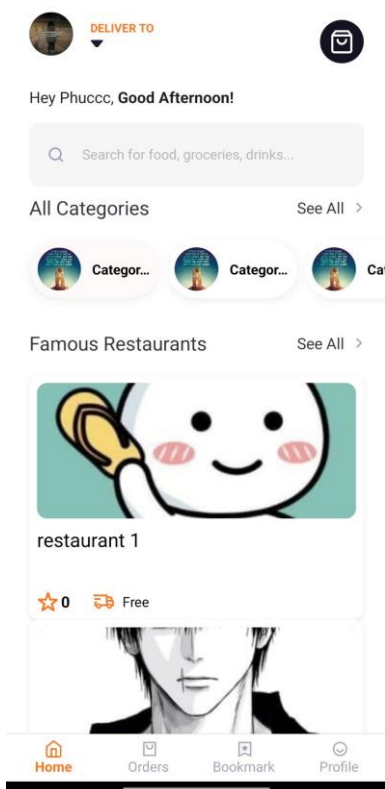
6	txtQuantity	TextBox	Nhập số lượng món ăn hiện có
7	btnCancel	Button	Hủy bỏ thao tác cập nhật món ăn
8	btnUpdate	Button	Gửi dữ liệu lên hệ thống để cập nhật món ăn

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Chỉnh sửa bất kỳ trường nào	Cập nhật giá trị vào form/state
2	Bấm nút Update	Kiểm tra hợp lệ dữ liệu, gọi API cập nhật nếu hợp lệ
3	Bấm nút Cancel	Hủy thao tác cập nhật, đóng form hoặc chuyển hướng về trang trước
4	Upload ảnh mới	Hiển thị ảnh xem trước, lưu ảnh vào dữ liệu cần gửi đi
5	Thiếu thông tin bắt buộc	Hiển thị thông báo lỗi, yêu cầu người dùng nhập đầy đủ thông tin

4.3.2.23. Màn hình trang chủ ứng dụng

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

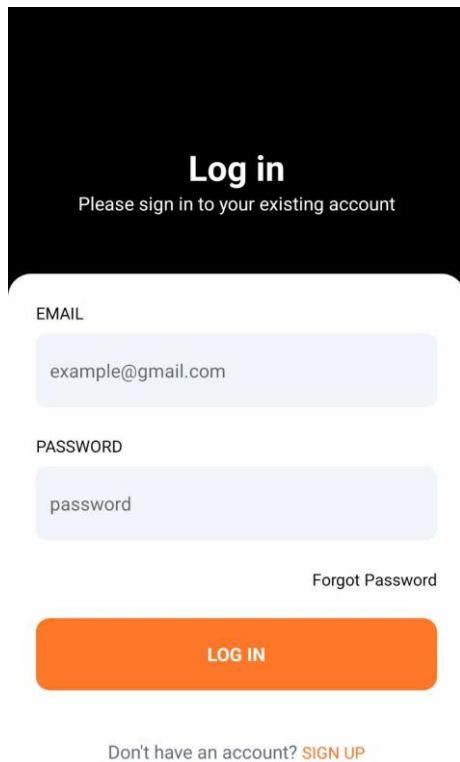
STT	Tên thành phần	Kiểu	Chức năng
1	listRestaurants	Grid/List	Hiển thị danh sách nhà hàng (ảnh, tên, rating, địa chỉ)
2	btnViewRestaurant	Button/Icon	Nhấn vào để xem chi tiết nhà hàng
3	refreshControl	Pull-to-Refresh	Làm mới danh sách nhà hàng
4	bottomNavBar	TabBar	Điều hướng đến các màn hình chính khác (Trang chủ, Tìm kiếm, Hồ sơ)

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Người dùng mở ứng dụng	Tải danh sách nhà hàng từ Firestore và hiển thị
2	Kéo xuống màn hình chính (pull-to-refresh)	Làm mới dữ liệu, hiển thị danh sách nhà hàng mới nhất
3	Bấm vào một nhà hàng trong danh sách	Chuyển sang màn hình chi tiết nhà hàng
4	Chuyển tab trên thanh điều hướng	Điều hướng sang trang tương ứng như Tìm kiếm, Hồ sơ, Giỏ hàng

4.3.2.24. Màn hình đăng ký/đăng nhập

a. Giao diện:



The image shows a login form with a dark header containing the text "Log in" and "Please sign in to your existing account". Below the header are two input fields: "EMAIL" with the placeholder "example@gmail.com" and "PASSWORD" with the placeholder "password". A link "Forgot Password" is positioned to the right of the password field. A large orange "LOG IN" button is centered below the inputs. At the bottom, there is a link "Don't have an account? SIGN UP".

b. Mô tả các đối tượng trên màn hình:

