

**BỘ GIÁO DỤC VÀ ĐÀO TẠO      BỘ NÔNG NGHIỆP VÀ PTNT**  
**TRƯỜNG ĐẠI HỌC THỦY LỢI**



**BÙI TIẾN DŨNG**

**XÂY DỰNG ỨNG DỤNG GIAO HÀNG VÀ ĐẶT ĐỒ ĂN  
TRỰC TUYẾN UBER EATS ĐA NỀN TẢNG FLUTTER  
VỚI NGÔN NGỮ DART**

**ĐỒ ÁN TỐT NGHIỆP**

**TP HỒ CHÍ MINH, THÁNG 7 NĂM 2024**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO      BỘ NÔNG NGHIỆP VÀ PTNT**  
**TRƯỜNG ĐẠI HỌC THỦY LỢI**

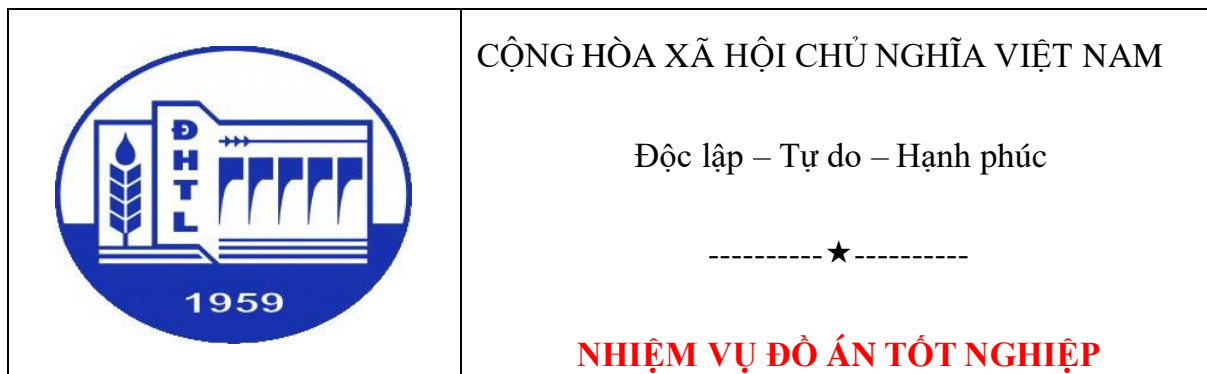
**BÙI TIẾN DŨNG**

**XÂY DỰNG ỨNG DỤNG GIAO HÀNG VÀ ĐẶT ĐỒ ĂN  
TRỰC TUYẾN UBER EATS ĐA NỀN TẢNG FLUTTER  
VỚI NGÔN NGỮ DART**

Ngành           : Công nghệ thông tin  
Mã số           : TLS106

**NGƯỜI HƯỚNG DẪN: Ths. Trần Ngọc Dân**

**TP HỒ CHÍ MINH, THÁNG 7 NĂM 2024**



**Họ tên sinh viên:** BÙI TIẾN DŨNG

**Hệ đào tạo:** Đại học chính quy

**Lớp:** S21-60TH1

**Ngành:** Công nghệ thông tin

**Khoa:** Công nghệ thông tin

### 1. Tên đề tài

**XÂY DỰNG ỨNG DỤNG GIAO HÀNG VÀ ĐẶT ĐỒ ĂN TRỰC TUYẾN UBER  
EATS ĐA NỀN TẢNG FLUTTER VỚI NGÔN NGỮ DART**

### 2. Các tài liệu cơ bản

Flutter, [Flutter - Build apps for any screen](#)

Firebase, [Firebase | Google's Mobile and Web App Development Platform](#)

Dart, [Dart programming language | Dart](#)

### 3. Nội dung phần thuyết minh

**Tỷ lệ %**

Chương 1: Mô tả bài toán	5%
Chương 2: Cơ sở lý thuyết	5%
Chương 3: Tổng quan về phần mềm	10%
Chương 4: Phân tích hệ thống	50%

Chương 5: Triển khai hệ thống 25%

Chương 6: Kết luận và hướng phát triển 5%

#### 4. Hình ảnh và bảng đồ án

Đồ án gồm 49 hình ảnh và 8 bảng.

#### 5. Giảng viên hướng dẫn

Giảng viên hướng dẫn toàn bộ đồ án: ThS.Trần Ngọc Dân

#### 6. Ngày giao nhiệm vụ đồ án tốt nghiệp

Ngày 8 tháng 4 năm 2024

**Trưởng Bộ môn**  
(Ký và ghi rõ họ tên)

**Giáo viên hướng dẫn chính**  
(Ký và ghi rõ họ tên)

**ThS. Trần Ngọc Dân**

Nhiệm vụ đồ án đã được Hội đồng thi tốt nghiệp của Khoa thông qua

**Chủ tịch Hội đồng Phân hiệu**  
(Ký và ghi rõ họ tên)

Sinh viên đã hoàn thành đồ án và nộp đồ án cho hội đồng thi ngày 14 tháng 7 năm 2024

**Sinh viên làm đồ án tốt nghiệp**  
(Ký và ghi rõ họ tên)

**Bùi Tiến Dũng**

# GÁY BÌA ĐỒ ÁN TỐT NGHIỆP

BÙI TIẾN DŨNG

ĐỒ ÁN TỐT NGHIỆP

HỒ CHÍ MINH, NĂM 2024

## LỜI CAM ĐOAN

Tôi xin cam đoan rằng:

Đồ án tốt nghiệp với đề tài “Xây dựng ứng dụng giao hàng đặt đồ ăn trực tuyến Uber Eats đa nền tảng Flutter với ngôn ngữ Dart” này là kết quả nghiên cứu và làm việc của chính bản thân tôi dưới sự hướng dẫn của thầy Trần Ngọc Dân.

Các số liệu, kết quả nghiên cứu trong đồ án là trung thực và chưa từng được công bố trong bất kỳ đồ án nào khác.

Mọi tham khảo từ các nguồn tài liệu khác đều được trích dẫn và ghi rõ nguồn gốc.

Nếu có bất kỳ sự sai phạm nào so với cam đoan trên, tôi xin hoàn toàn chịu trách nhiệm trước Nhà trường và Pháp luật.

Tác giả ĐATN

*(Ký và ghi rõ họ tên)*

**Bùi Tiến Dũng**

## LỜI CẢM ƠN

Trước tiên, em xin gửi lời cảm ơn chân thành đến Thầy Trần Ngọc Dân là người đã tận tình hướng dẫn, hỗ trợ và góp ý quý báu trong suốt quá trình thực hiện và hoàn thành đồ án tốt nghiệp này.

Em cũng xin cảm ơn quý thầy cô trong bộ môn Công nghệ thông tin, Phân hiệu Trường Đại học Thủy lợi đã truyền đạt những kiến thức quý báu và tạo điều kiện thuận lợi cho tôi trong suốt quá trình học tập và nghiên cứu tại trường.

Đặc biệt, em xin gửi lời cảm ơn sâu sắc đến gia đình và bạn bè, những người đã luôn động viên, hỗ trợ tinh thần và vật chất trong suốt thời gian học tập và thực hiện đồ án.

Cuối cùng, em xin cảm ơn tất cả những ai đã đóng góp ý kiến, giúp đỡ và tạo điều kiện cho tôi hoàn thành đồ án lần này.

## This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the entire width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the paper.

Chữ ký của giảng viên



## NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

TP. Hồ Chí Minh, ngày ... tháng ... năm 2024

Chữ ký của giảng viên

# MỤC LỤC

LỜI CAM ĐOAN .....	i
LỜI CẢM ƠN.....	ii
NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN .....	iii
NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN .....	iv
MỤC LỤC .....	v
DANH MỤC CÁC HÌNH ẢNH.....	viii
DANH MỤC BẢNG BIỂU .....	x
DANH MỤC CÁC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ .....	xi
CHƯƠNG 1 TỔNG QUAN VỀ ĐỀ TÀI.....	1
1.1. Giới thiệu đề tài .....	1
1.2. Mục tiêu đề tài .....	1
1.3. Ý nghĩa đề tài.....	2
1.4. Phương pháp nghiên cứu.....	3
1.5. Phạm vi nghiên cứu .....	4
1.6. Những đóng góp của đề tài .....	6
1.7. Phạm vi sử dụng .....	8
1.8. Flutter .....	10
1.8.1. Giới thiệu về Flutter .....	10
1.8.2. Đặc điểm chính của Flutter.....	10
1.8.3. Ưu điểm của Flutter.....	10
1.8.4. Kiến trúc của Flutter.....	11
1.8.5. Ứng dụng thực tế của Flutter .....	11
1.8.6. So sánh với các công cụ khác .....	11
1.8.7. Ưu điểm và nhược điểm của Flutter.....	11
1.9. Dart .....	13
1.9.1. Giới thiệu về Dart.....	13
1.9.2. Đặc điểm của Dart.....	13
1.9.3. Dart VM (Virtual Machine).....	14
1.9.4. Flutter và Dart .....	14
1.9.5. Cộng đồng và tài nguyên .....	14
1.9.6. Ứng dụng thực tế của Dart .....	14
1.9.7. Ưu nhược điểm của Dart .....	14
1.10. Firebase .....	16
1.10.1. Giới thiệu về Firebase .....	16
1.10.2. Các tính năng chính của Firebase .....	16
1.10.3. Lợi ích của Firebase .....	17
1.10.4. Nhược điểm của Firebase .....	18
CHƯƠNG 2 KHẢO SÁT HỆ THỐNG .....	19

2.1. Yêu cầu chung về phần mềm.....	19
2.1.1. Trang dành cho khách hàng:.....	19
2.1.2. Trang dành cho Nhà hàng:.....	19
2.1.3. Ứng dụng giành cho bên phía vận chuyển .....	19
2.2. Yêu cầu chức năng.....	19
2.2.1. Đối với ứng dụng đặt đồ ăn trực tuyến giành cho người dùng: .....	19
2.2.2. Đối với ứng dụng giao hàng đồ ăn trực tuyến cho người quản lý nhà hàng. .....	20
2.2.3. Đối với ứng dụng giao hàng đồ ăn trực tuyến cho bên phía vận chuyển....	20
2.3. Yêu cầu phi chức năng.....	21
2.3.1. Hiệu suất (Performance).....	21
2.3.2. Khả năng mở rộng (Scalability).....	21
2.3.3. Bảo mật (Security) .....	21
2.3.4. Khả năng sử dụng (Usability).....	21
2.3.5. Khả năng bảo trì (Maintainability).....	21
2.3.6. Độ tin cậy (Reliability).....	22
2.3.7. Tính tương thích (Compatibility).....	22
2.3.8. Khả năng mở rộng quốc tế (Internationalization and Localization) .....	22
2.4. Ràng buộc thiết kế .....	22
2.4.1. Công nghệ sử dụng.....	22
2.5. Đối tượng người dùng.....	22
2.6. Các ràng buộc .....	23
2.7. Giả định và phụ thuộc .....	23
<b>CHƯƠNG 3 PHÂN TÍCH HỆ THỐNG .....</b>	<b>24</b>
3.1. Thiết kế cơ sở dữ liệu.....	24
3.1.1. Mô hình ERD .....	24
3.1.2. Mô tả vắn tắt các thực thể.....	25
3.2. Mô hình Class Diagram .....	25
3.3. Biểu đồ Use Case.....	26
3.3.1. Biểu đồ Use Case tổng quát.....	26
3.3.2. Biểu đồ Use Case phân rã và đặc tả chi tiết .....	26
3.4. Biểu đồ hoạt động (Activity Diagram) .....	36
3.4.1. Activity Diagram: Đăng nhập.....	36
3.4.2. Activity Diagram: Đăng ký .....	37
3.4.2. Activity Diagram: Đăng xuất.....	37
3.4.5. Activity Diagram: Quản lý món ăn -Thêm .....	38
3.4.6. Activity Diagram: Quản lý món ăn - Sửa.....	38
3.4.7. Activity Diagram: Quản lý món ăn -Xóa .....	39
3.4.8. Activity Diagram: Quản lý món ăn -Xem thông tin .....	39
3.4.9. Activity Diagram: Quản lý món ăn – Tìm kiếm món ăn .....	40

3.4.10. Activity Diagram: Quản lý tài khoản – Xem thông tin tài khoản.....	40
3.4.11. Activity Diagram: Quản lý tài khoản – Sửa thông tin .....	41
3.5. Biểu đồ tuần tự (Sequence Diagram).....	42
3.5.1. Sequence Diagram: Đăng nhập.....	42
3.5.2. Sequence Diagram: Đăng ký .....	42
3.5.3. Sequence Diagram: Đăng xuất .....	42
3.5.4. Sequence Diagram: Quản lý món ăn -Thêm món ăn .....	43
3.5.5. Sequence Diagram: Quản lý món ăn -Sửa món ăn .....	43
3.5.6. Sequence Diagram: Quản lý món ăn -Xóa món ăn .....	44
3.5.7. Sequence Diagram: Quản lý món ăn -Tìm kiếm món ăn.....	45
3.5.8. Sequence Diagram: Quản lý món ăn -Xem thông tin món ăn.....	45
3.5.9. Sequence Diagram: Quản lý tài khoản- Xem thông tin .....	46
3.5.10. Sequence Diagram: Quản lý tài khoản- Sửa thông tin.....	46
CHƯƠNG 4 THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG .....	47
4.1. Xây dựng đăng nhập cho người dùng (khách hàng, nhà hàng, nhà vận chuyển.....	47
4.2. Xây dựng trang chủ cho khách hàng .....	49
4.3. Xây dựng trang Danh mục cho khách hàng.....	50
4.4. Xây dựng trang Danh mục cho khách hàng.....	51
4.5. Xây dựng trang Danh mục cho khách hàng.....	52
4.6. Xây dựng trang giỏ hàng cho khách hàng .....	53
4.7. Xây dựng trang tài khoản cho khách hàng.....	54
4.8. Xây dựng trang thêm địa chỉ mặc định cho khách hàng.....	55
4.9. Xây dựng trang phần thưởng cho khách hàng .....	56
4.10. Xây dựng trang quà tặng cho khách hàng.....	57
4.11 Xây dựng trang khuyến mãi cho khách hàng.....	58
4.12. Xây dựng trang thông tin thanh toán cho khách hàng.....	59
4.13. Xây dựng giao diện quản lý đơn hàng cho người dùng(khách hàng, nhà hàng và người vận chuyển đơn hàng) .....	60
4.14. Xây dựng giao diện đăng ký cho nhà hàng. ....	62
4.15. Xây dựng giao diện thêm món ăn cho nhà hàng.....	62
4.16. Xây dựng giao diện trang chủ cho nhà hàng.....	63
4.17. Xây dựng trang đăng ký cho Nhà vận chuyển (tài xế) .....	64
4.18. Xây dựng trang chủ cho tài xế.....	65
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....	67
1. Nội dung đã đạt được.....	67
1.1. Về lí thuyết.....	67
1.2. Về sản phẩm.....	67
2. Hạn chế của đề tài.....	68
3. Hướng phát triển đề tài .....	68
TÀI LIỆU THAM KHẢO.....	70

## DANH MỤC CÁC HÌNH ẢNH

Hình 3.1 Mô hình ERD.....	24
Hình 3.2 Mô hình Class Diagram.....	25
Hình 3.3 Biểu đồ Use Case tổng quát.....	26
Hình 3.4 Biểu đồ Use Case phân rã Quản lý món ăn- Khách hàng .....	29
Hình 3.5 Biểu đồ Use Case phân rã Quản lý món ăn- Nhà hàng.....	32
Hình 3.6 Biểu đồ Use Case phân rã Quản lý đơn hàng- Nhà hàng.....	34
Hình 3.7 Biểu đồ Use Case phân rã Quản lý đơn hàng- Người vận chuyển .....	35
Hình 3.8 Activity Diagram Đăng nhập.....	37
Hình 3.9 Activity Diagram Đăng ký .....	37
Hình 3.10 Activity Diagram Đăng xuất.....	38
Hình 3.11 Activity Diagram Quản lý món ăn- Thêm.....	38
Hình 3.12 Activity Diagram Quản lý món ăn- Sửa.....	39
Hình 3.13 Activity Diagram Quản lý món ăn- Xóa .....	39
Hình 3.14 Activity Diagram Quản lý món ăn- Xem thông tin .....	40
Hình 3.15 Activity Diagram Quản lý món ăn- Tìm kiếm món ăn .....	40
Hình 3.16 Activity Diagram Quản lý tài khoản- Xem thông tin .....	41
Hình 3.17 Activity Diagram Quản lý món ăn- Sửa thông tin.....	41
Hình 3.18 Biểu đồ tuần tự Đăng nhập .....	42
Hình 3.19 Biểu đồ tuần tự Đăng ký .....	42
Hình 3.20 Biểu đồ tuần tự Đăng xuất .....	43
Hình 3.21 Biểu đồ tuần tự Quản lý món ăn- Thêm món ăn .....	43
Hình 3.22 Biểu đồ tuần tự Quản lý món ăn- Sửa món ăn .....	44
Hình 3.23 Biểu đồ tuần tự Quản lý món ăn- Xóa món ăn .....	44
Hình 3.24 Biểu đồ tuần tự Quản lý món ăn- Tìm kiếm món ăn .....	45
Hình 3.25 Biểu đồ tuần tự Quản lý món ăn- Xem món ăn.....	45
Hình 3.26 Biểu đồ tuần tự Quản lý món ăn- Xem thông tin tài khoản .....	46
Hình 3.27 Biểu đồ tuần tự Quản lý tài khoản- Sửa thông tin .....	46
Hình 4.1 Giao diện đăng nhập- Khách hàng.....	47
Hình 4.2 Giao diện nhập mã OTP- Khách hàng .....	48
Hình 4.3 Giao diện Đăng ký (Khách hàng) .....	49
Hình 4.4 Giao diện trang chủ cho khách hàng.....	50
Hình 4.5 Giao diện danh mục cho khách hàng .....	51
Hình 4.6 Giao diện Tìm kiếm cho khách hàng .....	52
Hình 4.7 Giao diện Thông tin món ăn .....	53
Hình 4.8 Giao diện giỏ hàng .....	54
Hình 4.9 Giao diện tài khoản .....	55
Hình 4.10: Giao diện thêm thông tin tài khoản.....	56
Hình 4.11 Giao diện phần thưởng cho khách hàng.....	57

Hình 4.12 Giao diện quà tặng cho khách hàng .....	58
Hình 4.13 Giao diện khuyến mãi.....	59
Hình 4.14 Giao diện thêm thông tin thanh toán tín dụng cho khách hàng .....	60
Hình 4.15 Giao diện quản lý đơn hàng.....	61
Hình 4.16 Giao diện đăng ký cho Nhà hàng.....	62
Hình 5.17 Giao diện Thêm món ăn .....	63
Hình 4.18 Giao diện Trang chủ cho nhà hàng .....	64
Hình 4.19 Giao diện Trang đăng ký cho tài xế .....	65
Hình 4.20 Giao diện Trang chủ cho tài xế.....	66

## DANH MỤC BẢNG BIỂU

Bảng 3.1 Mô tả vắn tắt các thực thể mô hình ERD.....	25
Bảng 3.2 Đặc tả Use Case: Đăng nhập.....	26
Bảng 3.3 Đặc tả Use Case: Đăng ký.....	27
Bảng 3.4 Đặc tả Use Case: Đăng xuất.....	28
Bảng 3.5 Đặc tả Use Case: Quản lý món ăn.....	30
Bảng 3.6 Đặc tả Use Case: Quản lý món ăn- Nhà hàng.....	32
Bảng 3.7 Đặc tả Use Case: Quản lý đơn hàng- Nhà hàng .....	34
Bảng 3.8 Đặc tả Use Case: Quản lý đơn hàng- Người vận chuyển .....	36

## **DANH MỤC CÁC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ**



# CHƯƠNG 1 TỔNG QUAN VỀ ĐỀ TÀI

## 1.1. Giới thiệu đề tài

Ứng dụng giao hàng và đặt đồ ăn trực tuyến là một nền tảng kỹ thuật số, cho phép người dùng đặt món ăn từ các nhà hàng và quán ăn thông qua điện thoại di động hoặc máy tính. Với sự phát triển mạnh mẽ của công nghệ, ứng dụng này đã mang lại nhiều tiện ích và cải tiến vượt bậc trong việc tiếp cận dịch vụ ăn uống.

Đề tài giao hàng đặt đồ ăn trực tuyến nhằm phát triển các tính năng theo nhu cầu phù hợp của người sử dụng từ đó hoàn thành từ các tính năng nhỏ của ứng dụng và phát triển thành một ứng dụng hoàn thiện. Ứng dụng này có thể phù hợp cho nhiều đối tượng, phần lớn người dùng là học sinh, sinh viên, người đi làm hoặc người lớn tuổi.

## 1.2. Mục tiêu đề tài

Mục tiêu tiện lợi cho người dùng:

- + Cải thiện trải nghiệm người dùng: Tạo ra một giao diện thân thiện, dễ sử dụng giúp người dùng dễ dàng tìm kiếm, lựa chọn, đặt món ăn.
- + Đa dạng lựa chọn: Cung cấp nhiều lựa chọn về nhà hàng, quán ăn và món ăn để đáp ứng mọi nhu cầu ẩm thực của người dùng.
- + Tăng cường tính tiện lợi: Đảm bảo quá trình đặt hàng và giao hàng diễn ra nhanh chóng chính xác và an toàn, giúp người dùng tiết kiệm thời gian và công sức.

Mục tiêu hỗ trợ doanh nghiệp:

- + Mở rộng thị trường: Giúp các nhà hàng và quán ăn tiếp cận nhiều khách hàng hơn, tăng doanh thu và phát triển kinh doanh.
- + Cải thiện dịch vụ: Cung cấp các công cụ và dữ liệu phân tích các doanh nghiệp hiểu rõ hơn về khách hàng từ đó nâng cao chất lượng dịch vụ và sản phẩm.
- + Tăng hiệu quả vận hành: Hỗ trợ quản lý đơn hàng, thanh toán và giao nhận hiệu quả hơn, giúp các doanh nghiệp tối ưu hóa quy trình hoạt động.

Mục tiêu phát triển công nghệ và đổi mới:

- + Tăng cường bảo mật: Đảm bảo an toàn thông tin và bảo mật dữ liệu người dùng trong quá trình giao dịch trực tuyến.
- + Đổi mới sáng tạo: Liên tục cải tiến và giới thiệu các tính năng để đáp ứng nhu cầu ngày càng cao của người dùng và theo kịp xu hướng thị trường.

Mục tiêu xã hội và môi trường:

- + Tạo cơ hội việc làm: Tạo ra nhiều công việc cho tài xế giao hàng và nhân viên nhà hàng, góp phần giải quyết vấn đề thất nghiệp và thúc đẩy kinh tế địa phương.
- + Bảo vệ môi trường: Thúc đẩy sáng kiến về giao hàng xanh, sử dụng các phương tiện thân thiện với môi trường và giảm thiểu lượng khí thải.
- + Hỗ trợ cộng đồng: Tham gia vào các hoạt động từ thiện, hỗ trợ cộng đồng và phát triển bền vững.

Mục tiêu tài chính:

- + Tăng trưởng doanh thu: Tối ưu hóa mô hình kinh doanh để đạt được doanh thu bền vững và lợi nhuận cao.
- + Mở rộng thị trường: Đầu tư và mở rộng thị trường, phát triển các khu vực mới và thu hút thêm người dùng.
- + Tối ưu chi phí: Quản lý và giảm thiểu chi phí vận hành để đạt được hiệu quả kinh doanh cao nhất.

