



BÁO CÁO MINI-PROJECT GIAI ĐOẠN I

Lĩnh vực Software Engineering -Web Development
Đề tài: Xây dựng ứng dụng đặt đồ ăn quy trình logicstic

Trần Anh Tuấn
anhtuan9702@gmail.com

Hà Nội, tháng 6 năm 2024

Mục lục

Chương 1. Giới thiệu đề tài.....	3
1.1. Đặt vấn đề	3
1.2. Mục tiêu và phạm vi đề tài.....	3
1.3. Định hướng giải pháp.....	3
Chương 2. Công nghệ sử dụng.....	4
2.1. Spring Boot	4
2.2. ReactJS.....	4
2.3. MariaDB.....	4
2.4. Debezium	4
2.5. Elasticsearch	4
2.6. Mapbox	5
2.7. WebSocket	5
2.8. Cloudinary	5
2.9. Docker	5
Chương 3. Phân tích thiết kế xây dựng ứng dụng	6
3.1. Tổng quan chức năng.....	6
3.1.1. Biểu đồ ca sử dụng tổng quan	6
3.1.2. Biểu đồ hoạt động.....	6
3.2. Thiết kế ứng dụng	8
3.2.1. Lựa chọn kiến trúc.....	8
3.2.2. Thiết kế cơ sở dữ liệu.....	8
Chương 4: Triển khai cài đặt và thử nghiệm.....	10
4.1. Kết quả đạt được	10
4.2. Minh họa các chức năng chính	10
4.2.1. Giao diện danh Cửa hàng & Món ăn.....	10
4.2.2. Chức năng cập nhật vị trí	12
4.2.3. Chức năng Đăng nhập và Đăng ký	13
4.2.4. Chức năng Đặt hàng.....	14
4.2.5. Chức năng Quản lý cửa hàng	16
Kết luận và hướng phát triển	19

Chương 1. Giới thiệu đề tài

1.1. Đặt vấn đề

Trong bối cảnh xã hội hiện đại, việc đặt và giao đồ ăn trực tuyến đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày của nhiều người. Sự phát triển của công nghệ và nhu cầu tiện lợi đã thúc đẩy sự bùng nổ của các ứng dụng đặt đồ ăn như GrabFood, ShopeeFood hay beFood. Qua đề tài này em muốn tìm hiểu sâu hơn về quy trình hoạt động của các dịch vụ đặt đồ ăn thông qua việc nghiên cứu và xây dựng một hệ thống/ứng dụng đặt đồ ăn với quy trình logistic hiệu quả, có sử dụng bản đồ

1.2. Mục tiêu và phạm vi đề tài

Mục tiêu đề tài: Xây dựng được một ứng dụng phục vụ cho việc đặt đồ ăn, hoàn chỉnh từ việc khách hàng đặt hàng, nhà hàng xử lý đơn hàng, đến việc giao hàng tới khách hàng.

Phạm vi đề tài

- Tìm hiểu và ứng dụng các framework và công nghệ cần thiết: Bao gồm Spring Boot, ReactJS, MariaDB, Elastic Search, Debezium, Redis, Mapbox, WebSocket, Docker và Kubernetes.
- Phát triển các tính năng chính của hệ thống đặt đồ ăn: Gồm đặt hàng trực tuyến, quản lý đơn hàng, thông báo thời gian thực, quản lý phản hồi khách hàng và quản lý giao hàng.
- Triển khai và kiểm thử hệ thống: Đảm bảo hệ thống hoạt động chính xác và ổn định

1.3. Định hướng giải pháp

Để đạt được các mục tiêu đề ra, đề tài cần thực hiện các công việc:

- Nghiên cứu và phân tích các công nghệ và framework liên quan: Hiểu rõ các nguyên lý và cách thức hoạt động của Spring Boot, ReactJS, MariaDB, Elastic Search, Debezium, Redis, Mapbox, WebSocket, Docker và Kubernetes
- Thiết kế hệ thống: Bao gồm kiến trúc tổng thể, cấu trúc cơ sở dữ liệu, các module chức năng và giao diện người dùng.
- Phát triển và tích hợp các thành phần của hệ thống: Xây dựng từng module chức năng như đặt hàng, quản lý đơn hàng, thông báo, tìm kiếm,... Tích hợp các thành phần này vào hệ thống hoàn chỉnh.
- Triển khai và kiểm thử: Sử dụng Docker và Kubernetes để triển khai hệ thống. Thực hiện các kiểm tra để đảm bảo hệ thống thực hiện các chức năng đúng và ổn định

Chương 2. Công nghệ sử dụng

2.1. Spring Boot

Spring Boot là một framework mạnh mẽ và linh hoạt cho phép phát triển các ứng dụng Java một cách nhanh chóng và dễ dàng. Spring Boot được xây dựng dựa trên nền tảng Spring Framework, nó cung cấp các tính năng như tự động cấu hình, dịch vụ phụ trợ, và khả năng tích hợp với nhiều công nghệ khác.