STT	Tên thành phần	Kiểu	Chức năng
1	txtEmail	TextBox	Nhập địa chỉ email
2	txtPassword	PasswordBox	Nhập mật khẩu
3	btnLogin	Button	Gửi yêu cầu đăng nhập đến hệ thống
4	btnRegister	Button	Điều hướng sang màn hình đăng ký
5	btnLoginWithGoogle	Button/Icon	Đăng nhập bằng tài khoản Google
6	lblErrorMessage	Label	Hiển thị thông báo lỗi nếu đăng nhập không thành công

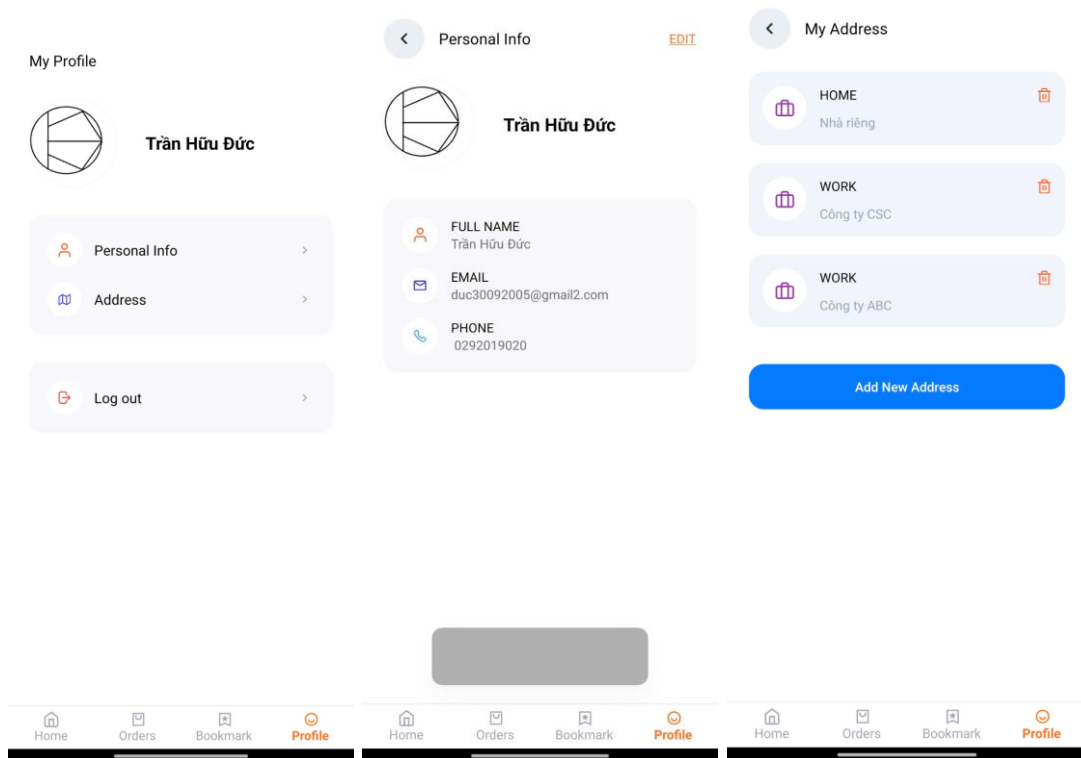
c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
-----	---------	-------

1	Nhấn nút "Đăng nhập"	Kiểm tra email/mật khẩu → gửi lên Firebase Auth để xác thực
2	Nhấn nút "Đăng ký"	Điều hướng sang màn hình tạo tài khoản mới
3	Nhấn nút "Đăng nhập bằng Google"	Thực hiện đăng nhập thông qua OAuth2 → nhận token và chuyển trạng thái
4	Đăng nhập thất bại	Hiển thị thông báo lỗi phía dưới form đăng nhập
5	Đăng nhập thành công	Điều hướng sang màn hình trang chủ người dùng

4.3.2.25. Màn hình hồ sơ

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên thành phần	Kiểu	Chức năng
1	imgAvatar	Image	Hiển thị ảnh đại diện người dùng
2	lblFullName	Label	Hiển thị họ tên người dùng
3	lblEmail	Label	Hiển thị email tài khoản
4	lblPhoneNumber	Label	Hiển thị số điện thoại
5	btnEditProfile	Button	Mở giao diện cập nhật thông tin cá nhân
6	btnChangePassword	Button	Mở giao diện đổi mật khẩu

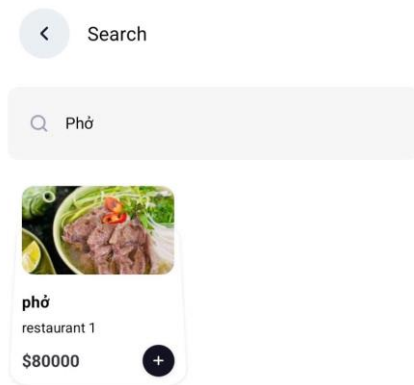
7	listAddresses	ListView	Hiển thị danh sách địa chỉ đã lưu
---	---------------	----------	-----------------------------------