### **1.3. Ý nghĩa đề tài**

Đối với người dùng:

- + Tiện lợi và tiết kiệm thời gian: Ứng dụng giúp người dùng tiết kiệm thời gian di chuyển và chờ đợi tại Nhà hàng. Chỉ cần vài thao tác trên điện thoại, họ có thể đặt món ăn yêu thích và nhận tại nhà hoặc nơi làm việc.
- + Sự lựa chọn đa dạng: Người dùng có thể tiếp cận nhiều nhà hàng và quán ăn khác

nhau, từ đó có nhiều sự lựa chọn về món ăn và phong cách ẩm thực phù hợp với sở thích cá nhân.

- + An toàn và tiện ích trong hoàn cảnh dịch bệnh phức tạp: Trong bối cảnh dịch bệnh, ứng dụng giúp giảm thiểu tiếp xúc trực tiếp và tuân thủ các biện pháp giãn cách xã hội, đảm bảo an toàn cho sức khỏe người dùng.

Đối với doanh nghiệp:

- + Mở rộng thị trường: Các nhà hàng và quán ăn có thể tiếp cận đến nhiều khách hàng hơn, kể cả những khách hàng ở xa hoặc không thể tiến đến trực tiếp nhà hàng, từ đó tăng doanh thu và mở rộng kinh doanh.
- + Cải thiện dịch vụ và chất lượng: Thông qua phản hồi và đánh giá của khách hàng trên ứng dụng, doanh nghiệp có thể nhận biết điểm mạnh yếu và không ngừng cải thiện chất lượng dịch vụ, hình ảnh món ăn.
- + Tăng khả năng cạnh tranh: Sự hiện diện trên nền tảng trực tuyến giúp các doanh nghiệp nhỏ và vừa cạnh tranh bình đẳng với các chuỗi nhà hàng lớn tạo cơ hội phát triển và khẳng định thương hiệu.

Đối với xã hội:

- + Thúc đẩy kinh tế số: Ứng dụng giao hàng, đặt đồ ăn trực tuyến góp phần thúc đẩy sự phát triển của nền kinh tế số, mang lại những lợi ích công nghệ và đổi mới sáng tạo.

#### **1.4. Phương pháp nghiên cứu**

Phương pháp khảo sát:

- + Khảo sát người dùng: Thu thập dữ liệu từ người dùng hiện tại và tiềm năng thông qua các bảng câu hỏi, phỏng vấn trực tiếp hoặc khảo sát trực tuyến để hiểu rõ nhu cầu, sở thích và mong muốn của họ.
- + Phân tích phản hồi: Thu thập và phân tích các đánh giá, phản hồi của người dùng trên ứng dụng để xác định các vấn đề cần cải thiện và điểm mạnh cần phát huy.

Phương pháp quan sát:

- + Quan sát hành vi khách hàng: Theo dõi và ghi nhận cách người dùng tương tác với ứng dụng trong môi trường thực tế để phát hiện các vấn đề trong trải nghiệm người dùng và tìm ra giải pháp cải thiện.
- + Quan sát thị trường: Quan sát và phân tích các hoạt động của đối thủ cạnh tranh và xu hướng thị trường để đưa ra các chiến lược kinh doanh phù hợp.

Phương pháp phân tích dữ liệu:

- + Phân tích dữ liệu sử dụng: Sử dụng các công cụ phân tích dữ liệu để thu thập và phân tích các dữ liệu từ ứng dụng, như lượt tải tần suất sử dụng, thời gian sử dụng, và các hành vi mua sắm người dùng.
- + Phân tích dữ liệu thị trường: Nghiên cứu dữ liệu thị trường để xác định các xu hướng ẩm thực, sở thích tiêu dùng và tiềm năng phát triển của ứng dụng trong từng khu vực cụ thể.

Phương pháp nghiên cứu thị trường:

- + Nghiên cứu định tính: Sử dụng các phương pháp như phỏng vấn sâu và nhóm tập trung để hiểu rõ hơn về tâm lý và hành vi của người dùng.
- + Nghiên cứu định lượng: Sử dụng các phương pháp thống kê và phân tích dữ liệu lớn để đưa ra các kết luận chính xác về thị trường và xu hướng tiêu dùng.

### **1.5. Phạm vi nghiên cứu**

Phạm vi nghiên cứu của ứng dụng giao hàng đặt đồ ăn trực tuyến cần được xác định một cách chi tiết và toàn diện, bao gồm phạm vi địa lý, đối tượng, chức năng, công nghệ, thời gian, kinh tế, pháp lý, xã hội và môi trường. Việc xác định rõ ràng phạm vi nghiên cứu giúp đảm bảo rằng dự án được triển khai hiệu quả, đáp ứng nhu cầu của người dùng và thị trường, đồng thời đảm bảo tính bền vững và phát triển lâu dài.

Phạm vi địa lý:

- + Khu vực nghiên cứu: Xác định cụ thể khu vực địa lý mà nghiên cứu sẽ được tiến

hành, chẳng hạn như thành phố, khu vực đô thị, nông thôn, hoặc quốc gia.

- + Quy mô thị trường: Đánh giá thị trường trong phạm vi địa lý đó, bao gồm số lượng nhà hàng, quán ăn, và tiềm năng khách hàng.

Phạm vi chức năng:

- + Chức năng chính của ứng dụng: Nghiên cứu và phát triển các chức năng cốt lõi như tìm kiếm nhà hàng, đặt món ăn, thanh toán trực tuyến, và theo dõi đơn hàng.
- + Chức năng phụ trợ: Các chức năng như đánh giá và phản hồi, chương trình khuyến mãi, tích điểm thưởng, và hỗ trợ khách hàng.

Phạm vi công nghệ:

- + Nền tảng: Xác định nền tảng mà ứng dụng sẽ hoạt động, bao gồm Android và IOS.
- + Công nghệ: Sử dụng chat công nghệ tiên tiến như chat Gemini để tối ưu hóa hoạt động.

Phạm vi thời gian

- + Giai đoạn nghiên cứu: Xác định các giai đoạn cụ thể của nghiên cứu từ khởi đầu, thử nghiệm, đến triển khai và đánh giá.
- + Kế hoạch dài hạn: Lên kế hoạch cho việc nâng cấp và cải tiến ứng dụng trong tương lai dựa trên phản hồi và xu hướng thị trường.

Phạm vi kinh tế

- + Phân tích chi phí: Xác định chi phí phát triển, vận hành, và tiếp thị ứng dụng.
- + Phân tích doanh thu: Dự đoán doanh thu từ các nguồn khác nhau như phí giao hàng, quảng cáo, và hợp tác với nhà hàng.

Phạm vi pháp lý và chính sách

- + Quy định pháp lý: Nghiên cứu các quy định pháp lý liên quan đến giao dịch trực tuyến, bảo mật thông tin người dùng, và quyền lợi khách hàng.

- + Chính sách: Đảm bảo tuân thủ các chính sách về an toàn thực phẩm, bảo vệ quyền lợi khách hàng và bảo mật dữ liệu.

#### Phạm vi xã hội và môi trường

- + Tác động xã hội: Đánh giá tác động của ứng dụng đối với cộng đồng, bao gồm việc tạo việc làm và thay đổi thói quen tiêu dùng.
- + Tác động môi trường: Nghiên cứu các biện pháp để giảm thiểu tác động môi trường, chẳng hạn như sử dụng bao bì thân thiện với môi trường và tối ưu hóa tuyến đường giao hàng để giảm khí thải.

### 1.6. Những đóng góp của đề tài

Ứng dụng giao hàng đặt đồ ăn trực tuyến không chỉ mang lại tiện ích cho người dùng mà còn đóng góp quan trọng cho các nhà hàng, tài xế giao hàng, xã hội, môi trường và sự phát triển công nghệ. Những đóng góp này không chỉ giúp cải thiện chất lượng cuộc sống mà còn thúc đẩy sự phát triển bền vững và đổi mới trong nhiều lĩnh vực khác nhau.

#### Đóng góp cho người dùng:

- + Tiện lợi và tiết kiệm thời gian: Ứng dụng giúp người dùng đặt món ăn một cách nhanh chóng và tiện lợi, tiết kiệm thời gian di chuyển và chờ đợi.
- + Đa dạng lựa chọn: Cung cấp nhiều lựa chọn về nhà hàng và món ăn, từ đó người dùng có thể dễ dàng tìm kiếm và thử nghiệm các món ăn mới.
- + Dịch vụ 24/7: Nhiều ứng dụng cung cấp dịch vụ đặt hàng và giao hàng 24/7, đáp ứng nhu cầu ăn uống bất cứ lúc nào, kể cả vào đêm khuya.

#### Đóng góp cho các nhà hàng và quán ăn

- + Mở rộng thị trường: Giúp các nhà hàng và quán ăn tiếp cận được nhiều khách hàng hơn, đặc biệt là những khách hàng không thể hoặc không muốn đến trực tiếp nhà hàng.
- + Tăng doanh thu: Tạo ra một kênh bán hàng mới, giúp tăng doanh thu và lợi nhuận.

cho các nhà hàng.

- + Phản hồi khách hàng: Thu thập phản hồi từ người dùng để cải thiện chất lượng dịch vụ và món ăn, từ đó nâng cao uy tín và sức cạnh tranh của nhà hàng.

Đóng góp cho tài xế giao hàng

- + Cơ hội việc làm: Tạo ra nhiều cơ hội việc làm cho người lao động, đặc biệt là những người cần công việc linh hoạt về thời gian.
- + Thu nhập ổn định: Cung cấp nguồn thu nhập ổn định cho tài xế giao hàng, giúp cải thiện đời sống kinh tế của họ.

Đóng góp cho xã hội

- + Thúc đẩy kinh tế số: Góp phần thúc đẩy sự phát triển của nền kinh tế số, ứng dụng công nghệ vào đời sống và kinh doanh.
- + Giảm tắc nghẽn giao thông: Khi người dùng chọn giao hàng thay vì di chuyển đến nhà hàng, điều này có thể giúp giảm tắc nghẽn giao thông, đặc biệt là ở các thành phố lớn.
- + An toàn trong thời gian dịch bệnh: Đáp ứng nhu cầu ăn uống mà không cần tiếp xúc trực tiếp, giảm nguy cơ lây nhiễm trong bối cảnh dịch bệnh.

Đóng góp cho môi trường

- + Giảm lượng khí thải: Khi nhiều đơn hàng được giao trong một chuyến đi, điều này có thể giảm lượng khí thải so với việc nhiều người tự lái xe đến nhà hàng.
- + Khuyến khích bao bì thân thiện môi trường: Nhiều ứng dụng khuyến khích hoặc yêu cầu nhà hàng sử dụng bao bì thân thiện với môi trường, góp phần giảm thiểu rác thải nhựa.

Đóng góp cho sự phát triển đổi mới công nghệ

- + Phát triển công nghệ mới: Thúc đẩy việc áp dụng và phát triển các công nghệ mới như trí tuệ nhân tạo (AI), học máy (ML), và dữ liệu lớn (Big Data) để cải thiện trải

nghiệm người dùng và hiệu quả vận hành.

- + Đổi mới trong lĩnh vực dịch vụ: Khuyến khích sự đổi mới trong lĩnh vực dịch vụ ăn uống và logistics, tạo ra các giải pháp sáng tạo và hiệu quả hơn.

### **1.7. Phạm vi sử dụng**

Người dùng, bao gồm tất cả các đối tượng như học sinh, sinh viên, người đi làm, người lớn tuổi ...., có thể truy cập và sử dụng ứng dụng giao hàng và đặt đồ ăn trực tuyến UBEREATS trên các thiết bị thông minh như điện thoại, máy tính, laptop với kết nối mạng để dễ dàng quản lý các thông tin về đồ ăn và các thông tin liên quan.

Phạm vi sử dụng của ứng dụng giao hàng đặt đồ ăn trực tuyến rất rộng, bao gồm nhiều địa điểm, đối tượng người dùng, nhà hàng, loại món ăn và dịch vụ khác nhau. Sự đa dạng này giúp ứng dụng đáp ứng nhu cầu của người dùng trong nhiều tình huống và môi trường khác nhau, từ đô thị đến nông thôn, từ cá nhân đến doanh nghiệp, và từ các nhà hàng sang trọng đến các dịch vụ ẩm thực đường phố.

Phạm vi địa lý

- + Đô thị và thành phố lớn: Ứng dụng phổ biến ở các đô thị và thành phố lớn nơi có nhiều nhà hàng và nhu cầu đặt đồ ăn cao.
- + Khu vực ngoại thành và nông thôn: Ứng dụng dần mở rộng phạm vi hoạt động đến các khu vực ngoại thành và nông thôn để phục vụ nhu cầu của người dân tại các vùng này.

Phạm vi người dùng

- + Cá nhân: Người dùng cá nhân có thể đặt món ăn cho riêng mình hoặc gia đình, từ các bữa ăn hàng ngày đến các bữa tiệc hoặc sự kiện đặc biệt.
- + Nhóm bạn bè: Ứng dụng hỗ trợ đặt đồ ăn cho nhóm bạn bè trong các buổi gặp gỡ, liên hoan, hoặc tụ tập.
- + Doanh nghiệp: Các công ty và tổ chức có thể sử dụng ứng dụng để đặt đồ ăn cho nhân viên trong các cuộc họp, sự kiện hoặc bữa trưa hàng ngày.



### Phạm vi nhà hàng và quán ăn

- + Nhà hàng sang trọng: Các nhà hàng cao cấp có thể sử dụng ứng dụng để tiếp cận thêm nhiều khách hàng và tăng doanh thu.
- + Quán ăn nhanh: Các quán ăn nhanh có thể tận dụng ứng dụng để phục vụ khách hàng nhanh chóng và hiệu quả hơn.
- + Quán cà phê và tiệm bánh: Ứng dụng cũng hỗ trợ các quán cà phê và tiệm bánh trong việc giao hàng các loại đồ uống, bánh ngọt và món ăn nhẹ.
- + Dịch vụ ẩm thực đường phố: Các dịch vụ ẩm thực đường phố có thể tiếp cận nhiều khách hàng hơn thông qua ứng dụng.

### Phạm vi thời gian

- + 24/7: Nhiều ứng dụng cung cấp dịch vụ đặt và giao hàng 24/7, đáp ứng nhu cầu ăn uống bất cứ lúc nào, kể cả vào đêm khuya hoặc sáng sớm.
- + Đặt trước: Người dùng có thể đặt món ăn trước cho các bữa tiệc, sự kiện hoặc các bữa ăn hàng ngày vào thời gian cụ thể.

### Phạm vi dịch vụ

Giao hàng tận nơi: Dịch vụ giao hàng tận nơi giúp người dùng nhận được món ăn tại nhà, nơi làm việc hoặc bất kỳ địa điểm nào mong muốn.

### Phạm vi loại món ăn

- + Ẩm thực quốc tế: Ứng dụng hỗ trợ đặt các món ăn từ nhiều quốc gia khác nhau như Ý, Nhật Bản, Hàn Quốc, Thái Lan, Ấn Độ, và nhiều quốc gia khác.
- + Ẩm thực địa phương: Người dùng có thể thưởng thức các món ăn đặc sản địa phương mà không cần phải đến trực tiếp các nhà hàng truyền thống.
- + Thức ăn nhanh: Các món ăn nhanh như pizza, burger, sandwich, gà rán và nhiều loại thức ăn nhanh khác.
- + Đồ uống: Đặt các loại đồ uống như cà phê, trà sữa, sinh tố, nước ép và nhiều loại

đồ uống khác.

## **1.8. Flutter**

### ***1.8.1. Giới thiệu về Flutter***

Ngày ra mắt: Flutter được công bố lần đầu tiên vào tháng 5 năm 2017.

Ngôn ngữ sử dụng: Flutter sử dụng Dart làm ngôn ngữ lập trình chính.

Mã nguồn mở: Flutter là một dự án mã nguồn mở và có cộng đồng đóng góp rộng rãi.

### ***1.8.2. Đặc điểm chính của Flutter***

Giao diện người dùng tùy chỉnh: Flutter cho phép tạo ra các giao diện người dùng đẹp mắt và tùy chỉnh dễ dàng với sự linh hoạt cao.

Hiệu suất cao: Flutter sử dụng cơ chế biên dịch Ahead-of-Time (AOT) cho phép chạy ứng dụng với hiệu suất cao gần như là ứng dụng gốc (native).

Hot Reload: Tính năng hot reload giúp các nhà phát triển nhanh chóng thấy được kết quả của các thay đổi trong mã nguồn mà không cần khởi động lại toàn bộ ứng dụng.

Widget: Flutter cung cấp một thư viện widget phong phú và mạnh mẽ, giúp việc xây dựng giao diện người dùng trở nên đơn giản và hiệu quả.

Đa nền tảng: Flutter hỗ trợ phát triển ứng dụng cho nhiều nền tảng khác nhau bao gồm iOS, Android, web và máy tính để bàn (Windows, macOS, Linux).

### ***1.8.3. Ưu điểm của Flutter***

Một mã nguồn duy nhất: Sử dụng cùng một mã nguồn để phát triển ứng dụng cho nhiều nền tảng, tiết kiệm thời gian và công sức của các nhà phát triển.

Giao diện đẹp mắt: Flutter cho phép tạo ra các giao diện người dùng đẹp mắt với các hiệu ứng chuyển động mượt mà và tùy chỉnh cao.

Tốc độ phát triển nhanh: Tính năng hot reload và thư viện widget phong phú giúp tăng tốc quá trình phát triển và thử nghiệm.

Cộng đồng và tài liệu: Flutter có một cộng đồng lớn và tài liệu phong phú, hỗ trợ các

nhà phát triển học và làm việc với công cụ này.

#### ***1.8.4. Kiến trúc của Flutter***

Flutter Engine: Là lõi của Flutter, được viết bằng C++, chịu trách nhiệm render giao diện và quản lý các hoạt động liên quan đến đồ họa.

Framework: Là phần trên của Flutter Engine, bao gồm các thư viện Dart như material, cupertino và các tiện ích khác để xây dựng giao diện người dùng.

Embedder: Là lớp giao tiếp giữa Flutter và nền tảng gốc (iOS, Android, web, máy tính để bàn), giúp tích hợp Flutter vào các ứng dụng gốc

#### ***1.8.5. Ứng dụng thực tế của Flutter***

Ứng dụng di động: Nhiều ứng dụng di động nổi tiếng được xây dựng bằng Flutter như Google Ads, Alibaba, và Reflectly.

Ứng dụng web: Flutter for Web cho phép các nhà phát triển tạo ra các ứng dụng web từ mã nguồn Flutter.

Ứng dụng máy tính để bàn: Flutter cũng hỗ trợ phát triển ứng dụng cho các hệ điều hành máy tính để bàn như Windows, macOS và Linux.

#### ***1.8.6. So sánh với các công cụ khác***

React Native: Flutter và React Native đều là công cụ phát triển đa nền tảng. Tuy nhiên, Flutter sử dụng Dart trong khi React Native sử dụng JavaScript và JSX. Flutter cung cấp một bộ widget tích hợp sẵn, trong khi React Native phụ thuộc nhiều vào các thư viện bên ngoài.

Xamarin: Xamarin sử dụng C# và .NET framework để phát triển ứng dụng đa nền tảng. Flutter có ưu thế về hiệu suất và sự dễ dàng trong việc tùy chỉnh giao diện người dùng.

#### ***1.8.7. Ưu điểm và nhược điểm của Flutter***

Ưu điểm:

Đa nền tảng từ một mã nguồn duy nhất:

Tiết kiệm thời gian và chi phí: Flutter cho phép phát triển ứng dụng cho nhiều nền tảng (iOS, Android, web, và máy tính để bàn) từ một mã nguồn duy nhất, giúp tiết kiệm thời gian và chi phí phát triển.

Dễ dàng bảo trì: Việc duy trì và cập nhật mã nguồn trở nên dễ dàng hơn do chỉ cần quản lý một bộ mã duy nhất.

Hiệu suất cao:

Biên dịch Ahead-of-Time (AOT): Flutter sử dụng biên dịch Ahead-of-Time (AOT) thành mã máy gốc, giúp cải thiện hiệu suất và tốc độ thực thi của ứng dụng.

Trực tiếp vẽ giao diện: Flutter vẽ giao diện trực tiếp lên canvas thay vì sử dụng các thành phần gốc, giúp kiểm soát tốt hơn về hiệu suất và giao diện.

Tăng tốc phát triển: Tính năng hot reload cho phép các nhà phát triển nhanh chóng thấy được kết quả của các thay đổi trong mã nguồn mà không cần khởi động lại toàn bộ ứng dụng, tăng tốc quá trình phát triển và thử nghiệm.

Thư viện widget phong phú: Flutter cung cấp một thư viện widget phong phú và mạnh mẽ, giúp việc xây dựng giao diện người dùng trở nên đơn giản và hiệu quả.

Hiệu ứng mượt mà: Flutter hỗ trợ các hiệu ứng chuyển động và hoạt ảnh mượt mà, giúp tạo ra các giao diện người dùng ấn tượng.

Tài liệu phong phú: Flutter có tài liệu chính thức phong phú và dễ hiểu, cùng với nhiều khóa học trực tuyến và ví dụ mã nguồn.

Cộng đồng lớn: Flutter có một cộng đồng phát triển rộng lớn và năng động, cung cấp nhiều gói và thư viện mở rộng.

Nhược điểm

Kích thước ban đầu: Ứng dụng Flutter có thể có kích thước ban đầu lớn hơn so với các ứng dụng native do tích hợp thư viện Flutter engine bên trong.

Thư viện bên thứ ba: Mặc dù cộng đồng Flutter đang phát triển nhanh chóng, nhưng vẫn có một số thư viện và SDK gốc chưa được hỗ trợ đầy đủ. Điều này có thể yêu cầu

nhà phát triển phải tự viết cầu nối (plugin) để tương tác với các chức năng gốc của hệ điều hành.

Tích hợp với ứng dụng hiện có: Việc tích hợp Flutter vào các ứng dụng hiện có hoặc làm việc với mã gốc phức tạp có thể đòi hỏi kiến thức sâu rộng về cả Flutter và nền tảng gốc.

Tính năng mới của hệ điều hành: Flutter có thể bị chậm trễ trong việc hỗ trợ các tính năng mới nhất của hệ điều hành vì phải chờ đội ngũ phát triển Flutter hoặc cộng đồng tạo ra các gói hỗ trợ.

Nguồn tài liệu: Dù có cộng đồng lớn, nhưng tài liệu và nguồn tài nguyên tiếng Việt về Flutter vẫn còn hạn chế, điều này có thể gây khó khăn cho những người không thành thạo tiếng Anh.

## **1.9. Dart**

### ***1.9.1. Giới thiệu về Dart***

Mục đích: Dart được thiết kế để xây dựng các ứng dụng hiện đại cho nhiều nền tảng khác nhau, bao gồm web, máy tính để bàn và di động.

Nhà phát triển: Dart được phát triển và duy trì bởi Google.

Ngày ra mắt: Dart được công bố lần đầu tiên vào tháng 10 năm 2011

### ***1.9.2. Đặc điểm của Dart***

Cú pháp: Dart có cú pháp đơn giản và dễ học, tương tự như các ngôn ngữ lập trình phổ biến khác như JavaScript, Java, và C#.

Hướng đối tượng: Dart là ngôn ngữ lập trình hướng đối tượng, hỗ trợ đầy đủ các tính năng của lập trình hướng đối tượng như lớp, kế thừa, và đa hình.

Trình biên dịch: Dart có thể được biên dịch thành mã máy (native code) hoặc JavaScript để chạy trên các trình duyệt web.

Hỗ trợ phát triển đa nền tảng: Dart hỗ trợ phát triển ứng dụng cho nhiều nền tảng khác nhau, bao gồm web, iOS, Android và máy tính để bàn

### ***1.9.3. Dart VM (Virtual Machine)***

Chạy Dart natively: Dart VM cho phép chạy mã Dart trực tiếp mà không cần phải biên dịch sang ngôn ngữ khác.

Hot reload: Dart VM hỗ trợ tính năng hot reload, giúp các nhà phát triển nhanh chóng thấy được kết quả của các thay đổi trong mã nguồn mà không cần phải khởi động lại toàn bộ ứng dụng.

### ***1.9.4. Flutter và Dart***

Chạy Dart natively: Dart VM cho phép chạy mã Dart trực tiếp mà không cần phải biên dịch sang ngôn ngữ khác.

Hot reload: Dart VM hỗ trợ tính năng hot reload, giúp các nhà phát triển nhanh chóng thấy được kết quả của các thay đổi trong mã nguồn mà không cần phải khởi động lại toàn bộ ứng dụng.

### ***1.9.5. Cộng đồng và tài nguyên***

Cộng đồng: Dart có một cộng đồng lớn và phát triển mạnh mẽ, với nhiều nhà phát triển đóng góp vào sự phát triển và cải tiến của ngôn ngữ.

Tài liệu: Dart có tài liệu phong phú và dễ hiểu, cùng với nhiều khóa học trực tuyến và ví dụ mã nguồn để hỗ trợ các nhà phát triển học và làm việc với ngôn ngữ này.

### ***1.9.6. Ứng dụng thực tế của Dart***

Ứng dụng di động: Dart, thông qua Flutter, được sử dụng để phát triển các ứng dụng di động hiệu suất cao cho cả iOS và Android.

Ứng dụng web: Dart có thể biên dịch sang JavaScript để chạy trên các trình duyệt web, giúp xây dựng các ứng dụng web phức tạp và tương tác.

Ứng dụng máy tính để bàn: Dart hỗ trợ phát triển các ứng dụng máy tính để bàn cho Windows, macOS và Linux.

### ***1.9.7. Ưu nhược điểm của Dart***

Ưu điểm của Dart

Hiệu suất cao:

Biên dịch Ahead-of-Time (AOT): Dart có thể biên dịch Ahead-of-Time (AOT) thành mã máy, giúp cải thiện hiệu suất và tốc độ thực thi của ứng dụng.

Hot Reload: Dart hỗ trợ tính năng hot reload khi sử dụng với Flutter, giúp các nhà phát triển nhanh chóng thấy được kết quả của các thay đổi mà không cần khởi động lại toàn bộ ứng dụng.

Hỗ trợ phát triển đa nền tảng:

Flutter: Dart là ngôn ngữ chính của Flutter, bộ công cụ phát triển UI của Google, cho phép phát triển ứng dụng cho cả iOS, Android, web và máy tính để bàn từ cùng một mã nguồn.

Web: Dart có thể được biên dịch sang JavaScript để chạy trên các trình duyệt web.

Cú pháp dễ học và sử dụng:

Tương tự các ngôn ngữ phổ biến: Cú pháp của Dart tương tự như JavaScript, Java và C#, giúp các nhà phát triển dễ dàng học và chuyển đổi sang Dart.

Hỗ trợ lập trình hướng đối tượng: Dart là ngôn ngữ lập trình hướng đối tượng, hỗ trợ đầy đủ các tính năng như lớp, kế thừa, và đa hình.

Thư viện và công cụ mạnh mẽ:

Thư viện phong phú: Dart có hệ sinh thái thư viện phong phú, giúp các nhà phát triển dễ dàng tìm kiếm và sử dụng các thư viện phù hợp cho dự án của mình.

Công cụ phát triển: Dart cung cấp nhiều công cụ hỗ trợ phát triển như Dart Analyzer, Dart DevTools và Dart Observatory, giúp tối ưu hóa quá trình phát triển và gỡ lỗi.

Bảo mật và an toàn:

Null Safety: Dart hỗ trợ null safety, giúp giảm thiểu các lỗi null pointer và cải thiện độ an toàn của mã nguồn.

Nhược điểm của Dart

Độ phổ biến hạn chế:

Cộng đồng nhỏ hơn: So với các ngôn ngữ lập trình phổ biến khác như JavaScript hay Python, Dart có cộng đồng nhỏ hơn, điều này có thể làm giảm khả năng tìm kiếm tài liệu và hỗ trợ từ cộng đồng.

Ít dự án mã nguồn mở: Dart có ít dự án mã nguồn mở lớn hơn so với một số ngôn ngữ khác, điều này có thể làm giảm sự phong phú trong việc học hỏi từ các dự án mẫu.

Tích hợp hạn chế:

Tương thích hạn chế: Mặc dù Dart có thể biên dịch sang JavaScript, nhưng vẫn có một số giới hạn trong việc tương thích với các thư viện JavaScript phổ biến và các công cụ phát triển web khác.

Thời gian học tập:

Đối với người mới: Mặc dù Dart có cú pháp dễ học, nhưng đối với những người mới bắt đầu lập trình hoặc chưa quen thuộc với các ngôn ngữ hướng đối tượng, việc học Dart có thể đòi hỏi thời gian và nỗ lực.

Độ trưởng thành:

Ngôn ngữ trẻ: Dart là một ngôn ngữ tương đối mới so với các ngôn ngữ lập trình khác như Java hoặc C++, do đó có thể thiếu một số tính năng và tối ưu hóa mà các ngôn ngữ đã trưởng thành hơn có.

## **1.10. Firebase**

### ***1.10.1. Giới thiệu về Firebase***

Firebase là một nền tảng của Google cung cấp các dịch vụ điện toán đám mây (cloud services) cho việc phát triển và quản lý ứng dụng di động và web. Được ra mắt vào năm 2011 và ngày càng phát triển mạnh mẽ, Firebase bao gồm nhiều công cụ và dịch vụ giúp cho việc xây dựng và triển khai các ứng dụng trở nên dễ dàng và nhanh chóng.

### ***1.10.2. Các tính năng chính của Firebase***

Realtime Database: Cơ sở dữ liệu thời gian thực cho phép đồng bộ dữ liệu trực tiếp



giữa các client và server mà không cần cấu hình server.

Authentication: Dịch vụ xác thực người dùng bằng nhiều phương thức như email/password, OAuth providers (Google, Facebook, Twitter), và các dịch vụ xác thực di động (SMS).

Cloud Firestore: Cơ sở dữ liệu linh hoạt hơn cho phép lưu trữ và truy vấn dữ liệu có cấu trúc bằng các tài liệu (documents) và bộ sưu tập (collections).

Storage: Lưu trữ đối tượng như hình ảnh, video, và tệp âm thanh trên cloud với khả năng quản lý và cung cấp quyền truy cập.

Hosting: Dịch vụ hosting web tĩnh cho phép phát triển và triển khai các ứng dụng web một cách dễ dàng.

Cloud Functions: Viết và triển khai các hàm (functions) backend mà không cần quản lý máy chủ (serverless), kết hợp với các sự kiện từ Firebase và Google Cloud.

Analytics: Theo dõi và phân tích dữ liệu về người dùng và sự tương tác của họ với ứng dụng để cải thiện trải nghiệm người dùng và hiệu quả của ứng dụng.

Performance Monitoring: Theo dõi hiệu suất của ứng dụng và xác định các vấn đề về tốc độ và thời gian phản hồi.

Remote Config: Quản lý các cài đặt và tính năng của ứng dụng từ xa mà không cần cập nhật ứng dụng.

ML Kit: Dịch vụ học máy để tích hợp các tính năng AI như phân tích hình ảnh, nhận diện văn bản, và phát hiện khuôn mặt vào ứng dụng một cách dễ dàng.

### ***1.10.3. Lợi ích của Firebase***

Nhanh chóng và dễ sử dụng: Firebase cung cấp các dịch vụ trên cloud sẵn sàng sử dụng và dễ tích hợp vào ứng dụng.

Tích hợp tốt với các nền tảng: Hỗ trợ đa nền tảng cho iOS, Android, và web, giúp cho việc phát triển ứng dụng đa nền tảng trở nên thuận lợi.

Bảo mật và tin cậy: Firebase cung cấp các cơ chế bảo mật mạnh mẽ và bảo vệ dữ liệu

của người dùng.

Mở rộng linh hoạt: Khả năng mở rộng với nhu cầu tăng trưởng của ứng dụng mà không cần lo lắng về việc quản lý cơ sở hạ tầng.

#### ***1.10.4. Nhược điểm của Firebase***

Giới hạn về tùy chỉnh: Mặc dù Firebase cung cấp nhiều tính năng, nhưng đôi khi có thể gặp hạn chế về tùy chỉnh nếu ứng dụng yêu cầu tính linh hoạt cao hơn.

Phí dịch vụ: Các dịch vụ Firebase có thể tạo chi phí phụ thuộc vào tài nguyên sử dụng và quy mô của ứng dụng.

Ràng buộc về vendor lock-in: Sử dụng Firebase có thể làm cho ứng dụng phụ thuộc nhiều vào nền tảng của Google, có thể làm cho việc chuyển đổi sang dịch vụ khác khó khăn hơn.

## **CHƯƠNG 2 KHẢO SÁT HỆ THỐNG**

### **2.1. Yêu cầu chung về phần mềm.**

Ứng dụng đặt đồ ăn trực tuyến bao gồm 3 ứng dụng chính:

#### ***2.1.1. Trang dành cho khách hàng:***

Người dùng (User) sẽ truy cập vào ứng dụng này để tiến hành các thao tác đăng ký thông tin tài khoản, tìm kiếm đồ ăn và đặt đồ ăn.

Người dùng bắt buộc phải đăng nhập để vào ứng dụng nếu muốn sử dụng các chức năng như tìm kiếm món ăn, đặt đồ ăn, xem giỏ hàng.

#### ***2.1.2. Trang dành cho Nhà hàng:***

Người quản lý nhà hàng hay chủ nhà hàng sẽ truy cập vào ứng dụng này để tiến hành các thao tác như đăng ký thông tin tài khoản, đăng quảng cáo, đăng hình món ăn, cập nhật giá cả, tên món ăn hay số lượng món ăn, kiểm tra doanh thu...

Tương tự người quản lý nhà hàng hay chủ nhà hàng phải buộc đăng nhập nhà hàng mới thực hiện các chức năng chính trên.

#### ***2.1.3. Ứng dụng giành cho bên phía vận chuyển***

Người bên phía vận chuyển sẽ truy cập ứng dụng này để tiến hành các thao tác như đăng ký thông tin tài khoản, cập nhật trạng thái “Nhận đơn”, “Hoàn thành đơn”, kiểm tra phí vận chuyển, cập nhật vị trí trên bản đồ,...

Tương tự người bên phía vận chuyển cũng sẽ đăng nhập để thực hiện các chức năng chính trên.

### **2.2. Yêu cầu chức năng**

#### ***2.2.1 Đối với ứng dụng đặt đồ ăn trực tuyến giành cho người dùng:***

**Tìm kiếm món ăn:** Chức năng này giúp cho người dùng có thể tìm thấy các món ăn phù hợp theo sở thích cá nhân và mong muốn của mình. Có thể tìm kiếm bằng tên món

ăn hoặc tên nhà hàng.

Đặt đồ ăn: Chức năng này giúp cho người dùng trực tiếp đặt đồ ăn thông qua ứng dụng, đồ ăn được đặt sẽ chuyển qua giỏ hàng.

Thanh toán: Chức năng này sẽ giúp cho người dùng thanh toán các món ăn đã đặt trong giỏ hàng qua 2 hình thức thanh toán trực tiếp hoặc qua ví điện tử.

Đăng ký: Người dùng nhập các thông tin cá nhân và hình ảnh để đăng ký thành công tài khoản của mình, thông tin cá nhân của người dùng sẽ được bảo mật vào lưu trên dữ liệu hệ thống.

Đăng nhập: Sau khi người dùng đăng ký thành công, người dùng sẽ nhập số điện thoại và 6 chữ số để đăng nhập thành công.

Đăng xuất: Người dùng có thể đăng xuất tài khoản của mình trên ứng dụng.

### ***2.2.2. Đối với ứng dụng giao hàng đồ ăn trực tuyến cho người quản lý nhà hàng.***

Đăng ký: Tương tự người quản lý hay chủ nhà hàng cũng phải nhập các thông tin cá nhân nhà hàng và hình ảnh nhà hàng để đăng ký thành công.

Đăng nhập: Người quản lý nhà hàng hay chủ nhà hàng phải đăng nhập thành công để sử dụng các năng chính của ứng dụng giành cho nhà hàng.

Thêm món ăn: Người quản lý nhà hàng hay chủ nhà hàng có thể thêm các món ăn trực tiếp trên ứng dụng cho nhà hàng.

Kiểm tra, thống kê doanh thu: Người quản lý nhà hàng, hay chủ nhà hàng có thể kiểm tra doanh thu cá nhân ngay trên ứng dụng.

Đăng xuất: Người quản lý nhà hàng hay chủ nhà hàng có thể đăng xuất tài khoản được tự động lưu trên ứng dụng.

### ***2.2.3. Đối với ứng dụng giao hàng đồ ăn trực tuyến cho bên phía vận chuyển.***

Đăng ký: Tương tự người giao hàng cũng phải nhập đủ các thông tin cá nhân liên quan để đăng ký tài khoản thành công.

Đăng nhập: Người vận chuyển, giao hàng phải đăng nhập thành công để nhận đơn hàng và giao đơn hàng.

Đăng xuất: Người vận chuyển, giao hàng có thể đăng xuất tài khoản trên ứng dụng.

## **2.3. Yêu cầu phi chức năng**

### **2.3.1. Hiệu suất (Performance)**

Thời gian phản hồi: Ứng dụng phải phản hồi yêu cầu của người dùng trong vòng 2 giây.

Tải lớn: Hệ thống phải hỗ trợ ít nhất 10,000 người dùng đồng thời mà không làm giảm hiệu suất.

### **2.3.2. Khả năng mở rộng (Scalability)**

Khả năng mở rộng theo chiều dọc: Ứng dụng phải hỗ trợ tăng tài nguyên hệ thống như CPU, RAM để cải thiện hiệu suất.

Khả năng mở rộng theo chiều ngang: Ứng dụng phải có khả năng thêm nhiều máy chủ để phân tải và duy trì hiệu suất.

### **2.3.3. Bảo mật (Security)**

Xác thực và phân quyền: Ứng dụng phải sử dụng cơ chế xác thực và phân quyền chặt chẽ để bảo vệ dữ liệu.

Mã hóa dữ liệu: Tất cả dữ liệu nhạy cảm phải được mã hóa trong khi lưu trữ và truyền tải.

### **2.3.4. Khả năng sử dụng (Usability)**

Giao diện người dùng: Ứng dụng phải có giao diện thân thiện với người dùng, dễ điều hướng và sử dụng.

Tài liệu hướng dẫn: Cung cấp tài liệu hướng dẫn chi tiết và rõ ràng cho người dùng.

### **2.3.5. Khả năng bảo trì (Maintainability)**

Kiến trúc hệ thống: Ứng dụng phải được xây dựng với kiến trúc module hóa để dễ dàng bảo trì và nâng cấp.

Quản lý phiên bản: Sử dụng hệ thống quản lý phiên bản để theo dõi các thay đổi và phiên bản phần mềm.

#### **2.3.6. Độ tin cậy (Reliability)**

Khả năng khôi phục: Hệ thống phải có khả năng khôi phục nhanh chóng sau khi gặp sự cố.

Độ sẵn sàng: Ứng dụng phải có độ sẵn sàng ít nhất 99.9%, nghĩa là thời gian ngừng hoạt động không quá 8.76 giờ mỗi năm.

#### **2.3.7. Tính tương thích (Compatibility)**

Tương thích trình duyệt: Ứng dụng phải hoạt động mượt mà trên tất cả các trình duyệt phổ biến như Chrome, Firefox, Safari, và Edge.

Tương thích thiết bị: Ứng dụng phải tương thích với nhiều thiết bị khác nhau, bao gồm máy tính bàn, laptop, và thiết bị di động.

#### **2.3.8. Khả năng mở rộng quốc tế (Internationalization and Localization)**

Ngôn ngữ: Ứng dụng phải hỗ trợ nhiều ngôn ngữ và cho phép người dùng dễ dàng chuyển đổi giữa các ngôn ngữ.

Định dạng ngày và giờ: Ứng dụng phải hỗ trợ các định dạng ngày giờ khác nhau dựa trên địa phương của người dùng.

### **2.4. Ràng buộc thiết kế**

#### **2.4.1. Công nghệ sử dụng**

Front-end: Dart

Back-end: Java

Database: Firebase

### **2.5. Đối tượng người dùng**

Ứng dụng đặt đồ ăn trực tuyến hướng tới tất cả các đối tượng người dùng, phục vụ tất cả các đối tượng người dùng.

## **2.6. Các ràng buộc**

Không đòi hỏi cao ở các cấu hình thiết bị di động.

Yêu cầu là thiết bị di động xài hệ điều hành Android hoặc IOS, được cập nhật phiên bản mới nhất.

Yêu cầu có thiết bị có kết nối Internet để được phục vụ tốt hơn.

## **2.7. Giả định và phụ thuộc**

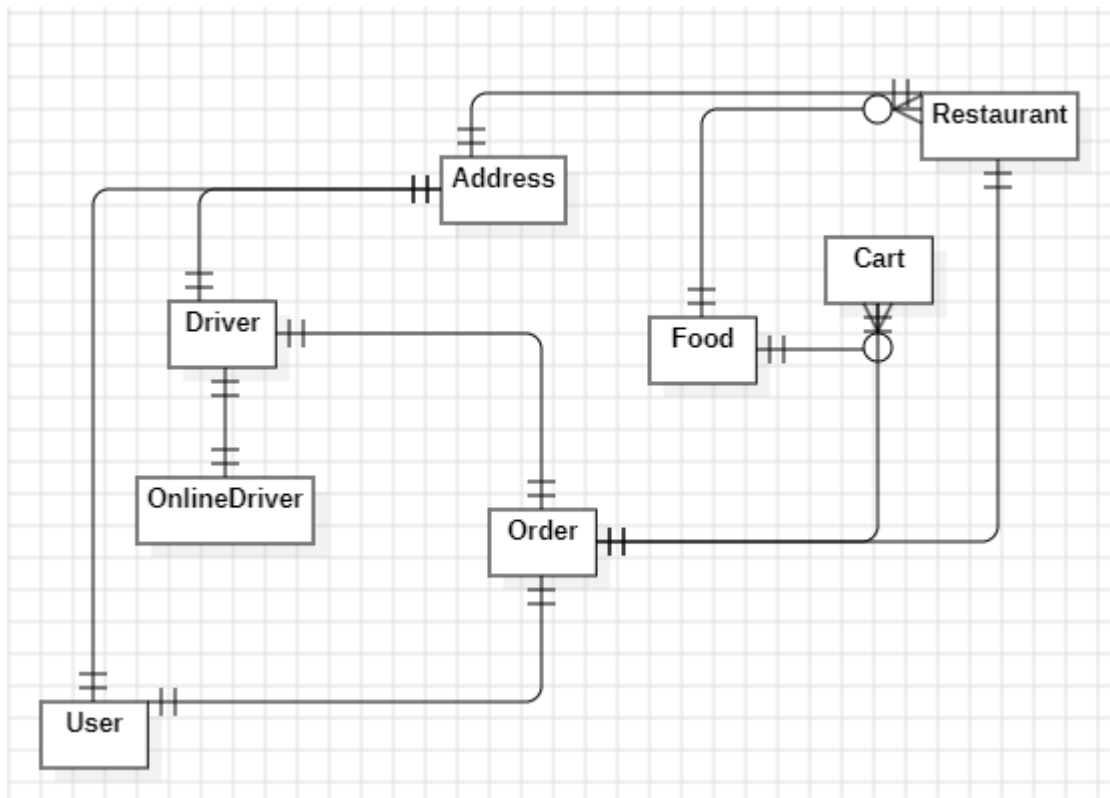
Ứng dụng chỉ được sử dụng ở thiết bị di động chạy hệ điều hành Android , IOS.

Yêu cầu có kết nối mạng Wi-fi hoặc dữ liệu di động để truy cập ứng dụng

## CHƯƠNG 3 PHÂN TÍCH HỆ THỐNG

### 3.1. Thiết kế cơ sở dữ liệu

#### 3.1.1. Mô hình ERD



Hình 3.1 Mô hình ERD

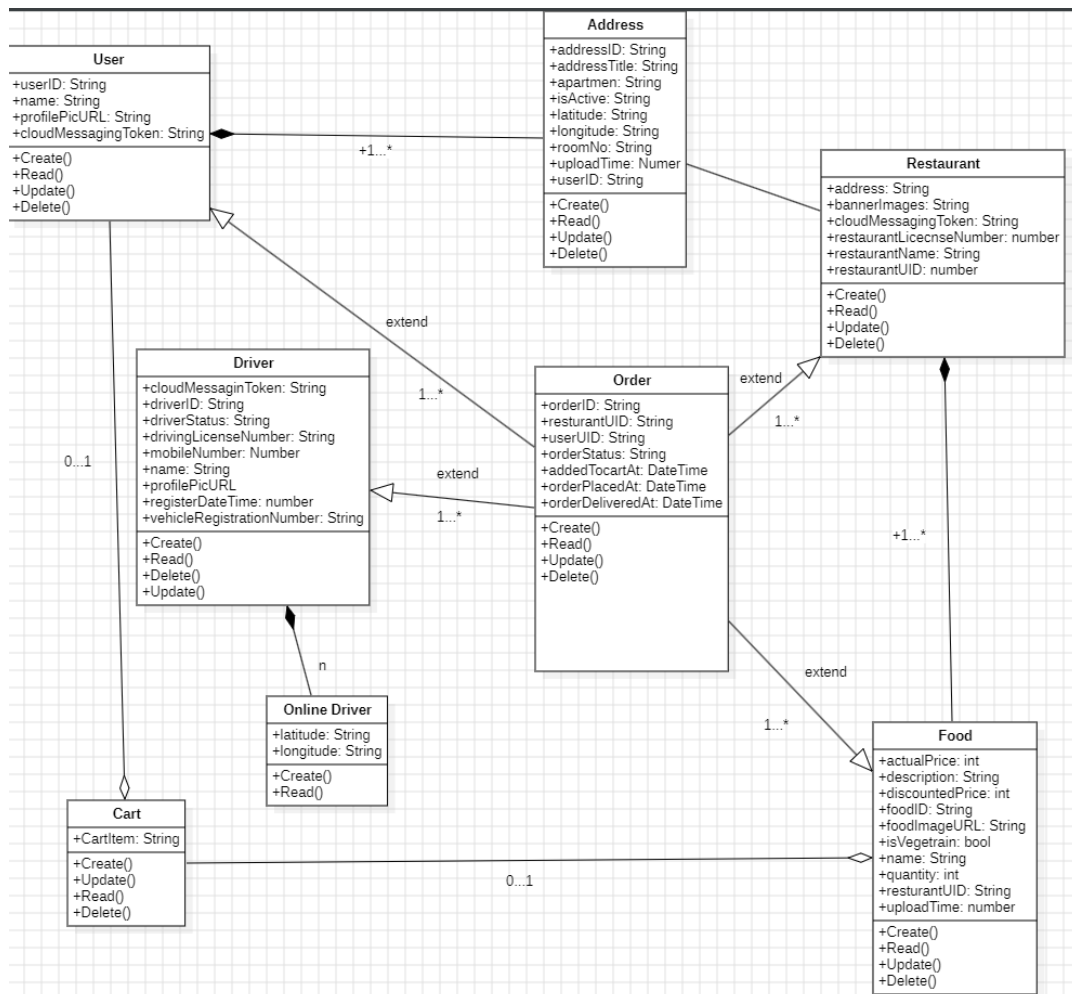


### 3.1.2. Mô tả vắn tắt các thực thể

Bảng 3.1 Mô tả vắn tắt các thực thể mô hình ERD

STT	Tên bảng	Mô tả
1	Order	Đơn hàng
2	User	Người dùng, khách hàng
3	Online Driver	Tài xế, người vận chuyển đơn hàng đang trực tuyến
4	Address	Địa chỉ của người dùng, khách hàng
5	Cart	Giỏ hàng
6	Restaurant	Nhà hàng
7	Driver	Tài xế, người vận chuyển đơn hàng đã đăng ký
8	Food	Món ăn từ Nhà hàng