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Vào tab "Hồ sơ" từ thanh điều hướng	Tải thông tin người dùng từ Firestore
2	Bấm “Chỉnh sửa thông tin”	Chuyển sang màn hình cập nhật hồ sơ
3	Bấm “Đổi mật khẩu”	Mở popup hoặc màn hình đổi mật khẩu
4	Tải lại trang	Đữ liệu người dùng được đồng bộ mới nhất từ server

4.3.2.26. Màn hình tìm kiếm món ăn

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

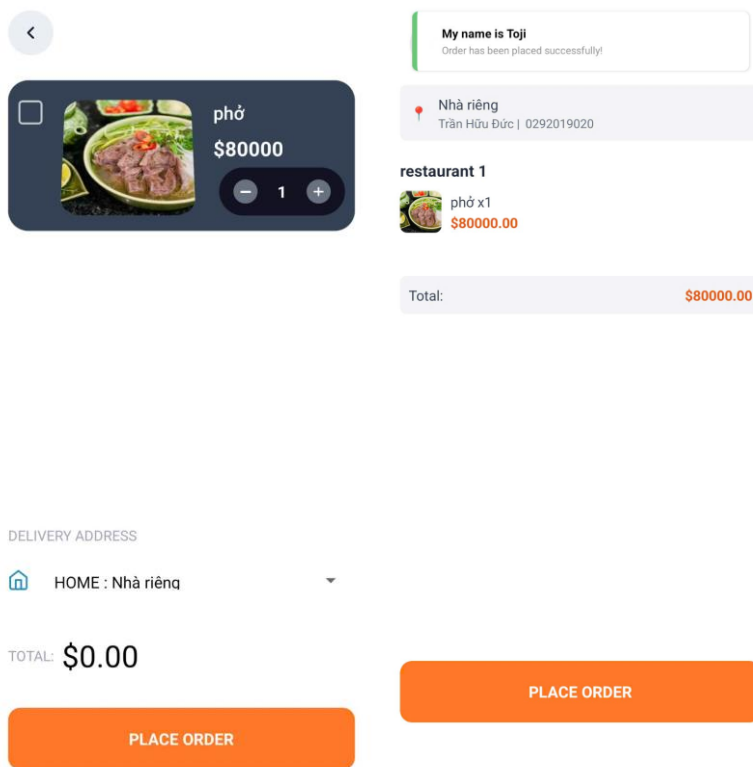
STT	Tên thành phần	Kiểu	Chức năng
1	txtSearchKeyword	TextBox	Nhập từ khóa để tìm kiếm món ăn
2	btnSearch	Button	Gửi yêu cầu tìm kiếm
3	listSearchResults	List/Grid	Hiển thị kết quả tìm kiếm món ăn
4	btnViewDish	Icon Button	Xem thông tin chi tiết món ăn

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập từ khóa vào ô tìm kiếm	Lưu từ khóa tìm kiếm vào state
2	Nhấn nút "Tìm kiếm"	Gửi truy vấn đến backend → nhận kết quả từ Firestore và hiển thị
3	Bấm vào món ăn trong kết quả	Chuyển sang màn hình chi tiết món ăn
4	Kết quả rỗng	Hiển thị thông báo "Không tìm thấy kết quả phù hợp"

4.3.2.27. Màn hình đặt hàng

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên thành phần	Kiểu	Chức năng
1	listCartItems	ListView	Hiển thị các món ăn đã thêm vào giỏ hàng
2	lblTotalAmount	Label	Tổng tiền của đơn hàng
3	dropdownAddress	Dropdown	Chọn địa chỉ giao hàng
4	dropdownPaymentMethod	Dropdown	Chọn phương thức thanh toán
5	btnConfirmOrder	Button	Gửi yêu cầu đặt hàng

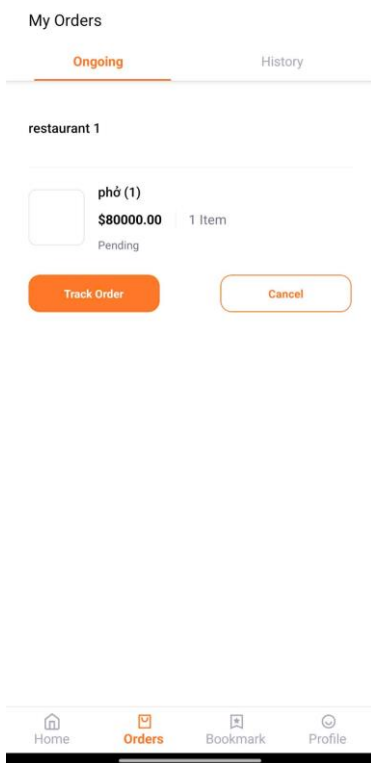
c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Hiển thị giỏ hàng	Lấy danh sách món ăn từ giỏ hàng trong Firestore
2	Thay đổi địa chỉ giao hàng	Cập nhật địa chỉ đã chọn vào đơn hàng
3	Chọn phương thức thanh toán	Cập nhật lựa chọn vào đơn hàng
4	Nhấn “Xác nhận đặt hàng”	Gửi đơn hàng lên Firestore (Order + Order_Detail), cập nhật số lượng