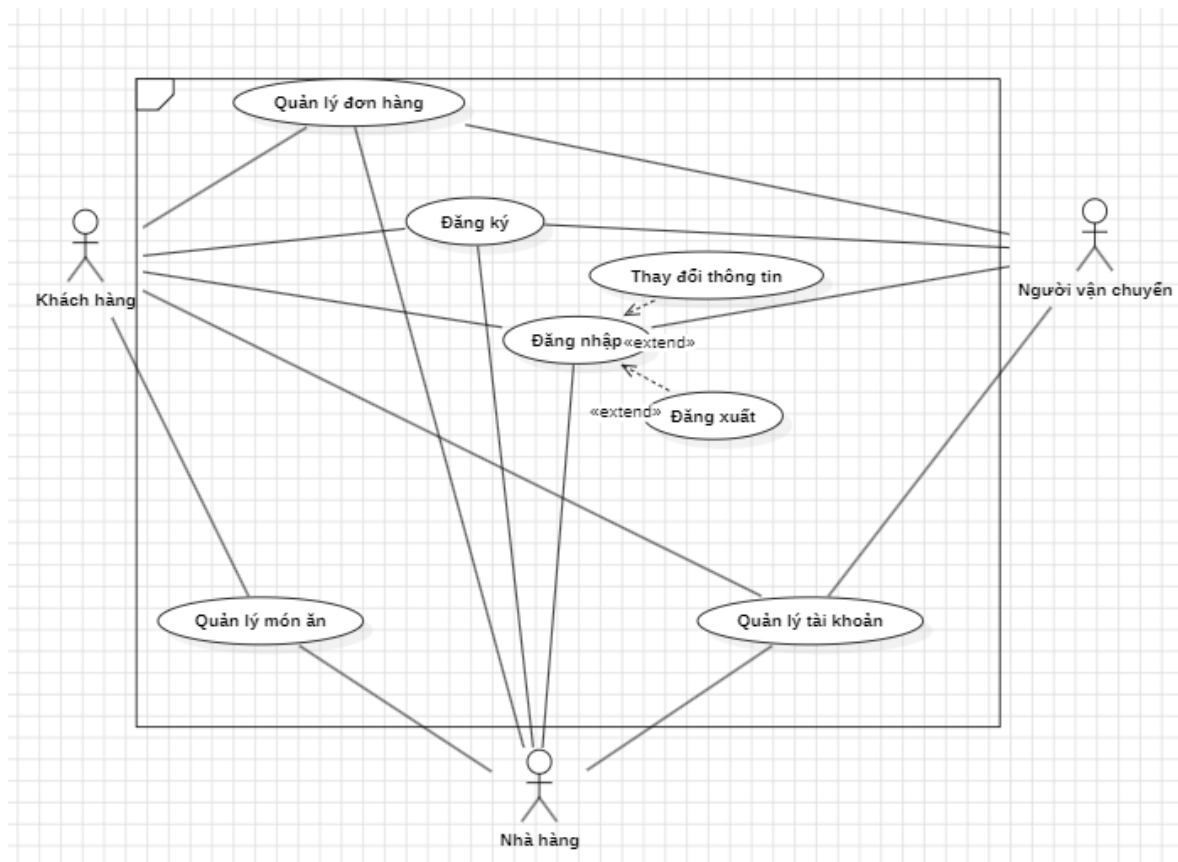
### 3.2. Mô hình Class Diagram



Hình 3.2 Mô hình Class Diagram

### 3.3. Biểu đồ Use Case

#### 3.3.1. Biểu đồ Use Case tổng quát



Hình 3.3 Biểu đồ Use Case tổng quát

#### 3.3.2. Biểu đồ Use Case phân rã và đặc tả chi tiết

##### 3.3.2.1. Use Case: Đăng nhập

Đặc tả Use Case: Đăng nhập

Bảng 3.2 Đặc tả Use Case: Đăng nhập

Tên Use Case: Đăng nhập	Mức quan trọng: Cao
Tác nhân chính: Khách hàng, Nhà hàng, Nhà vận chuyển	Loại Use Case: Chi tiết, cần thiết
Người liên quan và công việc quan tâm: Người dùng muốn đăng nhập vào hệ thống để truy cập vào trang đăng ký nếu tài khoản chưa đăng ký.	
Mô tả tóm tắt: Use Case này mô tả xử lý đăng nhập vào hệ thống của một người dùng.	

Ràng buộc: Người dùng chọn chức năng đăng nhập từ trang chủ của hệ thống Loại: Bên ngoài (External).
<p>Các mối quan hệ:</p> <ol style="list-style-type: none"> <li>1. Kết hợp: Không có.</li> <li>2. Bao hàm: Không có.</li> <li>3. Mở rộng: Không có.</li> <li>4. Tổng quát hóa: Không có.</li> </ol>
<p>Các dòng sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Hệ thống hiển thị màn hình đăng nhập.</li> <li>2. Người dùng chọn chức năng: Nếu người sử dụng nhập thông tin đăng nhập thì thực hiện 2.1: Xử lý đăng nhập</li> <li>3. Hệ thống hiển thị màn hình thông báo kết quả cho giao tác liên quan đến người dùng.</li> </ol>
<p>Các dòng sự kiện con</p> <ol style="list-style-type: none"> <li>2.1 Xử lý đăng nhập <ol style="list-style-type: none"> <li>2.1.1 Kiểm tra thông tin tài khoản</li> </ol> </li> </ol> <p>Hiển thị thông báo đăng nhập thành công</p>

### 3.3.2.2. Use Case :Đăng ký

Đặc tả Use Case: Đăng ký

*Bảng 3.3 Đặc tả Use Case: Đăng ký*

Tên Use Case: Đăng ký	Mức quan trọng: Cao
Tác nhân chính: Khách hàng, Nhà hàng, Nhà vận chuyển	Loại Use Case: Chi tiết, cần thiết
Người liên quan và công việc quan tâm: Người dùng muốn đăng ký để truy cập vào trang chủ.	
Mô tả tóm tắt: Use Case này mô tả xử lý những tài khoản chưa được đăng ký.	
Ràng buộc: Sau khi đăng nhập thành công, người dùng sẽ vào trang đăng ký nếu tài khoản chưa có thông tin cá nhân của Khách hàng.	

Loại: Bên ngoài (External).
<p>Các mối quan hệ:</p> <ol style="list-style-type: none"> <li>1. Kết hợp: Không có.</li> <li>2. Bao hàm: Không có.</li> <li>3. Mở rộng: Không có.</li> <li>4. Tổng quát hóa: Không có.</li> </ol>
<p>Các dòng sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Hệ thống hiển thị màn hình đăng ký.</li> <li>2. Người dùng chọn chức năng:</li> <li>3. Nếu người sử dụng nhập thông tin đăng ký thì thực hiện 2.1: Xử lý đăng ký</li> <li>3.1: Hệ thống hiển thị màn hình thông báo kết quả cho giao tác liên quan đến người dùng.</li> </ol>
<p>Các dòng sự kiện con</p> <ol style="list-style-type: none"> <li>2.1 Xử lý đăng ký</li> <li>2.2 Kiểm tra thông tin tài khoản</li> </ol> <p>Hiển thị thông báo đăng ký thành công</p>

### 3.3.2.3. Use Case: Đăng xuất

Đặc tả Use Case: Đăng xuất

*Bảng 3.4 Đặc tả Use Case: Đăng xuất*

Tên Use Case: Đăng xuất	Mức quan trọng: Cao
Tác nhân chính: Khách hàng, Nhà hàng, Nhà vận chuyển	Loại Use Case: Chi tiết, cần thiết
Người liên quan và công việc quan tâm: Người dùng muốn đăng xuất khỏi hệ thống.	
Mô tả tóm tắt: Use Case này mô tả xử lý đăng xuất khỏi hệ thống của một người dùng.	
<p>Ràng buộc: Người dùng chọn chức năng đăng xuất từ hệ thống.</p> <p>Loại: Bên ngoài (External)</p>	

Các mối quan hệ:

1. Kết hợp: Không có
2. Bao hàm: Không có
3. Mở rộng: Không có
4. Tổng quát hóa: Không có

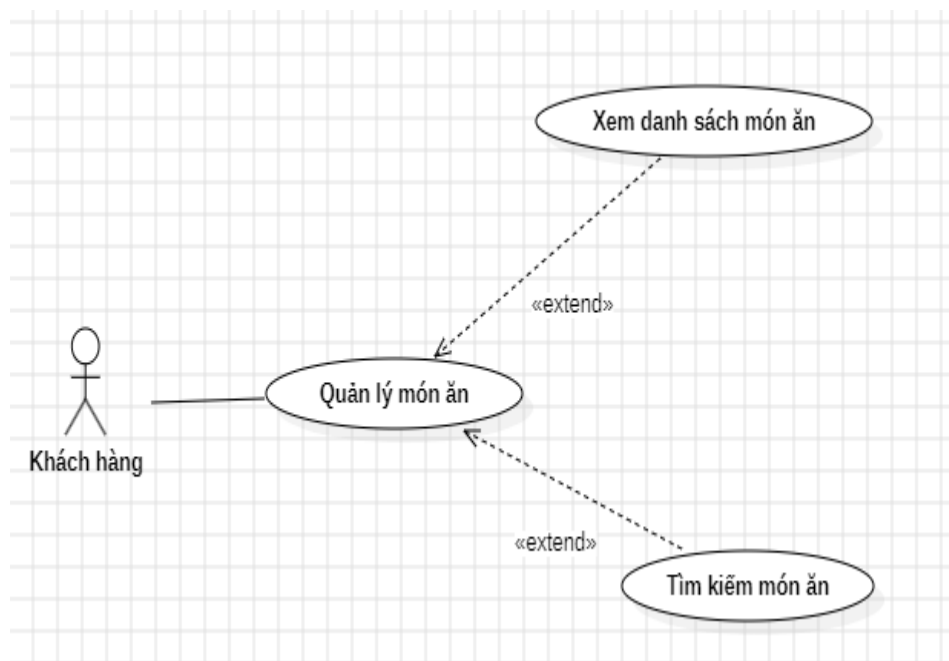
Các dòng sự kiện chính

1. Hệ thống hiển thị chức năng đăng xuất.
2. Người dùng chọn chức năng:  
Nếu người sử dụng chức năng đăng xuất thì thực hiện 2.1: Xử lý đăng xuất.
3. Hệ thống hiển thị màn hình thông báo kết quả cho giao tác liên quan đến người dùng.

Các dòng sự kiện con

- 2.1 Xử lý đăng xuất
  - 2.1.1 Kiểm tra thông tin tài khoản.
  - 2.1.2 Hiển thị thông báo đăng xuất thành công và quay về màn hình đăng nhập

### 3.3.2.4. Use Case: Quản lý món ăn -Khách hàng



Hình 3.4 Biểu đồ Use Case phân rã Quản lý món ăn- Khách hàng

Đặc tả Use Case: Quản lý món ăn

Bảng 3.5 Đặc tả Use Case: Quản lý món ăn

Tên Use Case: Quản lý món ăn	Mức quan trọng: Cao
Tác nhân chính: Khách hàng	Loại Use Case: Chi tiết, cần thiết
Người liên quan và công việc quan tâm: Người dùng muốn xem danh sách và tìm kiếm món ăn trên hệ thống.	
Mô tả tóm tắt: Use Case này mô tả xử lý quản lý món ăn trên hệ thống của một khách hàng.	
Ràng buộc: : Khách hàng chọn chức năng “Xem danh sách món ăn”, và chức năng “Tìm kiếm món ăn”	
Loại: Bên ngoài (External)	
Các mối quan hệ: <ol style="list-style-type: none"> <li>1. Kết hợp: Không có</li> <li>2. Bao hàm: Không có</li> <li>3. Mở rộng: Không có</li> </ol> Tổng quát hóa: Không có	
Các dòng sự kiện chính <ol style="list-style-type: none"> <li>1. Hệ thống hiển thị màn hình</li> <li>2. Khách hàng chọn chức năng:               <ul style="list-style-type: none"> <li>Nếu khách hàng sử dụng nhập thông tin tìm kiếm thì thực hiện</li> <li>2.1: Xử lý tìm kiếm.</li> </ul> </li> </ol> Nếu khách hàng chọn chức năng xem danh sách món ăn thì thực hiện <ul style="list-style-type: none"> <li>2.2: Xử lý xem thông tin</li> </ul>	
Các dòng sự kiện con <ol style="list-style-type: none"> <li>2.1 Xử lý tìm kiếm đối với Khách hàng               <ol style="list-style-type: none"> <li>2.1.1 Nhập thông tin tìm kiếm và chọn tìm kiếm.</li> <li>2.1.2 Hệ thống kiểm tra dữ liệu hợp lệ.</li> <li>2.1.3 Hiển thị thông tin tìm kiếm.</li> </ol> </li> <li>2.2 Xử lý xem thông tin với Khách hàng.</li> </ol> Hệ thống hiển thị danh sách món ăn cho Khách hàng.	

### 3.3.2.5. Use Case: Quản lý tài khoản- Khách hàng

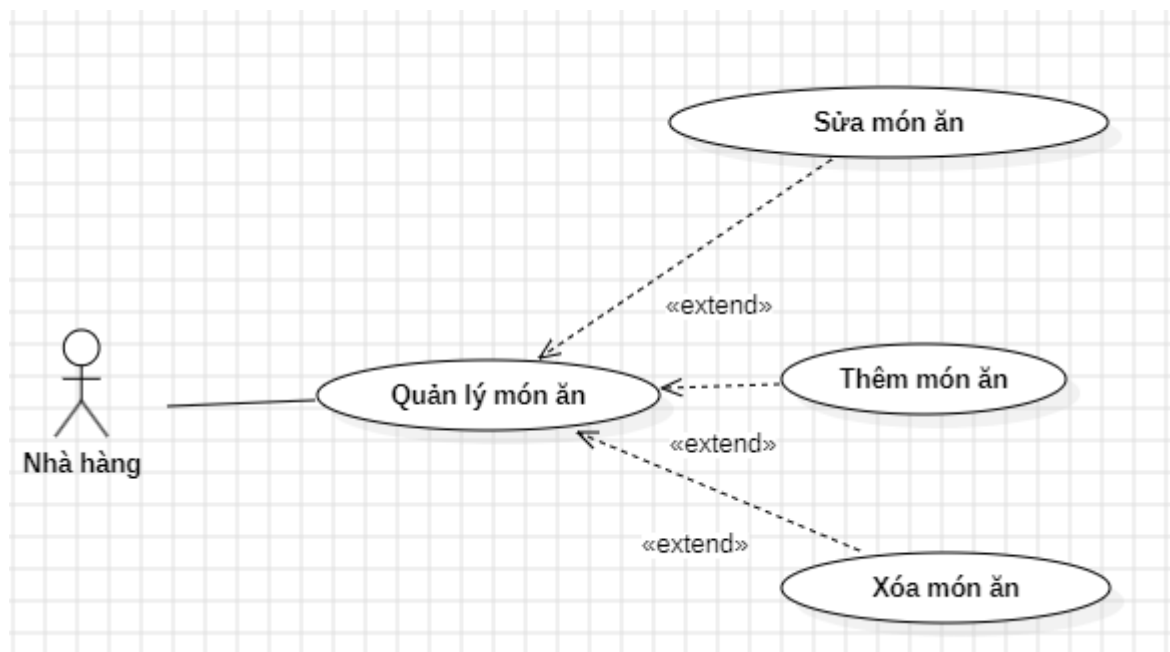
Đặc tả Use Case: Quản lý tài khoản – Khách hàng

*Bảng 3.6 Đặc tả Use Case: Quản lý tài khoản*

Tên Use Case: Quản lý tài khoản -Khách hàng	Mức quan trọng: Cao
Tác nhân chính: Quản lý tài khoản - Khách hàng	Loại Use Case: Chi tiết, cần thiết
Người liên quan và công việc quan tâm: Khách hàng muốn quản lý thông tin đơn hàng	
Mô tả tóm tắt: Use Case này mô tả xử lý quản lý đơn hàng trên hệ thống của một khách hàng	
Ràng buộc: Khách hàng chọn chức năng “Đơn hàng”	
Loại: Bên ngoài (External)	
<p>Các mối quan hệ:</p> <ol style="list-style-type: none"> <li>1. Kết hợp: Không có</li> <li>2. Bao hàm: Không có</li> <li>3. Mở rộng: Không có</li> <li>4. Tổng quát hóa: Không có</li> </ol>	
<p>Các dòng sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Hệ thống hiển thị màn hình quản lý đơn hàng</li> <li>2. Khách hàng chọn chức năng:  Nếu người dùng sử dụng chọn “Đơn hàng” để quản lý đơn hàng  2.1: Xử lý tác vụ xem thông tin đơn hàng, 2.2: Xử lý tác vụ thanh toán đơn hàng,</li> </ol>	
<p>Các dòng sự kiện con</p> <ol style="list-style-type: none"> <li>2.1 Xử lý xem thông tin <ol style="list-style-type: none"> <li>2.1.1 Chọn “Đơn hàng”</li> <li>2.1.2 Hệ thống hiển thị thông tin đơn hàng của Khách hàng.</li> </ol> </li> <li>2.2 Xử lý thanh toán đơn hàng <ol style="list-style-type: none"> <li>2.2.1 Hệ thống hiển thị màn hình Thanh toán</li> <li>2.2.2 Nếu Khách hàng chưa xác thực thẻ, sẽ chuyển đến trang xác thực thẻ thanh toán online.</li> </ol> </li> </ol>	

2.2.3 Hệ thống kiểm tra, xác thực thành công thông tin thanh toán và đơn hàng. lưu thông tin và trả về thông báo.

### 3.3.2.6. Use Case: Quản lý món ăn- Nhà hàng



Hình 3.5 Biểu đồ Use Case phân rã Quản lý món ăn- Nhà hàng

Đặc tả Use Case: Quản lý món ăn- Nhà hàng

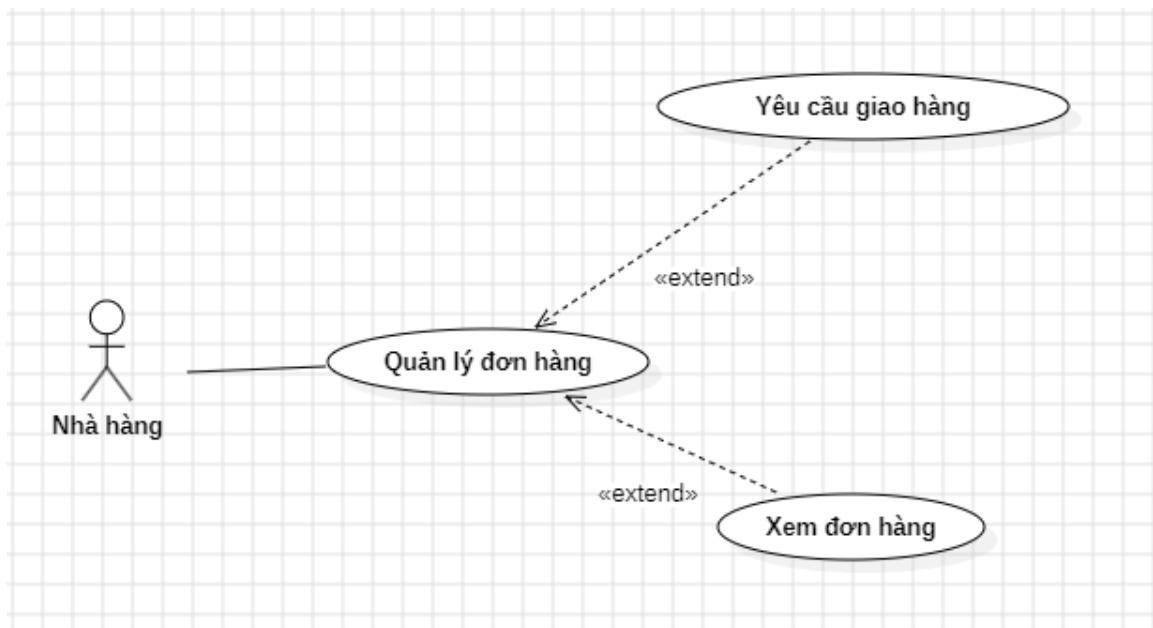
Bảng 3.6 Đặc tả Use Case: Quản lý món ăn- Nhà hàng

Tên Use Case: Quản lý món ăn - Nhà hàng	Mức quan trọng: Cao
Tác nhân chính: Quản lý món ăn - Nhà hàng	Loại Use Case: Chi tiết, cần thiết
Người liên quan và công việc quan tâm: Nhà hàng muốn thực hiện một số thao tác liên quan đến quản lý món ăn trên hệ thống.	
Mô tả tóm tắt: Use Case này mô tả xử lý quản lý món ăn trên hệ thống của một nhà hàng	
Ràng buộc: Nhà hàng chọn chức năng “Menu” từ trang quản lý của hệ thống	
Loại: Bên ngoài (External)	
Các mối quan hệ: <ol style="list-style-type: none"> <li>1. Kết hợp: Không có</li> <li>2. Bao hàm: Không có</li> <li>3. Mở rộng: Không có</li> </ol>	



4. Tổng quát hóa: Không có
<p>Các dòng sự kiện chính</p> <ol style="list-style-type: none"> <li>Hệ thống hiển thị màn hình menu món ăn</li> <li>Người dùng chọn chức năng: Nếu người sử dụng quản lý món ăn thì thực hiện 2.1: Xử lý sửa món ăn, 2.2: Xử lý thêm món ăn, 2.3: Xử lý xóa món ăn</li> <li>Hệ thống hiển thị màn hình thông báo kết quả cho giao tác liên quan đến nhà hàng.</li> </ol>
<p>Các dòng sự kiện con</p> <ol style="list-style-type: none"> <li>Xử lý sửa món ăn. <ol style="list-style-type: none"> <li>Hệ thống hiển thị dữ liệu và yêu cầu Nhà hàng nhập thông tin.</li> <li>Nhà hàng xác nhận và hệ thống lưu thông tin.</li> <li>Trả về kết quả thông báo.</li> </ol> </li> <li>Xử lý thêm món ăn. <ol style="list-style-type: none"> <li>Hệ thống hiển thị trang thêm món ăn.</li> <li>Hệ thống yêu cầu Nhà hàng nhập thông tin, hình ảnh món ăn.</li> <li>Nhà hàng chọn “Thêm”.</li> <li>Hệ thống lưu dữ liệu lên máy chủ và trả về thông báo.</li> </ol> </li> <li>Xử lý xóa món ăn. <ol style="list-style-type: none"> <li>Hệ thống hiển thị dữ liệu và yêu cầu Nhà hàng chọn 1 món ăn để xóa.</li> <li>Nhà hàng xác nhận và hệ thống xóa dữ liệu.</li> <li>Hệ thống cập nhật thông tin dữ liệu trên máy chủ và trả về thông báo cho Nhà hàng.</li> </ol> </li> </ol>

### 3.3.2.7. Use Case: Quản lý đơn hàng- Nhà hàng



Hình 3.6 Biểu đồ Use Case phân rã Quản lý đơn hàng- Nhà hàng

Đặc tả Use Case: Quản lý đơn hàng- Nhà hàng

Bảng 3.7 Đặc tả Use Case: Quản lý đơn hàng- Nhà hàng

Tên Use Case: Quản lý đơn hàng- Nhà hàng	Mức quan trọng: Cao
Tác nhân chính: Quản lý đơn hàng- Nhà hàng	Loại Use Case: Chi tiết, cần thiết
Người liên quan và công việc quan tâm: Nhà hàng muốn thực hiện một số thao tác liên quan đến đơn hàng của Nhà hàng.	
Mô tả tóm tắt: Use Case này mô tả xử lý các đơn hàng trên hệ thống của một Nhà hàng.	
Ràng buộc: Nhà hàng chọn chức năng từ trang quản lý của hệ thống	
Loại: Bên ngoài (External)	
Các mối quan hệ: <ol style="list-style-type: none"> <li>1. Kết hợp: Không có</li> <li>2. Bao hàm: Không có</li> <li>3. Mở rộng: Không có</li> <li>4. Tổng quát hóa: Không có</li> </ol>	

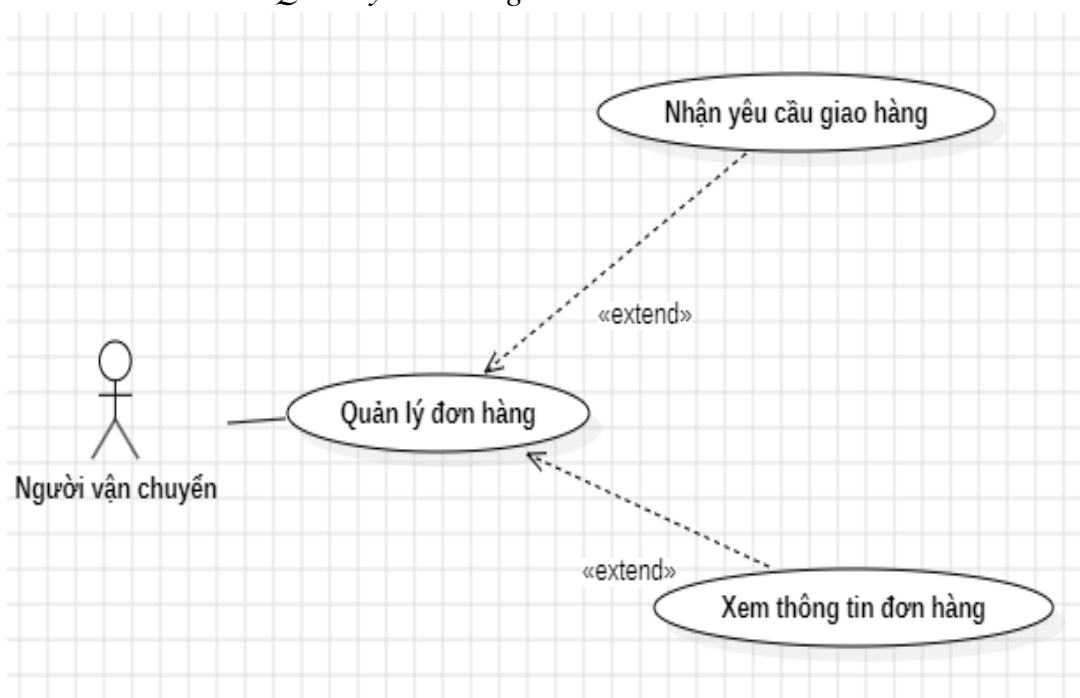
#### Các dòng sự kiện chính

1. Hệ thống hiển thị danh sách đơn
2. Người dùng chọn chức năng:  
Nếu Nhà hàng sử dụng chọn “Yêu cầu giao hàng” thì thực hiện 2.1: Xử lý yêu cầu giao hàng  
Nếu Nhà hàng sử dụng chọn “Xem thông tin đơn hàng” thì thực hiện 2.2: Xử lý xem thông tin đơn hàng.
3. Hệ thống hiển thị màn hình thông báo kết quả cho giao tác liên quan đến người dùng.

#### Các dòng sự kiện con

- 4.3 Xử lý yêu cầu giao hàng
  - 4.3.2 Nhà hàng chọn giao hàng một món ăn được đặt từ Khách hàng.
  - 4.3.3 Hệ thống gửi thông tin đơn hàng của Nhà hàng và Khách hơn tới người giao hàng( Nhà vận chuyển)
- 4.4 Xử lý xem thông tin đơn hàng  
Hệ thống hiển thị danh sách món ăn được đặt từ Khách hàng

#### 3.3.2.8. Use Case: Quản lý đơn hàng



Hình 3.7 Biểu đồ Use Case phân rã Quản lý đơn hàng- Người vận chuyển

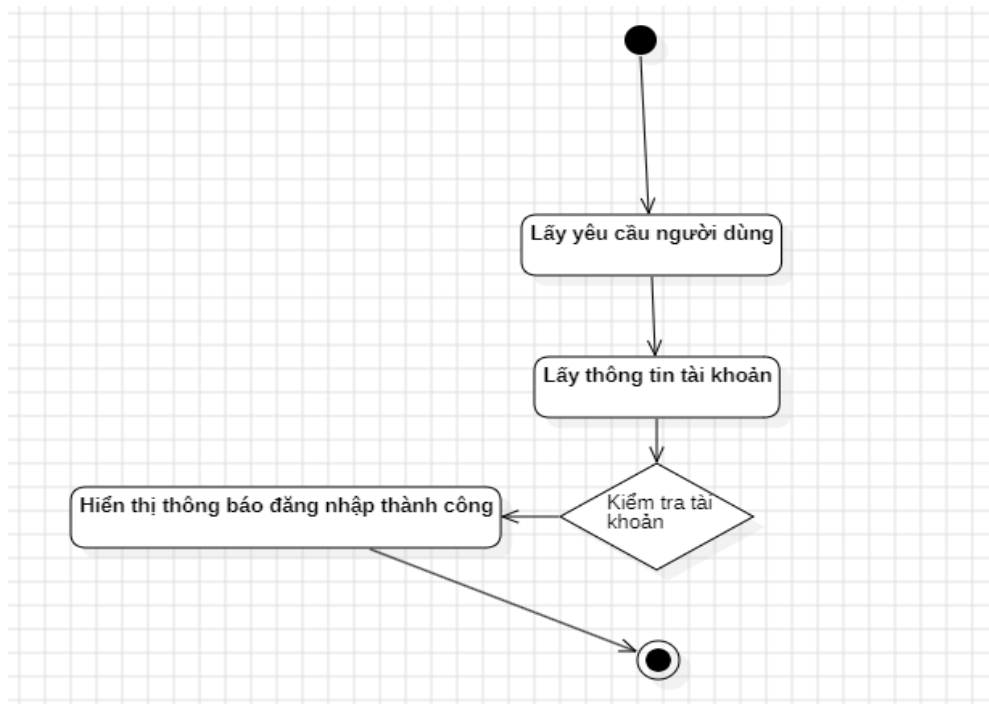
## Đặc tả Use Case: Quản lý đơn hàng- Người vận chuyển

*Bảng 3.8 Đặc tả Use Case: Quản lý đơn hàng- Người vận chuyển*

Tên Use Case: Quản lý đơn hàng- Người vận chuyển	Mức quan trọng: Cao
Tác nhân chính: Quản lý đơn hàng- Người vận chuyển	Loại Use Case: Chi tiết, cần thiết
Người liên quan và công việc quan tâm: Người vận chuyển muốn thực hiện một số thao tác liên quan đến quản lý đề tài trên hệ thống.	
Mô tả tóm tắt: Use Case này mô tả xử lý quản lý đơn hàng trên hệ thống của một người giao hàng.	
Ràng buộc: Người vận chuyển chọn chức năng “Đơn hàng” Loại: Bên ngoài (External)	
Các mối quan hệ: 1. Kết hợp: Không có 2. Bao hàm: Không có 3. Mở rộng: Không có 4. Tổng quát hóa: Không có	
Các dòng sự kiện chính 1.Hệ thống hiển thị đơn hàng cần vận chuyển 2.Khách hàng chọn chức năng: -Nếu khách hàng sử dụng chọn “Nhận đơn hàng” để giao hàng 2.1: Xử lý tác vụ giao hàng, 2.2: Xem thông tin	
Các dòng sự kiện con 2.1 Xử lý tác vụ giao hàng. 2.1.1 Hiển thị thông tin đơn hàng và bản đồ. 2.1.2 Hệ thống hiển thị vị trí đơn hàng và Nhà hàng. 2.1.3 Yêu cầu người giao hàng cập nhật tình trạng “Nhận đơn” và “Giao thành công” để Nhà hàng và Khách hàng dễ nắm bắt thông tin đơn hàng. 2.2 Xử lý tác vụ xem thông tin đối với người vận chuyển. Hệ thống hiển thị thông tin lịch sử đơn hàng.	

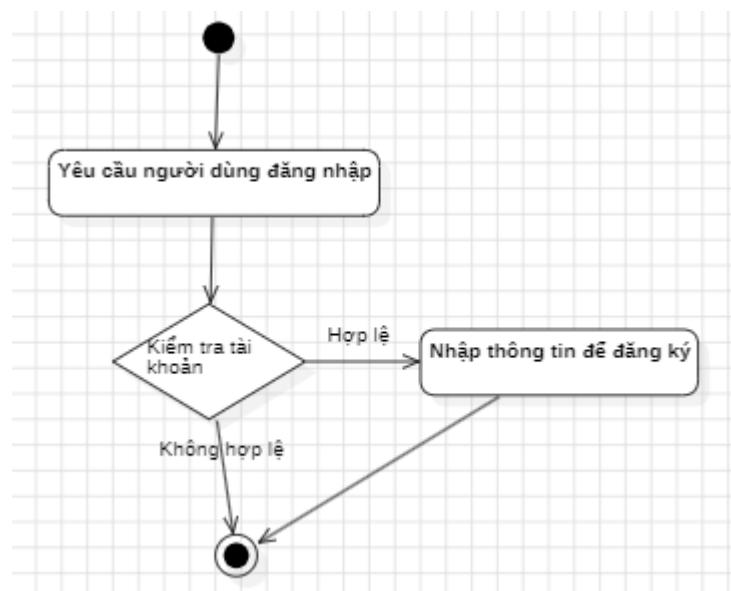
### 3.4. Biểu đồ hoạt động (Activity Diagram)

#### 3.4.1. Activity Diagram: Đăng nhập



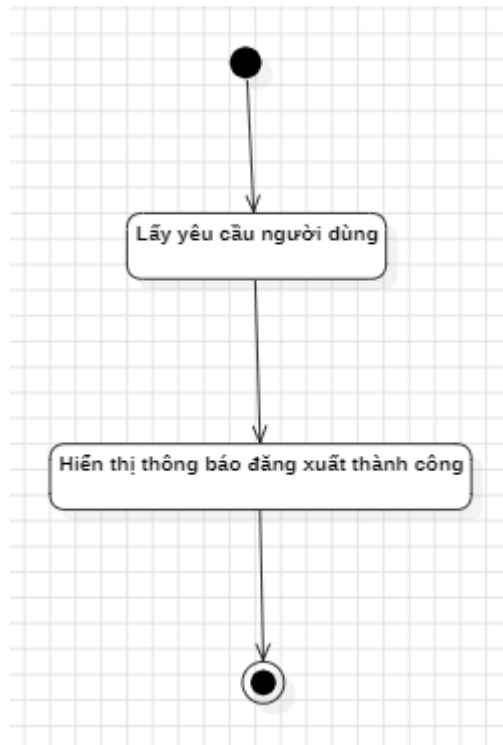
Hình 3.8 Activity Diagram Đăng nhập

### 3.4.2. Activity Diagram: Đăng ký



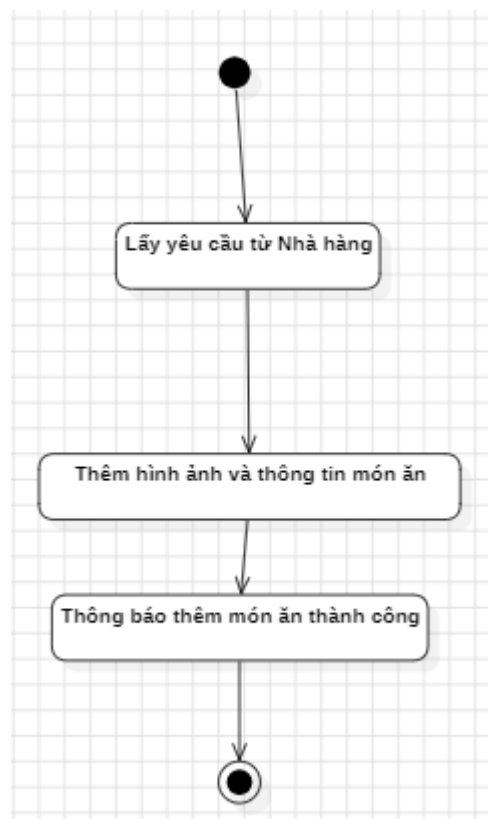
Hình 3.9 Activity Diagram Đăng ký

### 3.4.2. Activity Diagram: Đăng xuất



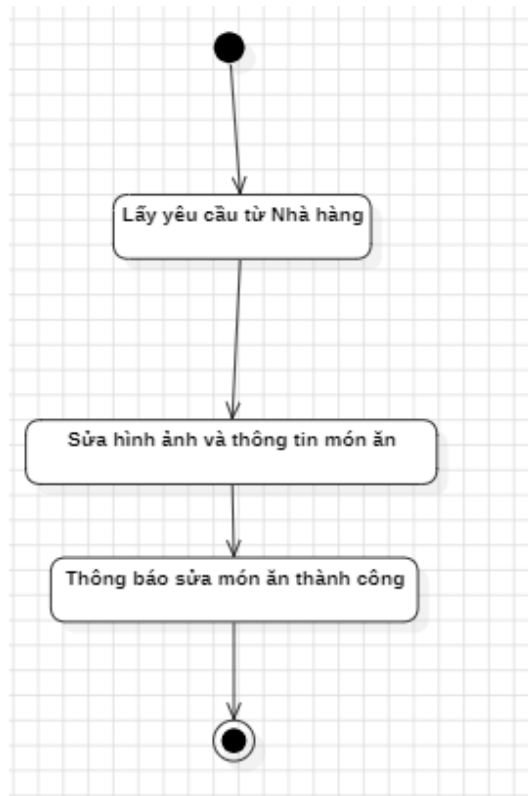
Hình 3.10 Activity Diagram Đăng xuất

#### 3.4.5. Activity Diagram: Quản lý món ăn -Thêm



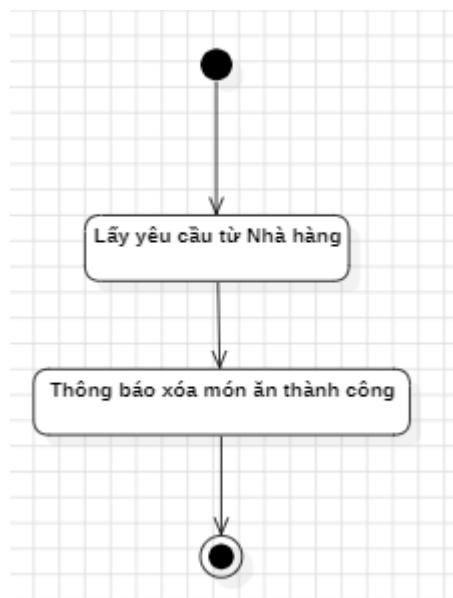
Hình 3.11 Activity Diagram Quản lý món ăn- Thêm

#### 3.4.6. Activity Diagram: Quản lý món ăn - Sửa



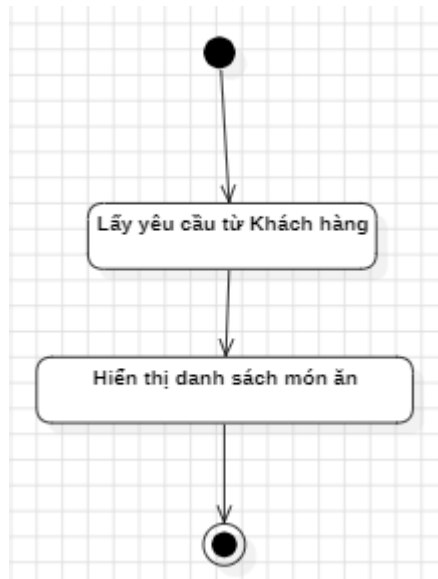
Hình 3.12 Activity Diagram Quản lý món ăn- Sửa

#### 3.4.7. Activity Diagram: Quản lý món ăn -Xóa



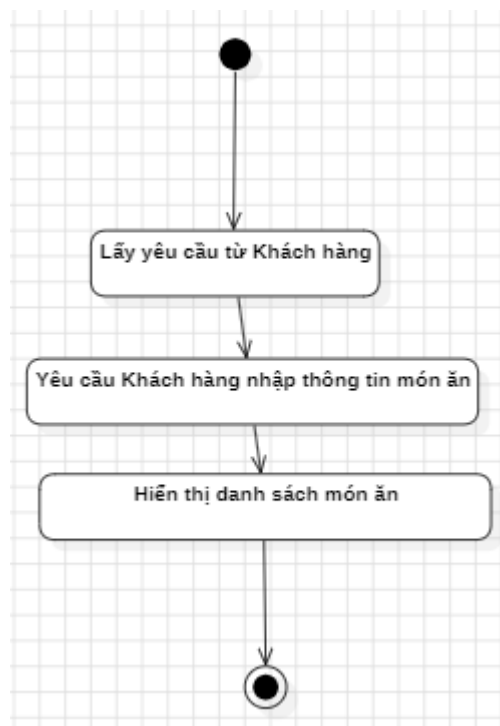
Hình 3.13 Activity Diagram Quản lý món ăn- Xóa

#### 3.4.8. Activity Diagram: Quản lý món ăn -Xem thông tin



Hình 3.14 Activity Diagram Quản lý món ăn- Xem thông tin

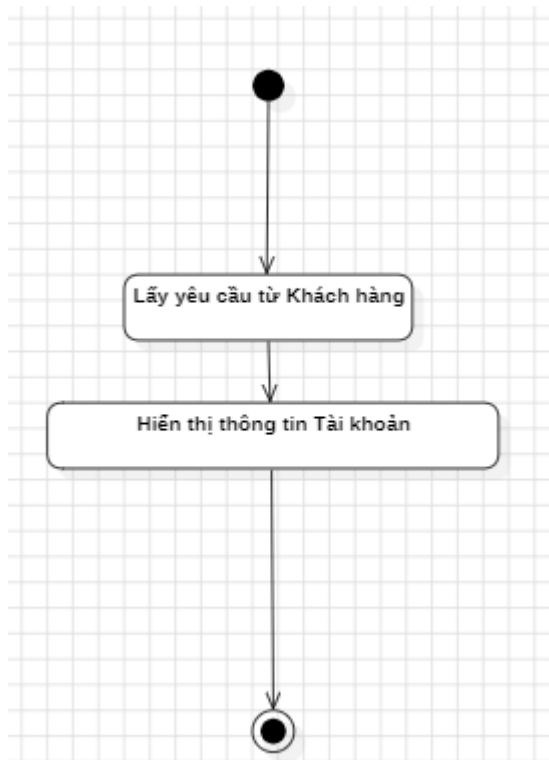
#### 3.4.9. Activity Diagram: Quản lý món ăn – Tìm kiếm món ăn



Hình 3.15 Activity Diagram Quản lý món ăn- Tìm kiếm món ăn

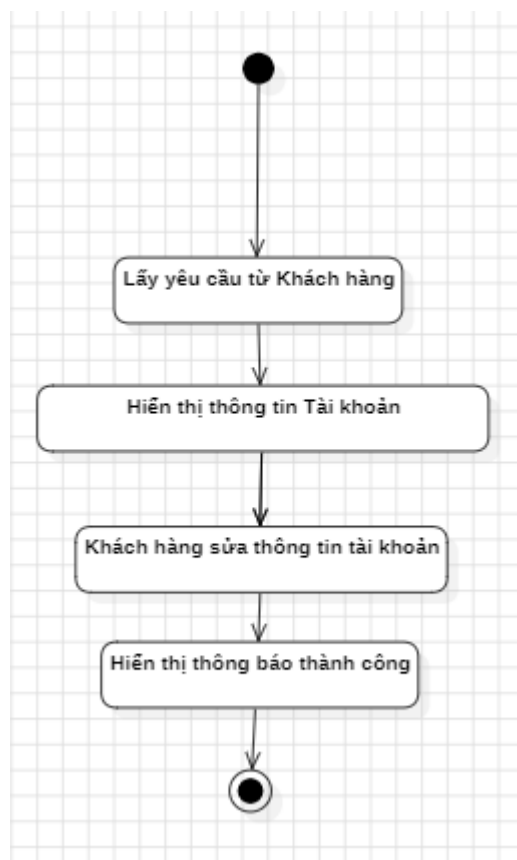
#### 3.4.10. Activity Diagram: Quản lý tài khoản – Xem thông tin tài khoản





Hình 3.16 Activity Diagram Quản lý tài khoản- Xem thông tin

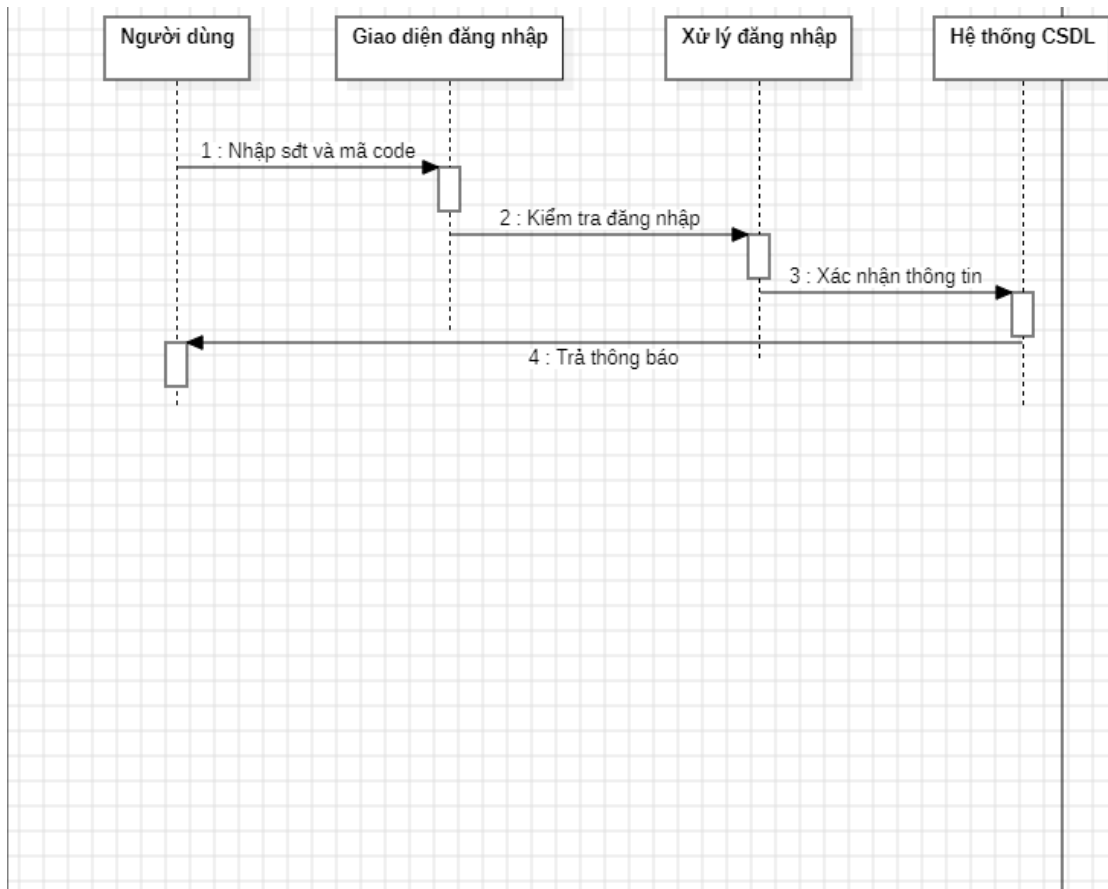
#### 3.4.11. Activity Diagram: Quản lý tài khoản – Sửa thông tin