5	Đặt hàng thành công	Hiện thị thông báo và điều hướng sang màn hình xem đơn hàng
---	---------------------	---

4.3.2.28. Màn hình xem đơn hàng

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

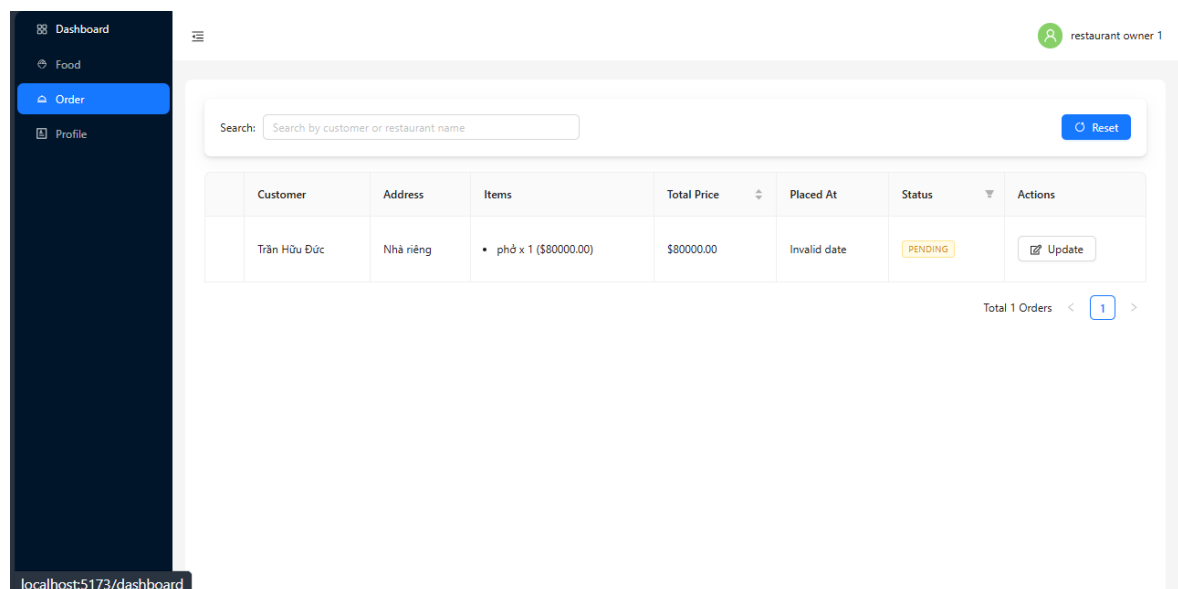
STT	Tên thành phần	Kiểu	Chức năng
1	listOrders	ListView	Hiện thị các đơn hàng của người dùng
2	lblOrderStatus	Label	Hiện thị trạng thái từng đơn hàng (Pending, Processing,...)
3	lblOrderDate	Label	Hiện thị thời gian đặt hàng
4	btnViewOrderDetails	Icon Button	Xem chi tiết đơn hàng
5	btnRateDish	Button	Đánh giá món ăn sau khi đơn đã hoàn thành (nếu có)

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Mở màn hình “Đơn hàng”	Tải danh sách đơn hàng của người dùng từ Firestore
2	Nhấn vào đơn hàng bất kỳ	Hiển thị popup hoặc trang chi tiết đơn hàng
3	Đơn hàng có trạng thái “Hoàn tất”	Hiển thị nút “Đánh giá”
4	Nhấn “Đánh giá”	Điều hướng đến màn hình gửi đánh giá món ăn
5	Trạng thái đơn thay đổi (realtime)	Cập nhật trạng thái mới từ Firestore và hiển thị

4.3.2.29. Màn hình Quản lý đơn đặt

a. Giao diện:



b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
1	txtSearch	TextBox	Tìm kiếm theo tên khách hàng hoặc nhà hàng
2	btnReset	Button	Đặt lại bộ lọc tìm kiếm
3	lblCustomer	Label	Hiển thị tên khách hàng
4	lblAddress	Label	Hiển thị địa chỉ giao hàng
5	lblItems	Label	Hiển thị danh sách món ăn
6	lblTotalPrice	Label	Hiển thị tổng giá đơn hàng
7	lblPlacedAt	Label	Hiển thị thời gian đặt hàng
8	lblStatus	Label	Hiển thị trạng thái đơn hàng