Hình 3.17 Activity Diagram Quản lý món ăn- Sửa thông tin

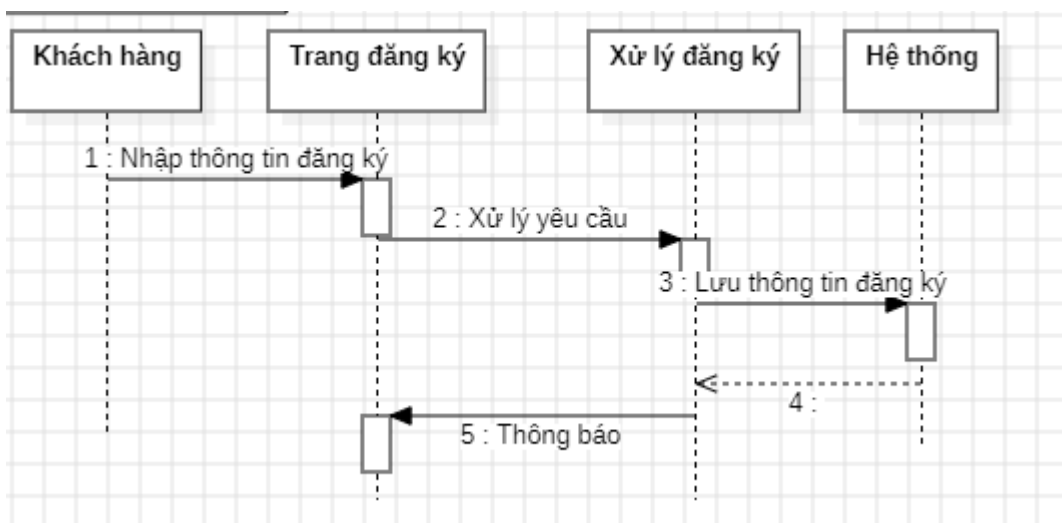
### 3.5. Biểu đồ tuần tự (Sequence Diagram)

#### 3.5.1. Sequence Diagram: Đăng nhập



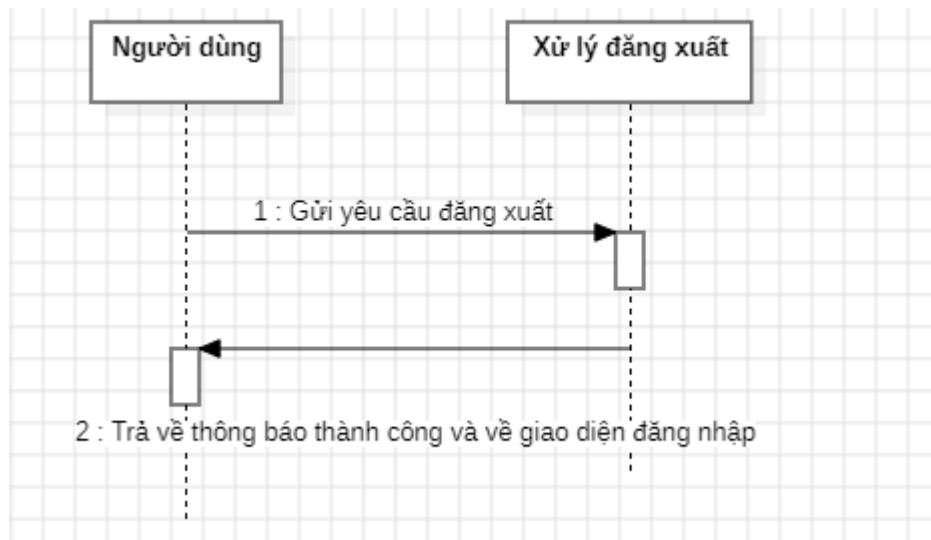
Hình 3.18 Biểu đồ tuần tự Đăng nhập

#### 3.5.2. Sequence Diagram: Đăng ký



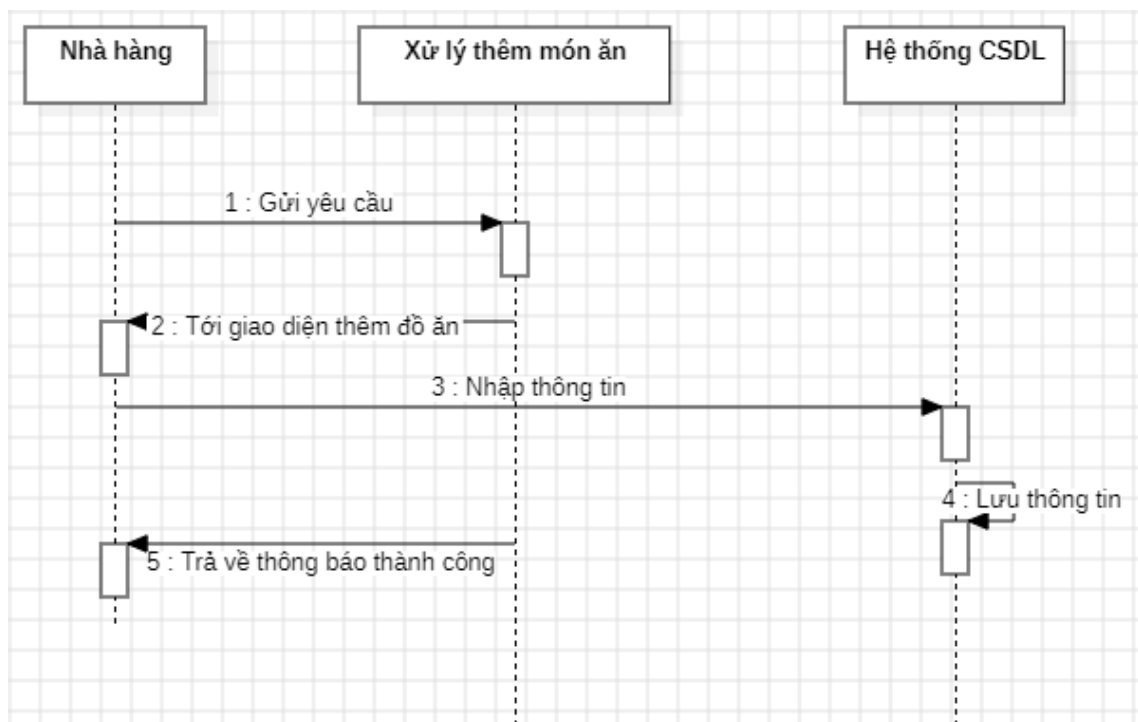
Hình 3.19 Biểu đồ tuần tự Đăng ký

#### 3.5.3. Sequence Diagram: Đăng xuất



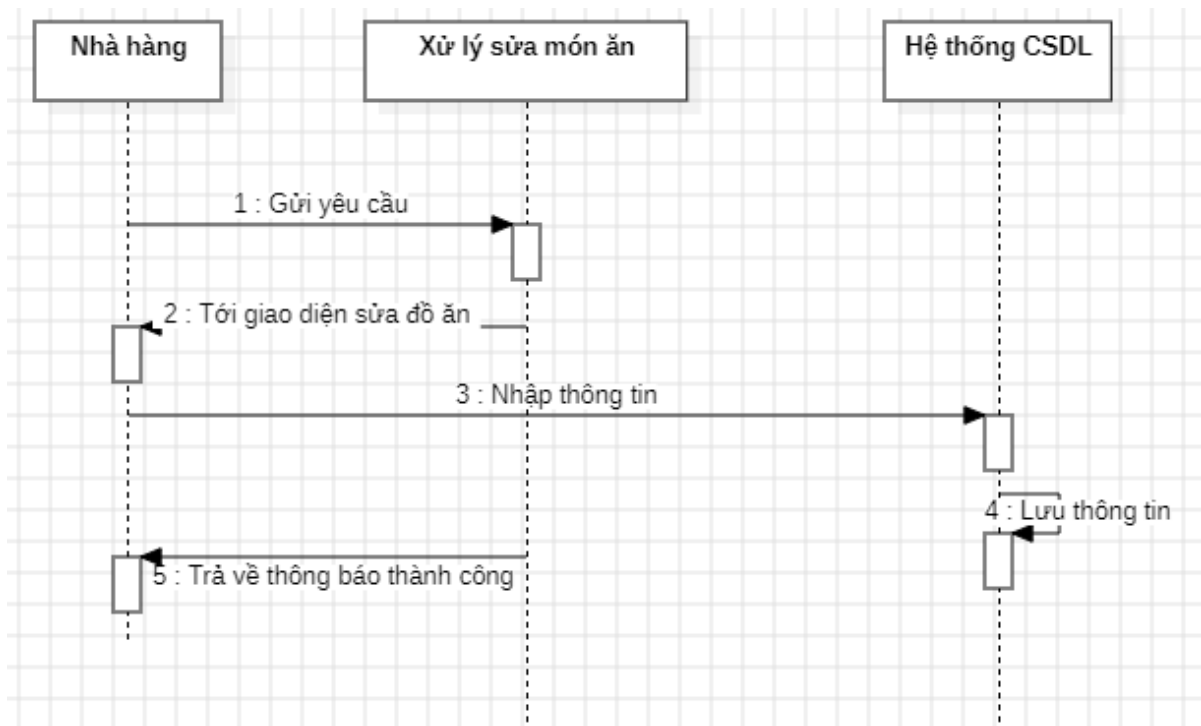
Hình 3.20 Biểu đồ tuần tự Đăng xuất

#### 3.5.4. Sequence Diagram: Quản lý món ăn -Thêm món ăn



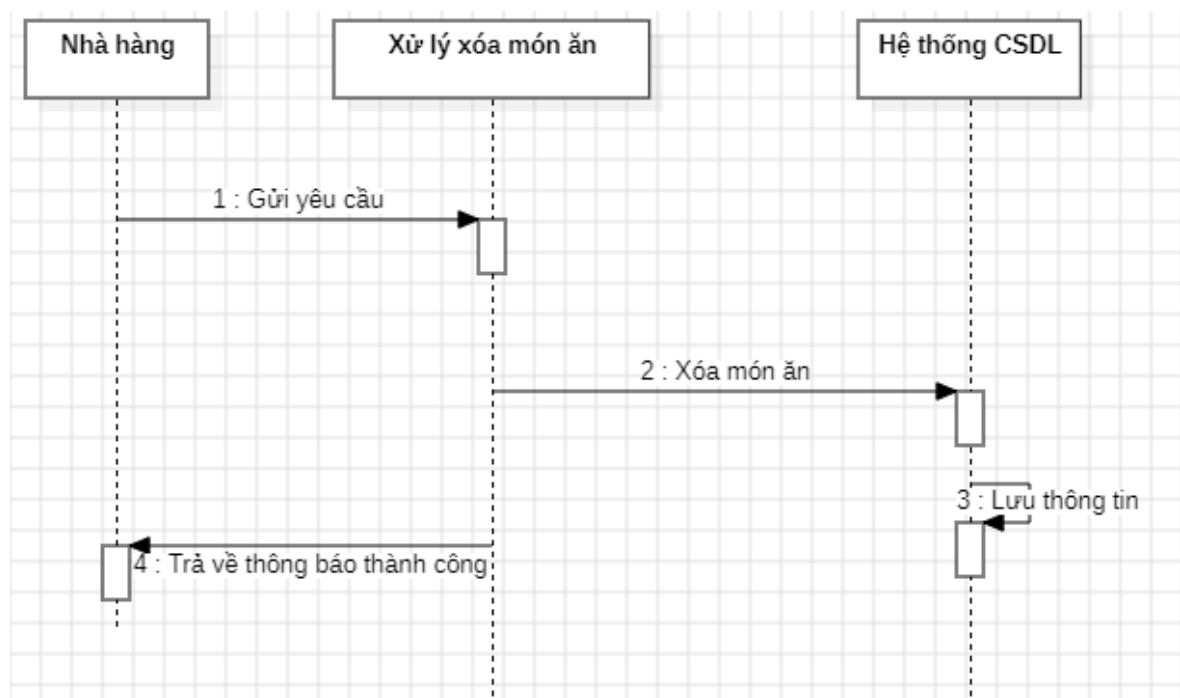
Hình 3.21 Biểu đồ tuần tự Quản lý món ăn- Thêm món ăn

#### 3.5.5. Sequence Diagram: Quản lý món ăn -Sửa món ăn



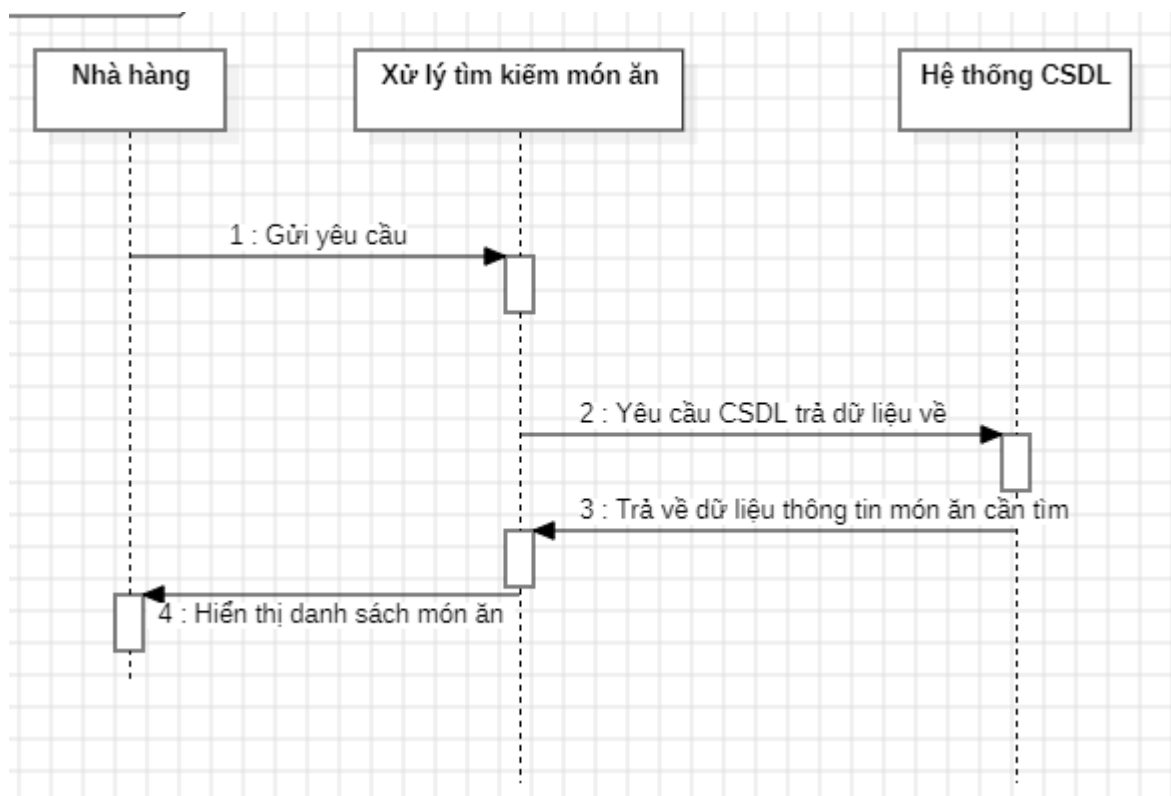
Hình 3.22 Biểu đồ tuần tự Quản lý món ăn- Sửa món ăn

### 3.5.6. Sequence Diagram: Quản lý món ăn -Xóa món ăn



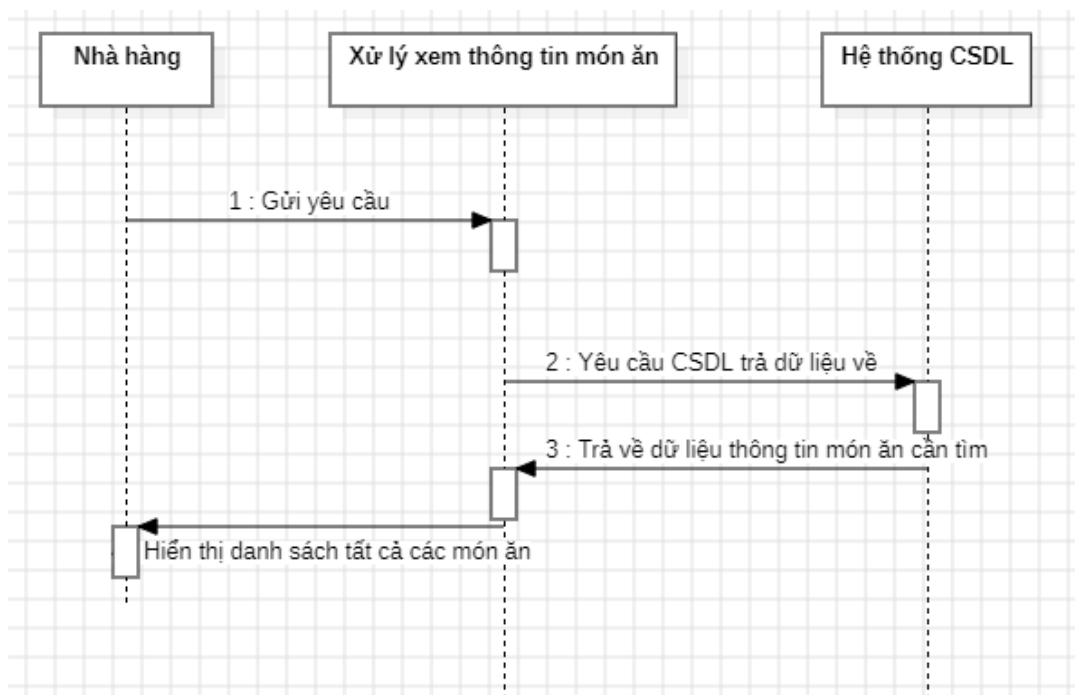
Hình 3.23 Biểu đồ tuần tự Quản lý món ăn- Xóa món ăn

### 3.5.7. Sequence Diagram: Quản lý món ăn -Tìm kiếm món ăn



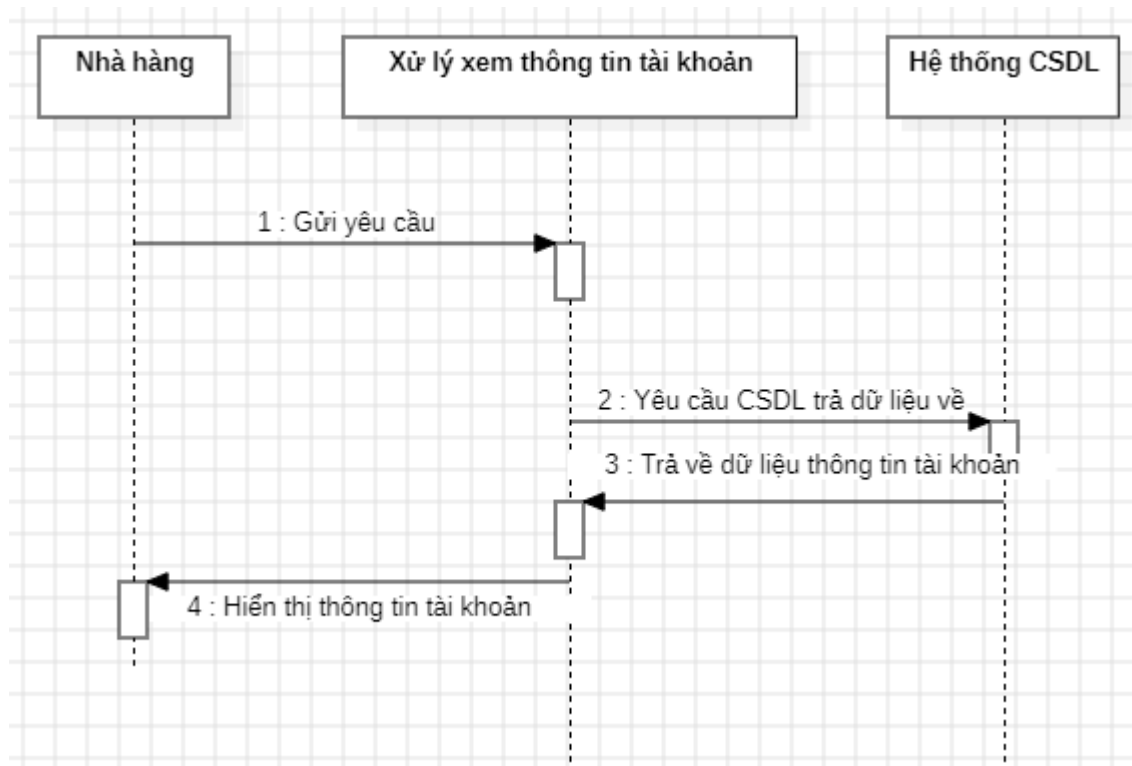
Hình 3.24 Biểu đồ tuần tự Quản lý món ăn- Tìm kiếm món ăn

### 3.5.8. Sequence Diagram: Quản lý món ăn -Xem thông tin món ăn



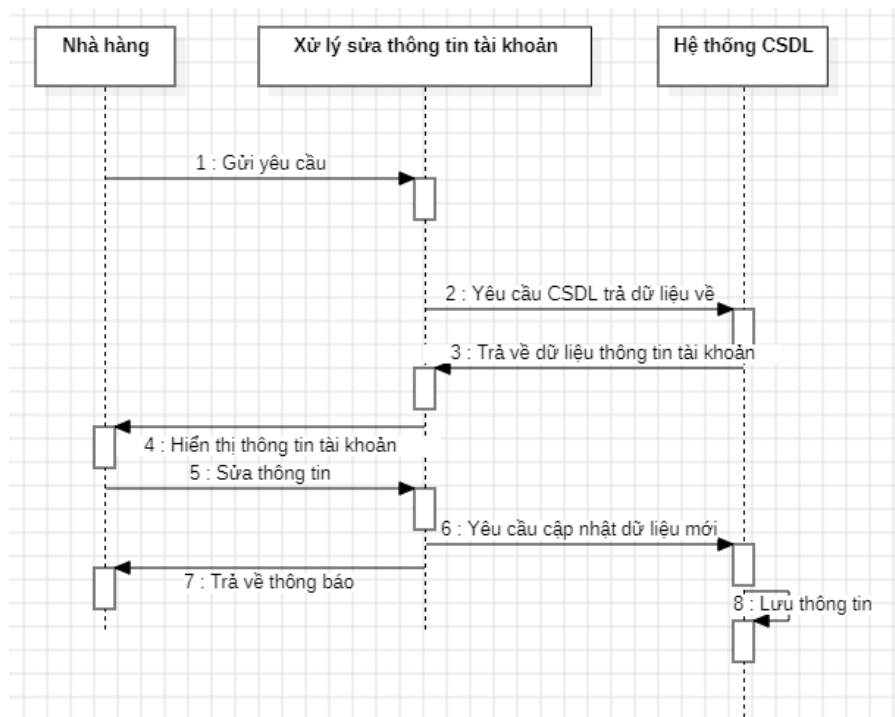
Hình 3.25 Biểu đồ tuần tự Quản lý món ăn- Xem món ăn

### 3.5.9. Sequence Diagram: Quản lý tài khoản- Xem thông tin



Hình 3.26 Biểu đồ tuần tự Quản lý món ăn- Xem thông tin tài khoản

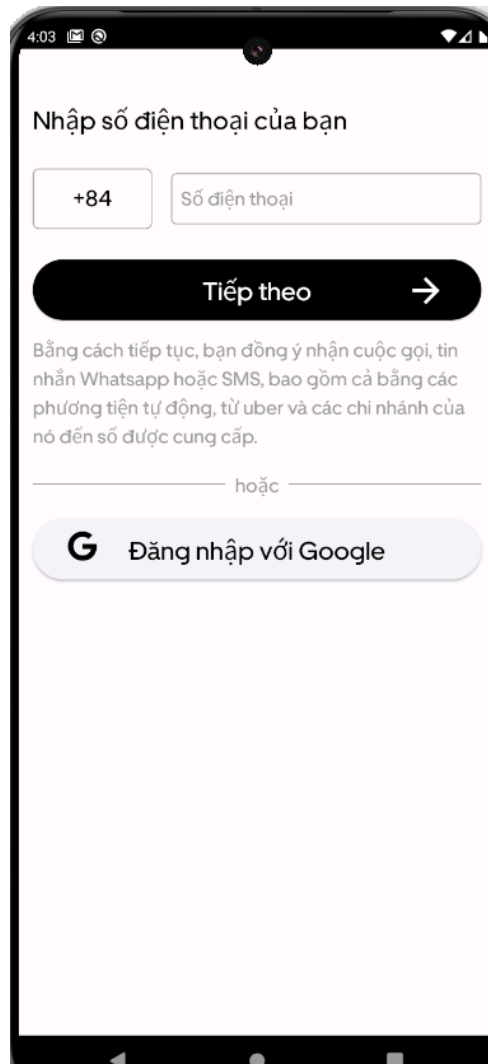
### 3.5.10. Sequence Diagram: Quản lý tài khoản- Sửa thông tin



Hình 3.27 Biểu đồ tuần tự Quản lý tài khoản- Sửa thông tin

## CHƯƠNG 4 THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

### 4.1. Xây dựng đăng nhập cho người dùng (khách hàng, nhà hàng, nhà vận chuyển)



Hình 4.1 Giao diện đăng nhập- Khách hàng

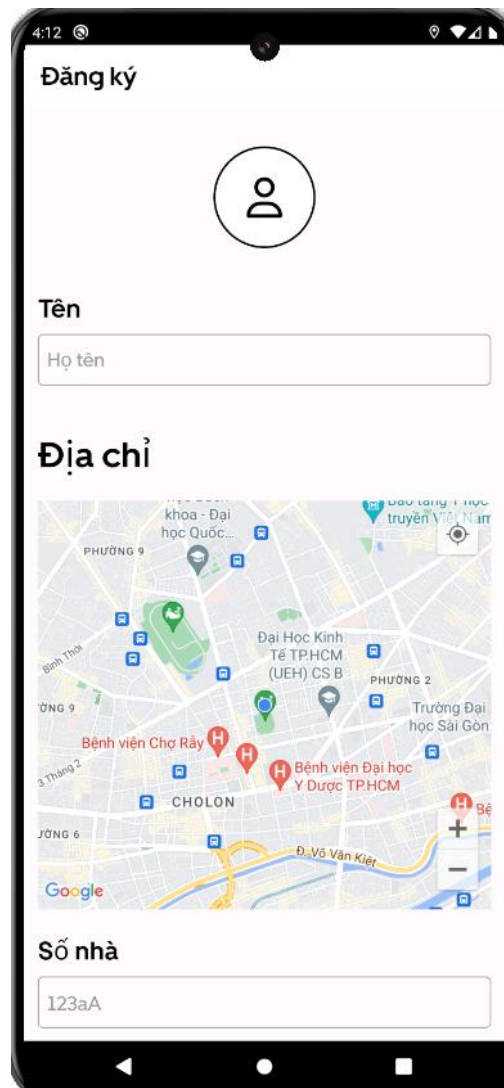
Mô tả giao diện đăng nhập: Giao diện yêu cầu người dùng nhập thông tin để vào giao diện nhập mã OTP.



*Hình 4.2 Giao diện nhập mã OTP- Khách hàng*

Mô tả giao diện danh sách đề tài: Giao diện yêu cầu người dùng nhập mã OTP để truy cập vào trang chủ.

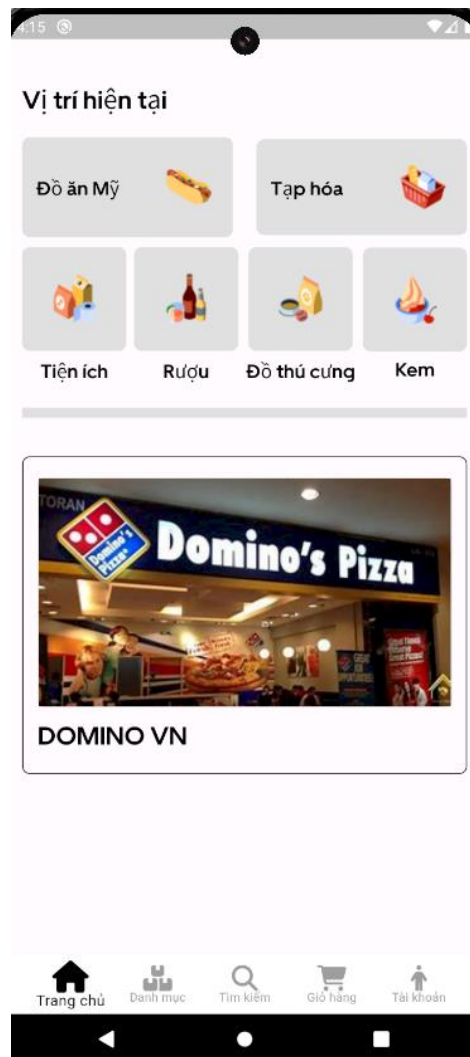




*Hình 4.3 Giao diện Đăng ký (Khách hàng)*

Mô tả giao diện đăng ký: Giao diện đăng ký để truy cập vào trang chủ.

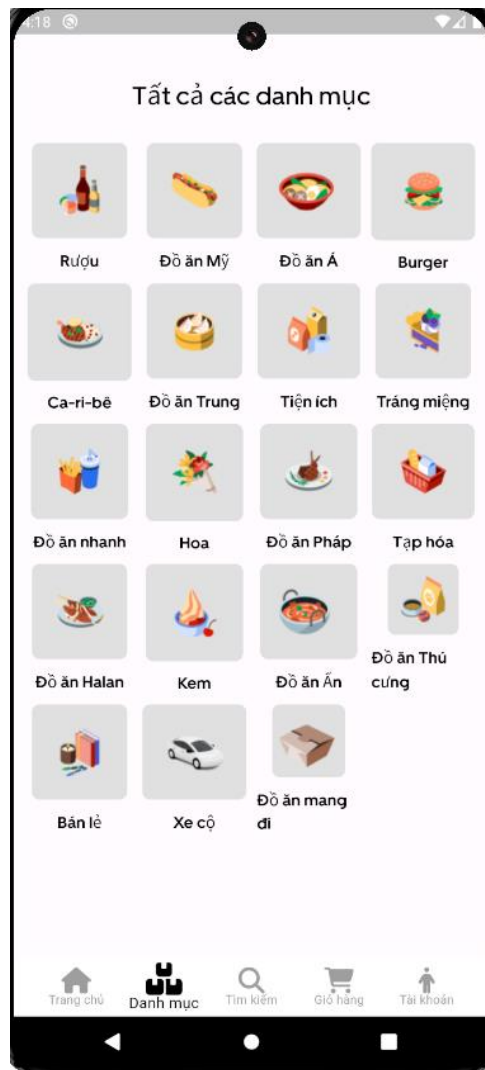
#### **4.2. Xây dựng trang chủ cho khách hàng**



*Hình 4.4 Giao diện trang chủ cho khách hàng*

Mô tả giao diện trang chủ cho khách hàng: Giao diện hiển thị thông tin món ăn và nhà hàng.

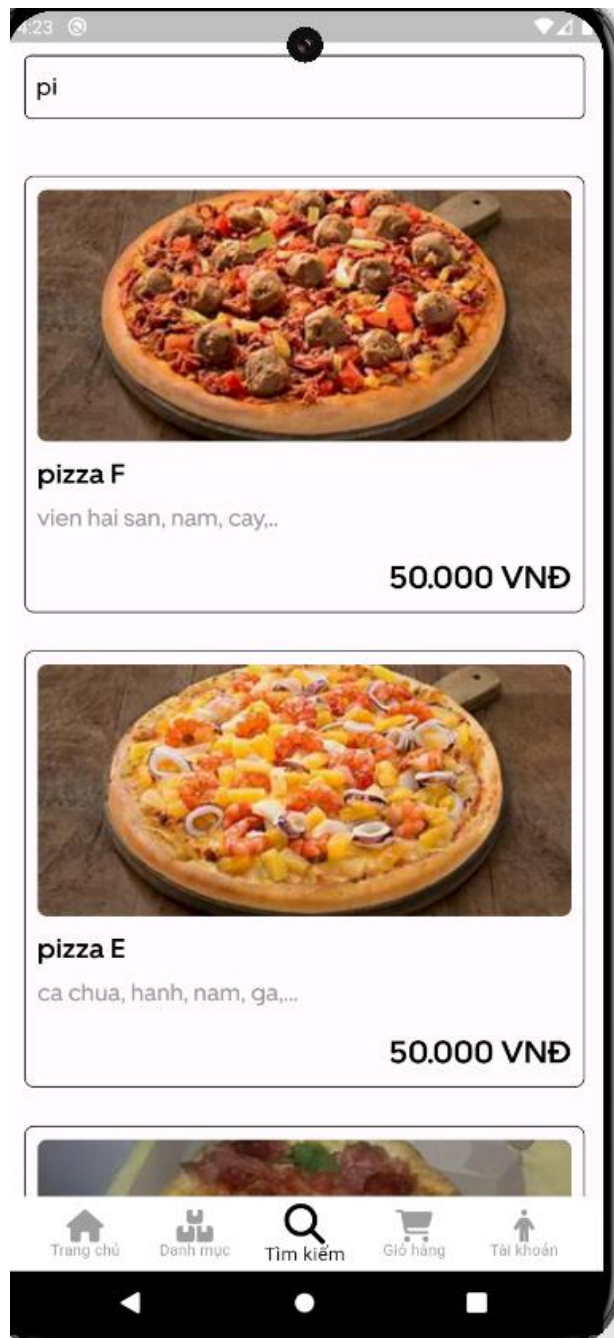
### **4.3. Xây dựng trang Danh mục cho khách hàng**



*Hình 4.5 Giao diện danh mục cho khách hàng*

Mô tả giao diện danh mục: Giao diện cho phép hiển thị tất cả các danh mục cho khách hàng.

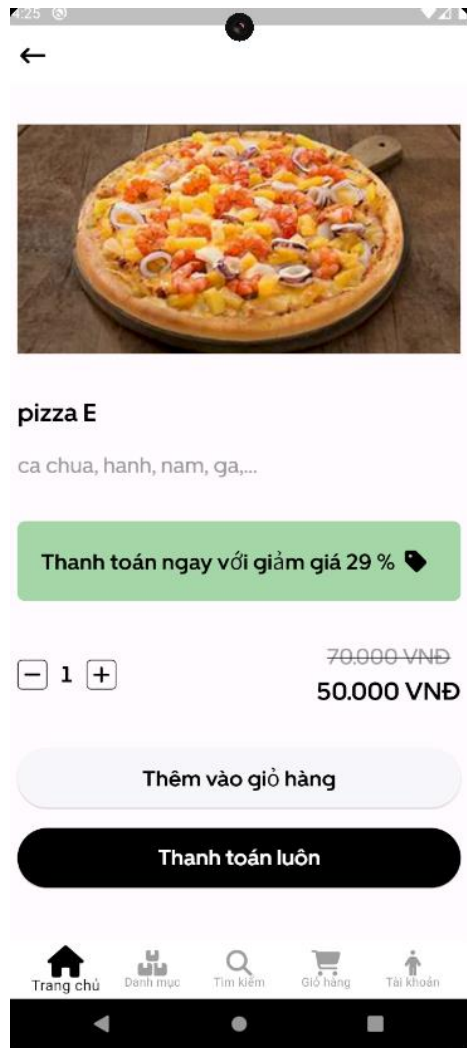
#### **4.4. Xây dựng trang Danh mục cho khách hàng**



Hình 4.6 Giao diện Tìm kiếm cho khách hàng

Mô tả giao diện tìm kiếm cho khách hàng: Giao diện hiển thị tất cả các danh mục cho Khách hàng.

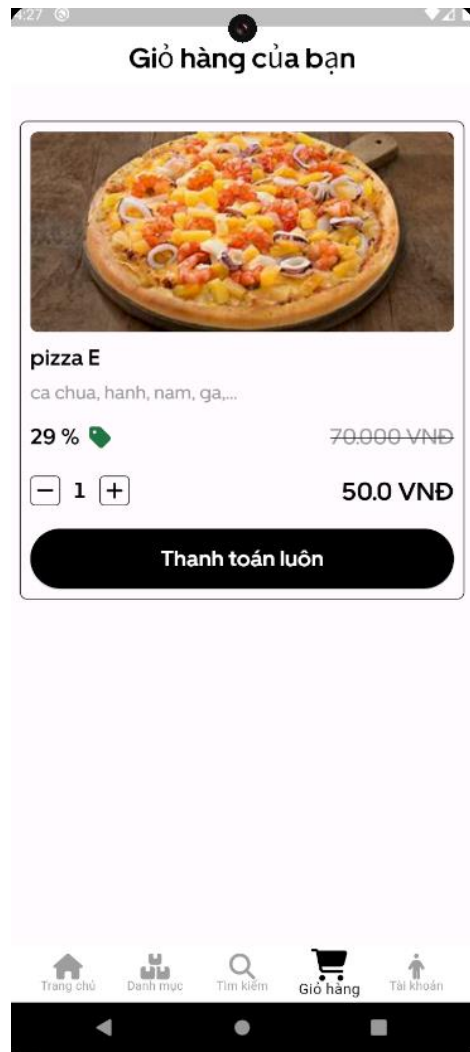
#### 4.5. Xây dựng trang Danh mục cho khách hàng



Hình 4.7 Giao diện Thông tin món ăn

Mô tả giao diện Thông tin món ăn: Giao diện hiển thị thông tin của món ăn mà khách hàng đã chọn.

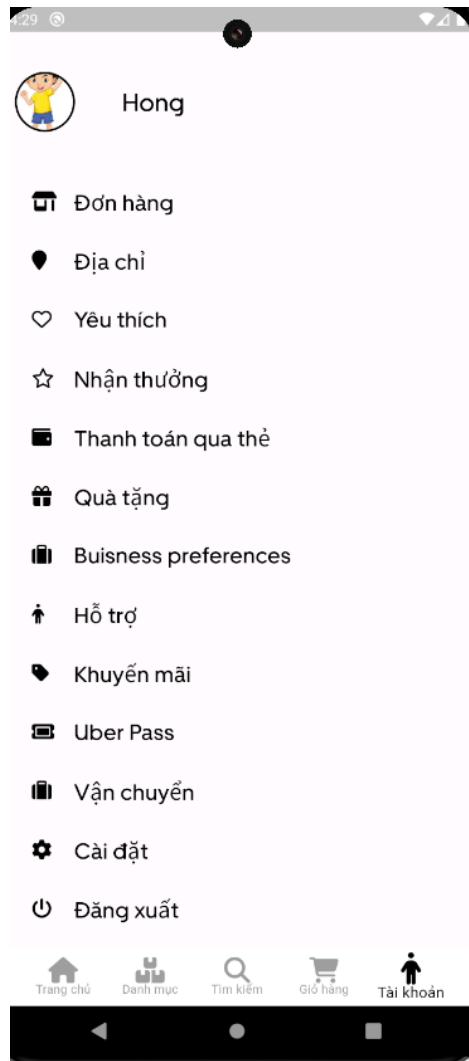
#### 4.6. Xây dựng trang giỏ hàng cho khách hàng



*Hình 4.8 Giao diện giỏ hàng*

Mô tả giao diện giỏ hàng: Giao diện hiển thị danh sách món ăn mà khách hàng đã chọn “Thêm vào giỏ hàng”.

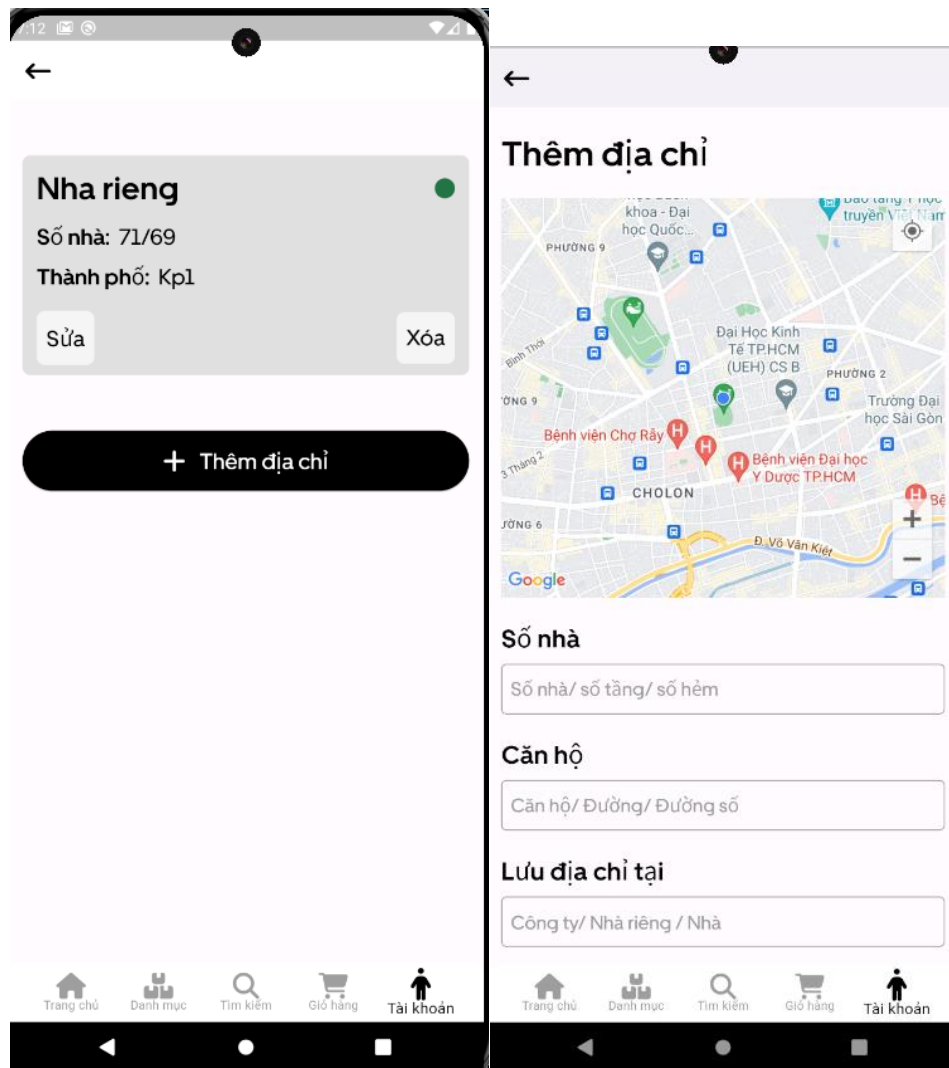
#### **4.7. Xây dựng trang tài khoản cho khách hàng**



*Hình 4.9 Giao diện tài khoản*

Mô tả giao diện tài khoản: Giao diện hiển thị các thông tin trong tài khoản của khách hàng.

#### **4.8. Xây dựng trang thêm địa chỉ mặc định cho khách hàng**



Hình 4.10: Giao diện thêm thông tin tài khoản

Mô tả giao diện thêm thông tin tài khoản: Giao diện hiển thị thông tin địa chỉ của khách hàng.

#### 4.9. Xây dựng trang phần thưởng cho khách hàng

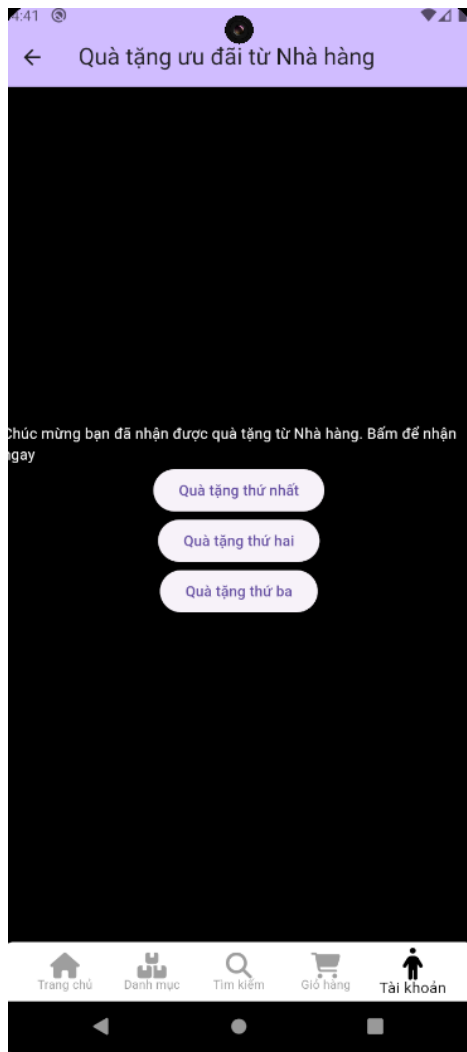




*Hình 4.11 Giao diện phần thưởng cho khách hàng*

Mô tả giao diện phần thưởng: Giao diện hiển thị thông tin phần thưởng cho khách hàng.

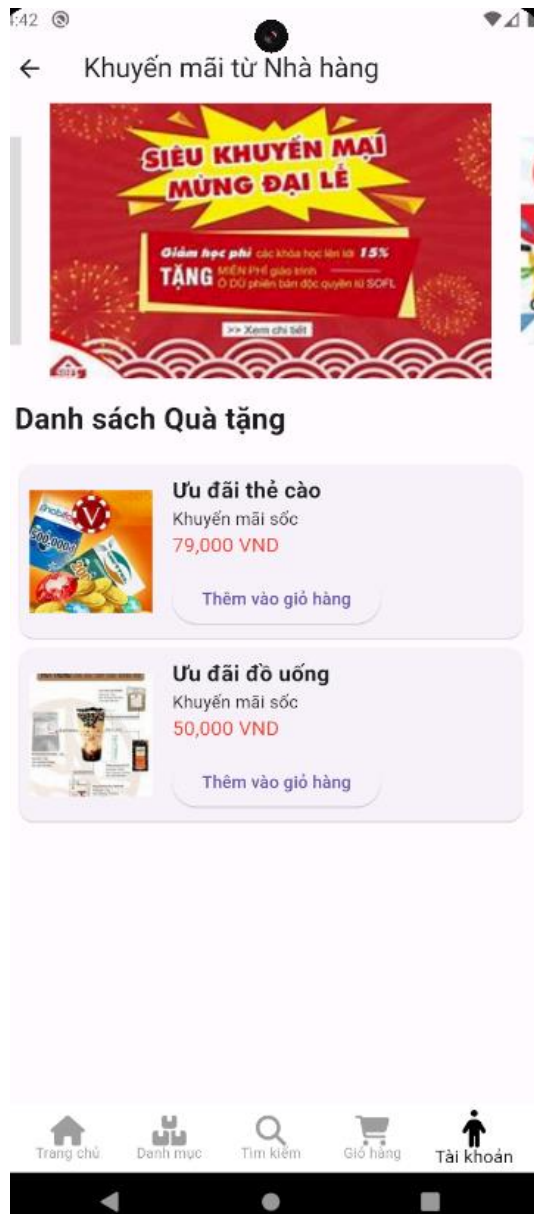
#### **4.10. Xây dựng trang quà tặng cho khách hàng**



*Hình 4.12 Giao diện quà tặng cho khách hàng*

Mô tả giao diện phần thưởng: Giao diện hiển thị thông tin quà tặng cho khách hàng.