9	btnUpdate	Button	Cập nhật thông tin đơn hàng
10	lblTotalOrders	Label	Hiển thị tổng số đơn hàng
11	userAvatar	Icon	Hiển thị icon avatar người dùng
12	lblUserName	Label	Hiển thị tên người dùng đăng nhập

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Nhập dữ liệu vào txtSearch	Lọc danh sách đơn hàng theo từ khóa
2	Bấm nút Reset	Xóa bộ lọc tìm kiếm, hiển thị toàn bộ đơn hàng
3	Bấm nút Update	Mở form chỉnh sửa thông tin đơn hàng
4	Tải lại trang	Cập nhật danh sách đơn hàng mới nhất
5	Thay đổi trạng thái đơn hàng	Cập nhật trạng thái trên giao diện
6	Bấm vào tên người dùng	Hiển thị menu hoặc chuyển đến trang tài khoản

4.3.2.30. Màn hình Cập nhật đơn đặt

a. Giao diện:

b. Mô tả các đối tượng trên màn hình:

STT	Tên	Kiểu	Chức năng
-----	-----	------	-----------

1	lblUpdateOrder	Label	Tiêu đề "Update Order"
2	cbStatus	ComboBox (Dropdown)	Chọn trạng thái đơn hàng (Pending / Ongoing / Completed / Cancelled,...)
3	txtPhone	TextField	Nhập số điện thoại người nhận
4	txtShipper	TextField	Nhập tên hoặc ID shipper phụ trách đơn hàng
5	btnCancel	Button	Hủy bỏ thao tác cập nhật, đóng form
6	btnUpdate	Button	Xác nhận cập nhật đơn hàng với các thông tin đã nhập
7	btnCloseX	Button (Icon X)	Đóng cửa sổ cập nhật

c. Danh sách các biến cố và xử lý tương ứng trên màn hình:

STT	Biến cố	Xử lý
1	Chọn trạng thái trong cbStatus	Gán trạng thái mới cho đơn hàng
2	Nhập thông tin Phone	Cập nhật số điện thoại liên hệ của khách hàng
3	Nhập thông tin Shipper	Cập nhật shipper phụ trách giao hàng
4	Nhấn Update	Gửi request cập nhật đơn hàng; đóng form nếu thành công
5	Nhấn Cancel hoặc nút X	Đóng form mà không lưu thay đổi
6	Nếu thiếu trường bắt buộc (Status, Phone, Shipper)	Hiển thị cảnh báo và không cho phép cập nhật

4.4. Mô tả logic xử lý chính

4.4.1. Xác thực người dùng

- Người dùng đăng ký/đăng nhập → gửi email, password lên Backend
- Backend gọi Firebase Auth tạo/check user, trả về token
- Mọi API cần xác thực đều kiểm tra token qua middleware

4.4.2. Thêm/Xóa/Sửa dữ liệu

- Giao diện gửi request POST/PUT/DELETE tới API Backend
- Backend ghi dữ liệu vào Firestore (hoặc xóa document)
- Khi tạo mới: sinh id tự động, có thể upload ảnh lên Cloudinary rồi lưu link vào Firestore
- Khi xóa: backend kiểm tra quyền, xóa bản ghi, cập nhật các liên kết (ví dụ: xóa user thì xóa giỏ hàng, order liên quan...)

4.4.3. Hiển thị danh sách

- Frontend gọi API GET (ví dụ /restaurants, /foods, /orders)
- Backend truy vấn Firestore, trả về dữ liệu phân trang/lọc/sắp xếp
- Frontend đổ dữ liệu ra FlatList (mobile) hoặc Table/List (web)

4.5. Khó khăn gặp phải và cách khắc phục

4.5.1. Đồng bộ dữ liệu giữa Firebase Auth và Firestore

Vấn đề: Khi xóa user trên Auth, cần đồng bộ xóa cả dữ liệu trên Firestore và ngược lại.

Khắc phục: Viết logic xóa user đồng bộ cả hai phía, kiểm tra kỹ quyền truy cập.

4.5.2. Quản lý ảnh và upload file

Vấn đề: Lưu ảnh trên Firebase Storage phức tạp, tốc độ tải không ổn định.

Khắc phục: Dùng Cloudinary để upload ảnh, chỉ lưu URL vào Firestore.

4.5.3. Quản lý state phức tạp trên mobile

Vấn đề: Khi chuyển tab hoặc quay về màn hình trước, dữ liệu dễ bị mất/không đồng bộ.

Khắc phục: Dùng React Context, React Query để quản lý global state, cache dữ liệu.

4.5.4. Cấu hình môi trường đa nền tảng

Vấn đề: IP backend thay đổi khi chạy trên thật/emulator, dễ quên sửa config.

Khắc phục: Định nghĩa biến môi trường rõ ràng, hướng dẫn rõ cách sửa config trước khi build.