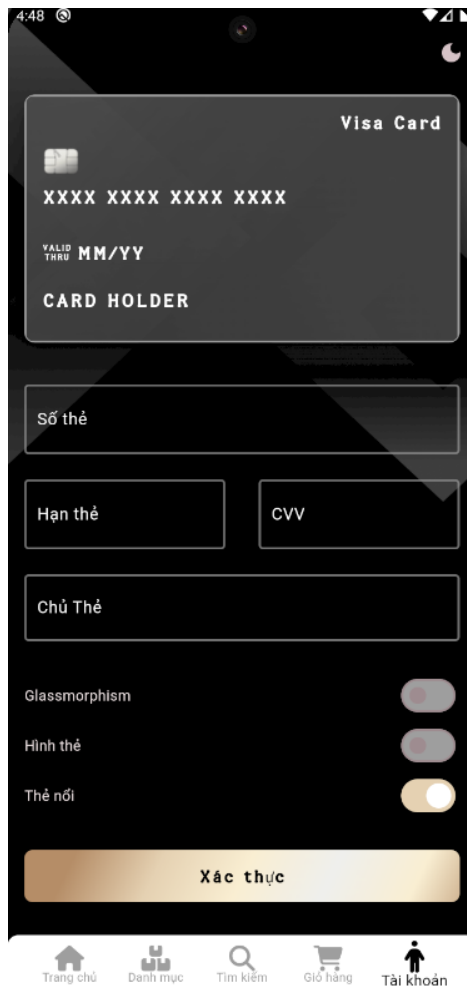
#### **4.11 Xây dựng trang khuyến mãi cho khách hàng**



Hình 4.13 Giao diện khuyến mãi

Mô tả giao diện phần thưởng: Giao diện hiển thị thông tin khuyến mãi cho khách hàng.

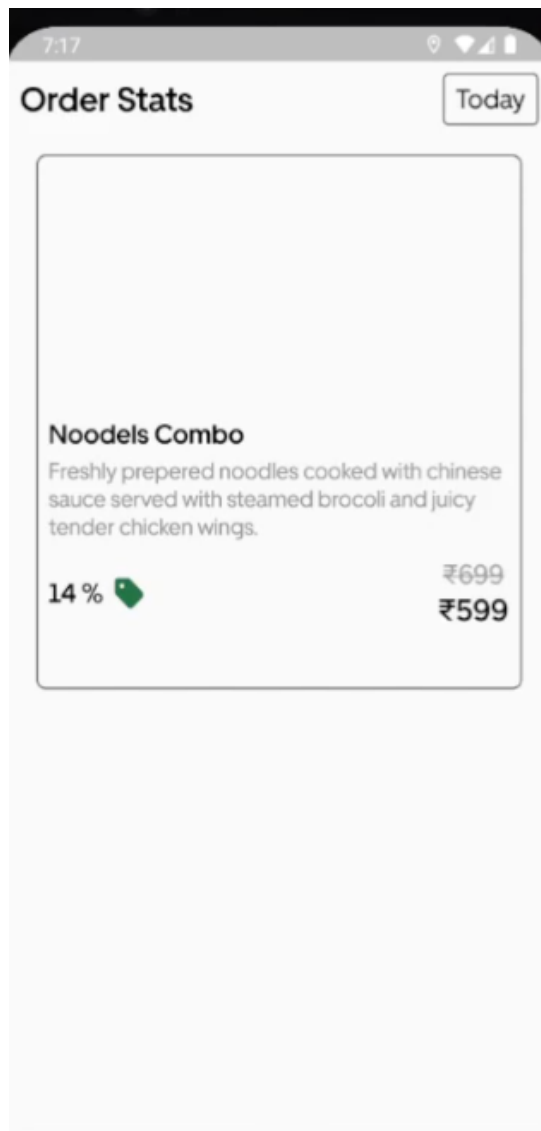
#### 4.12. Xây dựng trang thông tin thanh toán cho khách hàng



Hình 4.14 Giao diện thêm thông tin thanh toán tín dụng cho khách hàng

Mô tả giao diện thêm thông tin thanh toán: Giao diện yêu cầu người dùng xác định thẻ tín dụng của mình để tiện cho việc thanh toán Online.

#### 4.13. Xây dựng giao diện quản lý đơn hàng cho người dùng(khách hàng, nhà hàng và người vận chuyển đơn hàng)



*Hình 4.15 Giao diện quản lý đơn hàng*

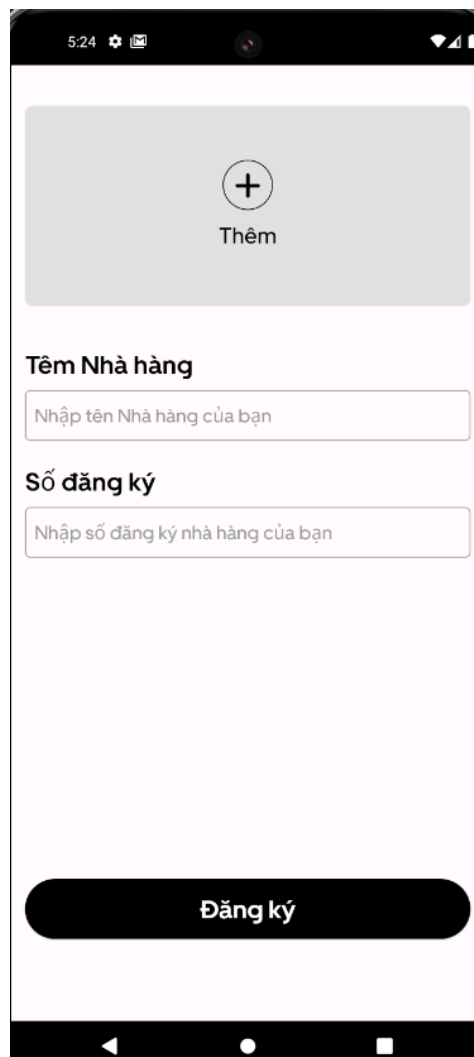
Mô tả giao diện trang quản lý đơn hàng: Giao diện hiển thị thông tin thống kê.

Thông tin món ăn.

Giá món ăn.

Giá vận chuyển.

#### 4.14. Xây dựng giao diện đăng ký cho nhà hàng.

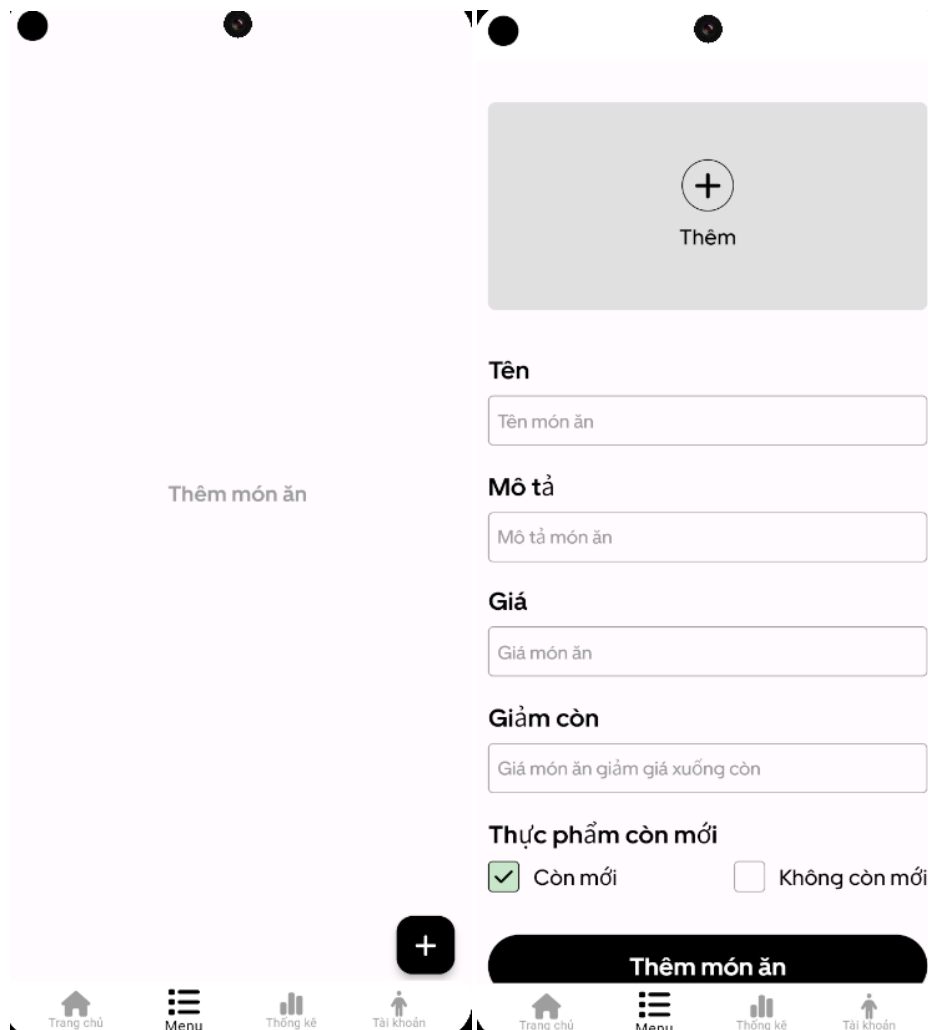


The screenshot shows a mobile application interface for restaurant registration. At the top, there is a status bar with the time 5:24 and various icons. Below the status bar is a large gray button with a plus sign icon and the text "Thêm". Underneath this button is a section titled "Tên Nhà hàng" (Restaurant Name) with a text input field containing the placeholder "Nhập tên Nhà hàng của bạn". Below the name field is a section titled "Số đăng ký" (Registration Number) with a text input field containing the placeholder "Nhập số đăng ký nhà hàng của bạn". At the bottom of the form is a large black button with the text "Đăng ký" (Register). The entire form is set against a light pink background.

Hình 4.16 Giao diện đăng ký cho Nhà hàng

Mô tả giao diện đổi đăng ký cho nhà hàng: Giao diện yêu cầu Nhà hàng đăng ký để truy cập vào trang chủ thành công.

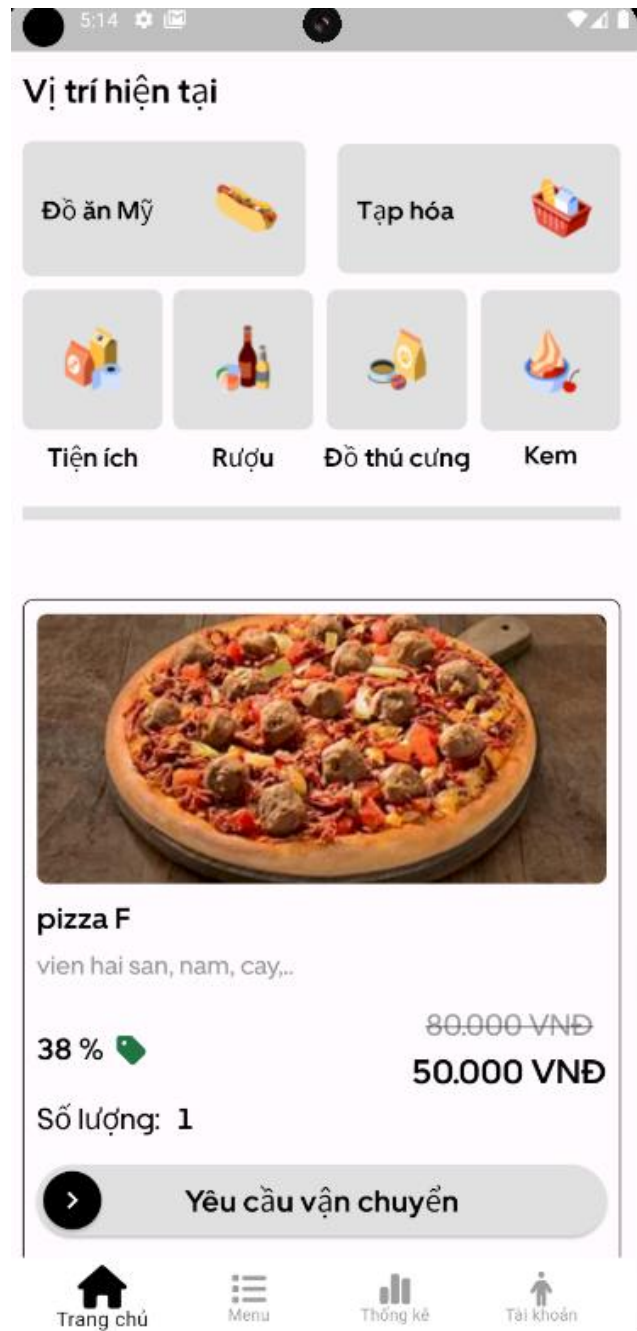
#### 4.15. Xây dựng giao diện thêm món ăn cho nhà hàng



Hình 5.17 Giao diện Thêm món ăn

Mô tả giao diện để thêm món ăn cho nhà hàng: Giao diện yêu cầu Nhà hàng thêm thông tin món ăn để thêm thành công món ăn hiển thị cho khách hàng.

#### 4.16. Xây dựng giao diện trang chủ cho nhà hàng



Hình 4.18 Giao diện Trang chủ cho nhà hàng

Mô tả giao diện trang chủ cho nhà hàng: Giao diện hiển thị thông tin món ăn được đặt hàng từ khách hàng.

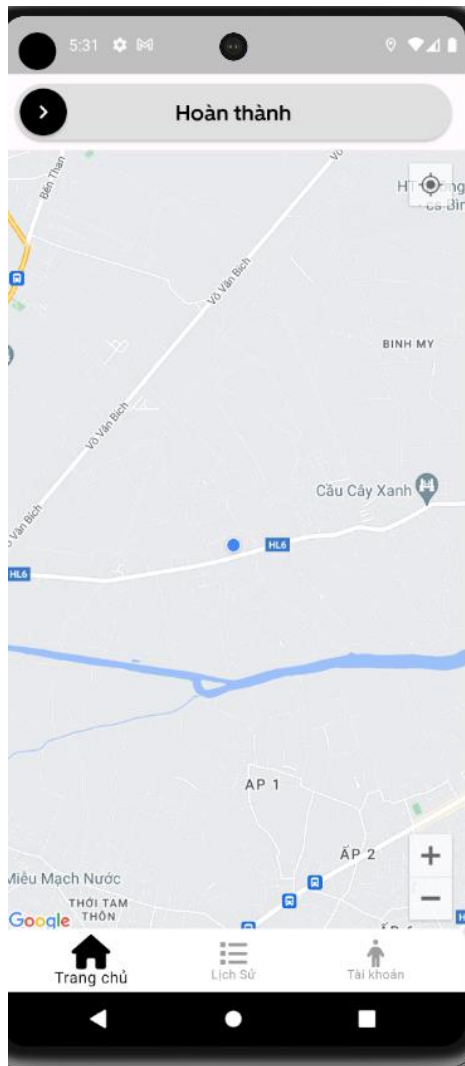
#### 4.17. Xây dựng trang đăng ký cho Nhà vận chuyển (tài xế)



*Hình 4.19 Giao diện Trang đăng ký cho tài xế*

Mô tả giao diện đăng ký thông tin cho tài xế giao hàng: Giao diện yêu cầu tài xế điền thông tin cá nhân chính xác để truy cập trang chủ dành cho tài xế.

#### **4.18. Xây dựng trang chủ cho tài xế**



*Hình 4.20 Giao diện Trang chủ cho tài xế*

Mô tả giao diện trang chủ cho tài xế: Hiển thị bản đồ , quãng đường vận chuyển đơn hàng

Giao diện yêu cầu: Tài xế chuyển sang chế độ trực tuyến và cập nhật tình trạng nhận đơn trước khi giao và hoàn thành đơn sau khi giao hàng thành công.

## KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 1. Nội dung đã đạt được

#### 1.1. Về lý thuyết

Dự án đã thành công trong việc phát triển một ứng dụng đặt hàng và giao hàng cơ bản cho người dùng.

Hệ thống cung cấp một nền tảng trực tuyến để đặt hàng và giao hàng. Khách hàng tìm kiếm món ăn, tìm các địa chỉ nhà hàng gần, lựa chọn món ăn, thêm vào giỏ hàng, cập nhật thông tin cá nhân, thanh toán món ăn...Nhà hàng thêm món ăn, thống kê doanh thu, yêu cầu tài xế nhận đơn.. Tài xế nhận yêu cầu của nhà hàng, có thông tin địa chỉ nhà hàng, khách hàng và giao hàng từ Nhà hàng đến địa chỉ của khách hàng.

Hệ thống được xây dựng bằng các công nghệ, thư viện như Dart, Flutter, Firebase. Nó hướng đến việc dễ dàng nâng cấp, bảo trì và phát triển các tính năng của ứng dụng.

Tuy nhiên thời gian có hạn và nội dung cần làm thì rất rộng nên em vẫn chưa giải quyết được hết tất cả vấn đề, một số chức năng vẫn chưa phát triển được.

#### 1.2. Về sản phẩm

Sản phẩm giải quyết được vấn đề giao hàng và đặt hàng cơ bản ở một số ứng dụng cần thiết cho người dùng hiện nay.

Khách hàng đăng ký thông tin, cập nhật địa chỉ lên hệ thống cơ sở dữ liệu. Từ đó có được thông tin địa chỉ của khách hàng, thuận tiện trong việc tìm các Nhà hàng gần với địa chỉ khách hàng. Khách hàng đặt hàng, hệ thống cơ sở dữ liệu lưu thông tin đơn hàng gồm thông tin nhà hàng, khách hàng và tài xế (nhà vận chuyển).

Nhà hàng đăng ký thông tin và thêm thông tin nhà hàng như địa chỉ, món ăn, hình ảnh quảng bá lên hệ thống cơ sở dữ liệu. Hệ thống tiếp nhận và hiển thị thông tin của nhà hàng lên trang chủ của khách hàng, khách hàng dễ dàng lựa chọn món ăn mình thích. Nhà hàng cũng có thể quản lý đơn hàng, doanh thu để dễ dàng rà soát tình trạng hiện tại của Nhà hàng.

Đối với tài xế ( nhà vận chuyển) sau khi đăng ký thành công, hệ thống cơ sở dữ liệu lưu thông tin đăng ký của tài xế gồm cả địa chỉ. Từ đó tài xế có thể thấy bản đồ ngay ở trang chủ, địa điểm , vị trí đơn hàng và của khách hàng. Thuận tiện trong việc tiếp nhận đơn hàng và giao đơn hàng.

## **2. Hạn chế của đề tài**

Đề tài “Xây dựng ứng dụng giao hàng đặt đồ ăn trực tuyến Uber Eats đa nền tảng Flutter với ngôn ngữ Dart” tuy đã hoàn thành xong vẫn không tránh khỏi những thiếu sót cũng như những hạn chế nhất định:

Ứng dụng vẫn còn thiếu chức năng, chưa hoàn thiện như: Chưa có các chức năng thống kê cũng như chưa thể in file thống kê, và một số chức năng phụ khác.

## **3. Hướng phát triển đề tài**

Để nâng cao chất lượng và hiệu quả của ứng dụng, có một số hướng phát triển tiềm năng sau đây:

- + Phân tích đối thủ cạnh tranh: Tìm hiểu các ứng dụng giao hàng hiện có, như Grab, Gojek, Baemin, Now, để hiểu rõ điểm mạnh và yếu của họ
- + Xác định đối tượng khách hàng: Đối tượng khách hàng mục tiêu là ai? (sinh viên, dân văn phòng, hộ gia đình, ...).
- + Đăng ký và đăng nhập: Cho phép người dùng đăng ký tài khoản và đăng nhập dễ dàng.
- + Duyệt và tìm kiếm nhà hàng: Người dùng có thể tìm kiếm và duyệt qua các nhà hàng.
- + Chọn món và đặt hàng: Người dùng có thể chọn món và đặt hàng trực tuyến.
- + Thanh toán: Hỗ trợ nhiều phương thức thanh toán như tiền mặt, thẻ tín dụng, ví điện tử.
- + Theo dõi đơn hàng: Cung cấp khả năng theo dõi tình trạng đơn hàng theo thời gian thực.
- + Đánh giá và phản hồi: Người dùng có thể đánh giá và phản hồi về dịch vụ và món ăn.
- + Thiết kế giao diện: Giao diện phải thân thiện, dễ sử dụng và trực quan.

- + Tối ưu hóa trải nghiệm người dùng: Tập trung vào việc giảm thiểu các bước không cần thiết để đặt hàng.
- + Lựa chọn công nghệ: Sử dụng các công nghệ phù hợp như Flutter cho phát triển ứng dụng di động đa nền tảng.
- + Phát triển backend: Xây dựng backend để quản lý cơ sở dữ liệu, xử lý yêu cầu từ ứng dụng di động và tương tác với hệ thống thanh toán.
- + Tích hợp API: Tích hợp các API như Google Maps để định vị, Firebase cho thông báo đẩy.
- + Kiểm thử ứng dụng: Thực hiện kiểm thử trên nhiều thiết bị và hệ điều hành để đảm bảo ứng dụng hoạt động mượt mà.
- + Triển khai ứng dụng: Đưa ứng dụng lên các cửa hàng ứng dụng (Google Play và App Store).
- + Chiến lược marketing: Sử dụng các chiến lược marketing trực tuyến và ngoại tuyến để thu hút người dùng mới.
- + Hợp tác với nhà hàng: Xây dựng mối quan hệ với các nhà hàng để mở rộng danh sách đối tác.
- + Chăm sóc khách hàng: Cung cấp dịch vụ chăm sóc khách hàng tận tâm để giữ chân người dùng.
- + Thu thập phản hồi người dùng: Liên tục thu thập phản hồi từ người dùng để cải thiện dịch vụ.
- + Phân tích dữ liệu: Sử dụng phân tích dữ liệu để hiểu rõ hơn về hành vi người dùng và tối ưu hóa ứng dụng.
- + Cải thiện giao diện người dùng: Tối ưu hóa giao diện người dùng để đảm bảo trải nghiệm người dùng tốt hơn và tăng khả năng sử dụng hệ thống.
- + Phát triển ứng dụng di động: Xây dựng ứng dụng di động (iOS, Android) để dễ dàng thực hiện quản lý trên đa nền tảng.

## TÀI LIỆU THAM KHẢO

- [1] "Flutter," [Online]. Available: [Học Flutter Từ Cơ Bản Đến Nâng Cao. Phần 1 - Viblo](#).
- [2] "Dart," [Online]. Available: [Tìm hiểu về ngôn ngữ Dart - Phần I \(viblo.asia\)](#).
- [3] "Firebase," [Online]. Available: [Firebase dùng trong Flutter \(Phần 1\): — techover.io](#)