4.5.5. Truy vấn dữ liệu phức tạp trên Firestore (NoSQL)

Vấn đề: Không join được như SQL; truy vấn đa điều kiện, phân trang tốn nhiều round-trip.

Khắc phục: Thiết kế cấu trúc dữ liệu hợp lý, lưu thêm các trường phụ (denormalized data) để truy vấn nhanh.

Chương 5. TỔNG KẾT

5.1. Kết quả sản phẩm

Sau quá trình nghiên cứu và phát triển, nhóm đã hoàn thiện hệ thống Foodibidy với đầy đủ các tính năng cốt lõi, đáp ứng tốt nhu cầu đặt món trực tuyến cho cả khách hàng và nhà hàng. Ứng dụng không chỉ cung cấp giải pháp toàn diện cho quy trình đặt món, quản lý đơn hàng, thực đơn mà còn chú trọng vào trải nghiệm người dùng, bảo mật thông tin và khả năng vận hành ổn định trong môi trường thực tế.

Về Sản Phẩm:

Ứng dụng đã được phát triển hoàn chỉnh với giao diện trực quan, thân thiện cho cả người dùng cuối và đối tác nhà hàng. Kiến trúc dữ liệu được xây dựng hợp lý, đảm bảo lưu trữ, truy xuất và xử lý thông tin nhanh chóng, hiệu quả. Hệ thống phân quyền chi tiết theo vai trò (khách hàng, nhà hàng, quản trị viên), giúp tăng tính bảo mật và đảm bảo mỗi người dùng chỉ thao tác trong phạm vi cho phép. Độ ổn định, tin cậy của hệ thống luôn được ưu tiên, mang lại trải nghiệm sử dụng mượt mà, hạn chế tối đa lỗi phát sinh trong quá trình vận hành.

Về Chức Năng:

Toàn bộ chức năng cốt lõi như đăng ký, đăng nhập, tìm kiếm món ăn/nhà hàng, thêm vào giỏ hàng, đặt món, thanh toán, quản lý đơn hàng... đều đã được hoàn thiện và kiểm thử kỹ lưỡng. Người dùng có thể đặt nhiều món trong một lần thao tác, tiết kiệm thời gian và nâng cao hiệu suất trải nghiệm. Hệ thống hỗ trợ thanh toán trực tuyến an toàn, nhanh chóng, giúp khách hàng hoàn tất giao dịch dễ dàng. Ngoài ra, hệ thống còn hỗ trợ phân quyền chi tiết, quản lý các đối tượng như người dùng, nhà hàng, món ăn, đơn hàng và thống kê doanh thu. Logic xử lý được tổ chức chặt chẽ, dễ bảo trì và có khả năng mở rộng trong tương lai.

5.1.1. Những chức năng đã hoàn thành

- Đăng ký, đăng nhập, xác thực người dùng (khách hàng, nhà hàng)
- Quản lý, chỉnh sửa thông tin cá nhân và nhà hàng
- Quản lý món ăn, danh mục món ăn
- Tìm kiếm, lọc, xem chi tiết món ăn, nhà hàng
- Thêm, sửa, xóa món ăn (dành cho nhà hàng)

- Giỏ hàng: thêm/xóa món, cập nhật số lượng
- Đặt hàng, theo dõi và quản lý đơn hàng
- Đánh giá, nhận xét món ăn/nhà hàng

5.1.2. Mức độ hoàn thiện của sản phẩm

- Giao diện người dùng: Đã hoàn thiện cho cả mobile (React Native) và web admin (ReactJS), thân thiện, dễ sử dụng.
- Chức năng: Đáp ứng đầy đủ các nghiệp vụ cốt lõi trong đặt món và quản lý nhà hàng.
- Tích hợp thực tế: Kết nối thành công tới Firebase (Auth, Firestore), Cloudinary (upload ảnh).
- Kiểm thử: Đã kiểm thử trên nhiều thiết bị, các chức năng cơ bản hoạt động ổn định.

5.2. Đánh giá hiệu quả

- Giao diện: Hiện đại, bố cục rõ ràng, dễ thao tác cho cả người dùng và nhà hàng.
- Tốc độ: Phản hồi nhanh, dữ liệu realtime qua Firestore, tối ưu trải nghiệm người dùng.
- Độ ổn định: Hệ thống hoạt động tốt, ít gặp lỗi nghiêm trọng trong quá trình thử nghiệm.

5.3. Hạn chế

Mặc dù ứng dụng đã đáp ứng được hầu hết các yêu cầu đề ra ban đầu, vẫn còn một số hạn chế nhất định cần được khắc phục trong tương lai:

- Một số tính năng nâng cao (ví dụ: thanh toán online, thông báo realtime, tracking đơn hàng) chưa kịp triển khai.
- Giao diện mobile/web còn có thể cải thiện thêm về độ mượt, hiệu ứng chuyển cảnh.
- Một số lỗi nhỏ về validate dữ liệu đầu vào, UX chưa hoàn thiện 100%.
- Chưa tối ưu truy vấn dữ liệu lớn (nhiều nhà hàng, nhiều món ăn).
- Chưa có phân hệ quản trị cho admin tổng thể.

5.4. Định hướng phát triển

Để hoàn thiện hơn và nâng cao trải nghiệm người dùng, ứng dụng có thể tiếp tục được mở rộng và nâng cấp theo các hướng sau:

- Phát triển phiên bản mobile (Android/iOS) để mở rộng khả năng tiếp cận người dùng, hỗ trợ thao tác mọi lúc mọi nơi, đặc biệt với nhóm khách hàng bận rộn hoặc thường xuyên di chuyển.

- Tích hợp hệ thống thông báo đẩy (push notification) và gửi email tự động nhắc lịch bay, giúp người dùng chủ động theo dõi hành trình và giảm thiểu rủi ro bỏ lỡ chuyến.
- Cải tiến UI/UX, đặc biệt là các yếu tố màu sắc, kích thước chữ, bố cục để thân thiện hơn với người lớn tuổi hoặc những người không quen sử dụng công nghệ.
- Kết nối trực tiếp với API của các hãng hàng không, cho phép cập nhật dữ liệu chuyến bay theo thời gian thực, giảm thiểu việc nhập liệu thủ công và nâng cao độ chính xác thông tin.

Chương 6. ĐÁNH GIÁ KẾT QUẢ THỰC HIỆN

6.1. Cấp độ tập thể

Trong suốt quá trình thực hiện đề tài, nhóm đã thể hiện tinh thần làm việc nghiêm túc, chủ động và có sự phối hợp hiệu quả giữa các thành viên. Mỗi cá nhân đều được phân công rõ ràng theo thế mạnh và sở trường, từ đó phát huy tốt vai trò của mình trong tổng thể dự án.

Ở cấp độ tập thể, nhóm đã xây dựng được quy trình làm việc khoa học, chia nhỏ công việc theo giai đoạn và thường xuyên tổ chức họp nhóm để cập nhật tiến độ, xử lý các vấn đề phát sinh kịp thời. Việc sử dụng các công cụ hỗ trợ như Google Meet, Discord, Github... cũng góp phần nâng cao hiệu suất làm việc và đảm bảo sự đồng bộ giữa các thành viên. Nhóm không chỉ tập trung vào hoàn thiện sản phẩm mà còn chú trọng đến chất lượng mã nguồn, tính bảo trì của hệ thống và khả năng mở rộng trong tương lai. Tinh thần hỗ trợ lẫn nhau, cùng nhau học hỏi và chia sẻ kiến thức là một trong những điểm mạnh nổi bật của tập thể nhóm trong đề tài lần này.

6.2. Cấp độ cá nhân

Ở cấp độ cá nhân, mỗi thành viên đều chủ động nghiên cứu công nghệ phù hợp với phần việc được giao. Các bạn không ngại tiếp cận những nội dung mới như xử lý thanh toán online, phân quyền người dùng, tổ chức cấu trúc cơ sở dữ liệu hay triển khai các kỹ thuật bảo mật cơ bản trong ứng dụng web và mobile. Ngoài kiến thức kỹ thuật, mỗi thành viên còn rèn luyện được các kỹ năng mềm quan trọng như làm việc nhóm, giao tiếp, tự học, quản lý thời gian và giải quyết vấn đề thực tiễn.

Thông qua đề tài này, từng thành viên không chỉ hoàn thành một sản phẩm cụ thể mà còn tích lũy được nhiều kinh nghiệm thực tế, từng bước làm quen với quy trình phát triển phần mềm bài bản, từ phân tích, thiết kế, lập trình đến kiểm thử và triển khai. Đây là một tiền đề quý giá giúp nhóm tự tin hơn khi tiếp cận các dự án lớn hơn trong tương lai, đặc biệt là trong môi trường làm việc chuyên nghiệp.

6.3. Thuận lợi và khó khăn trong quá trình thực hiện

6.3.1. Thuận lợi

- Đề tài gần gũi và có tính ứng dụng thực tế cao: Việc chọn đề tài về ứng dụng đặt đồ ăn là một điểm thuận lợi lớn vì quy trình nghiệp vụ rõ ràng, dễ phân tích và có thể mô phỏng tương đối đầy đủ trong môi trường học thuật. Đồng thời, đây là một bài toán phổ biến trong thực tiễn, nên nhóm có thể dễ dàng tìm kiếm tài liệu tham khảo và các mô hình hệ thống mẫu để học hỏi.
- Thành viên có tinh thần tự học và phối hợp tốt: Nhóm gồm các thành viên có chung mục tiêu và tinh thần trách nhiệm cao. Mỗi người đều cố gắng hoàn thành tốt phần việc của mình, đồng thời sẵn sàng hỗ trợ lẫn nhau khi gặp khó khăn. Chính điều này giúp tiến độ thực hiện không bị gián đoạn và tránh được tình trạng dồn việc vào giai đoạn cuối.
- Nguồn tài liệu và công nghệ phong phú: Nhờ vào sự phát triển mạnh mẽ của cộng đồng lập trình và tài nguyên học tập trên internet (StackOverflow, GitHub, YouTube, tài liệu từ các khóa học trực tuyến,...), nhóm dễ dàng tra cứu và áp dụng các công nghệ phù hợp.

6.3.2. Khó khăn

- Chưa có kinh nghiệm thực tế: Mặc dù đã có kiến thức nền tảng về phát triển phần mềm, nhưng khi tiếp cận các chức năng chuyên sâu như tích hợp thanh toán online, xử lý đặt nhiều vé cùng lúc, hay thiết kế phân quyền chi tiết theo vai trò, nhóm vẫn gặp không ít trở ngại. Việc lựa chọn công nghệ phù hợp, hiểu rõ cách hoạt động của từng thành phần trong hệ thống cũng là một thử thách lớn ở giai đoạn đầu.
- Thiếu thiết bị kiểm thử trên nhiều nền tảng: Do ứng dụng hiện mới chỉ hoạt động trên nền web, nhóm chưa có điều kiện thử nghiệm trên các thiết bị di động hoặc trình duyệt cũ. Điều này khiến việc tối ưu giao diện cho các nhóm người dùng khác nhau (ví dụ: người lớn tuổi, người ít tiếp xúc công nghệ) gặp nhiều hạn chế.
- Không có người dùng thật để kiểm thử: Trong môi trường học tập, nhóm chỉ có thể mô phỏng các tình huống sử dụng. Không có người dùng thực tế nên nhóm khó đánh giá được hiệu quả thực tế của quy trình, trải nghiệm người

dùng và hiệu suất hệ thống khi tải cao. Điều này sẽ cần được khắc phục nếu sản phẩm được triển khai trong môi trường doanh nghiệp thật.

- Khó khăn về thời gian: Các thành viên đều có lịch học cá nhân dày đặc, nên việc sắp xếp thời gian làm việc nhóm đôi khi bị ảnh hưởng. Nhóm phải chia nhỏ công việc và tận dụng thời gian rảnh để phối hợp thực hiện, điều này đòi hỏi sự linh hoạt và nỗ lực lớn từ mỗi thành viên.

TÀI LIỆU THAM KHẢO

[1] React Native, “React Native Documentation,” React Native. [Online].

Available:

<https://reactnative.dev/docs/getting-started>

. [Accessed: Jun. 15, 2025]. [2] Firebase, “Get started with Firebase for web,”

Firebase Documentation. [Online]. Available:

<https://firebase.google.com/docs/web/setup>

. [Accessed: Jun. 15, 2025]. [3] Node.js Foundation, “About Node.js,” Node.js.

[Online]. Available:

<https://nodejs.org/en/about/>

. [Accessed: Jun. 15, 2025]. [4] Cloudinary, “Cloudinary Documentation,”

Cloudinary. [Online]. Available:

<https://cloudinary.com/documentation>

. [Accessed: Jun. 15, 2025]. [5] Mozilla, “Express.js Guide,” MDN Web Docs.

[Online]. Available:

https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs

. [Accessed: Jun. 15, 2025]. [6] Ant Design, “Ant Design of React,” Ant Design

Docs. [Online]. Available:

<https://ant.design/docs/react/introduce>

. [Accessed: Jun. 15, 2025]. [7] Google Material Design, “Material Design

Guidelines,” Material.io. [Online]. Available:

<https://material.io/design>

. [Accessed: Jun. 15, 2025]. [8] RESTful API, “REST API Tutorial,”

RestApiTutorial.com. [Online]. Available:

<https://restfulapi.net/>

. [Accessed: Jun. 15, 2025]. [9] Trần Văn Nam, “Ứng dụng React Native xây dựng hệ thống đặt đồ ăn trực tuyến,” Luận văn tốt nghiệp, Trường Đại học Bách Khoa Hà Nội, 2024. [10] Nguyễn Thị Hà, “Thiết kế và triển khai hệ thống quản lý nhà hàng sử dụng Firebase,” Đồ án tốt nghiệp, Trường Đại học Công nghệ Thông tin TP.HCM, 2023. [11] Google, “Firebase Authentication Documentation,” Firebase. [Online]. Available:

<https://firebase.google.com/docs/auth>

. [Accessed: Jun. 15, 2025]. [12] GitHub, “Awesome Food Delivery,” GitHub Repository. [Online]. Available:

<https://github.com/topics/food-delivery-app>

. [Accessed: Jun. 15, 2025].