Spring Boot cung cấp khả năng tích hợp với các công nghệ như Elastic Search, MariaDB và WebSocket thông qua các module và thư viện trong hệ sinh thái của Spring. Người dùng mới có thể dễ dàng thêm các thư viện này tại Spring Initializr - một công cụ trực tuyến được cung cấp bởi Spring để giúp người phát triển dễ dàng khởi tạo và cấu hình các dự án Spring Boot.

2.2. ReactJS

ReactJS là một thư viện JavaScript được phát triển bởi Facebook, sử dụng để xây dựng giao diện người dùng cho các ứng dụng web đơn trang (SPA). ReactJS cho phép phát triển các ứng dụng web một cách hiệu quả và linh hoạt thông qua việc sử dụng các thành phần có thể tái sử dụng.

2.3. MariaDB

MariaDB là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở được phát triển từ MySQL và được duy trì bởi cộng đồng. MariaDB hỗ trợ các kiểu dữ liệu địa lý thông qua các kiểu dữ liệu không gian (Spatial Data Types). Các kiểu dữ liệu này được sử dụng để lưu trữ và thao tác với các dữ liệu không gian như điểm, đường, và đa giác.

2.4. Debezium

Debezium được sử dụng để theo dõi các thay đổi trong cơ sở dữ liệu theo thời gian thực. Khi sử dụng cùng MariaDB, Debezium sẽ đọc các file binlog (binary log) của MariaDB để nhận biết và chuyển đổi các thay đổi này thành các sự kiện trong Kafka. Hiện nay với Debezium Embedded Engine cho phép tích hợp CDC trực tiếp vào ứng dụng mà không cần cấu hình thêm về Kafka.

2.5. Elasticsearch

Elasticsearch là một công cụ tìm kiếm và phân tích mạnh mẽ, cung cấp khả năng tìm kiếm toàn văn bản (full-text search), phân tích và lưu trữ dữ liệu theo thời gian thực. Elasticsearch thường được sử dụng trong các hệ thống log, phân tích dữ liệu và tìm kiếm phức tạp.

Dữ liệu trong Elasticsearch ở ứng dụng này được đồng bộ từ MariaDB thông qua các sự kiện được Debezium phát ra.

2.6. Mapbox

Mapbox là một nền tảng dịch vụ bản đồ và dữ liệu địa lý mạnh mẽ, cung cấp các công cụ và dịch vụ để xây dựng bản đồ tương tác, theo dõi và phân tích dữ liệu địa lý. Nó thường được sử dụng trong các ứng dụng web và di động để cung cấp các tính năng bản đồ phong phú và dễ tùy chỉnh.

Mapbox cung cấp nhiều API phục vụ cho các nhu cầu khác nhau về bản đồ và dữ liệu địa lý. Một số API chính của Mapbox bao gồm:

- Maps API: cho phép nhúng bản đồ tương tác vào ứng dụng
- Geocoding API: cung cấp khả năng chuyển đổi giữa địa chỉ và tọa độ địa lý.
- Directions API: cung cấp các tuyến đường lái xe, đi bộ, hoặc đi xe đạp giữa các điểm.
- Matrix API: cung cấp khả năng tính toán thời gian di chuyển và khoảng cách giữa nhiều điểm.

2.7. WebSocket

WebSocket là một giao thức giúp thiết lập kết nối hai chiều giữa client và server, cho phép trao đổi dữ liệu trong thời gian thực với độ trễ thấp. STOMP là một giao thức dùng để tương tác với các message broker. Nó được thiết kế để làm việc trên WebSocket, nhưng cũng có thể hoạt động trên các giao thức khác như TCP.

2.8. Cloudinary

Cloudinary là một nền tảng cung cấp các công cụ để lưu trữ, quản lý, chỉnh sửa thông qua các tham số URL và phân phối hình ảnh và video trên web và di động. Cloudinary cung cấp các SDK cho nhiều ngôn ngữ lập trình và nền tảng khác nhau như Java, JavaScript, Python, Go hay NodeJS

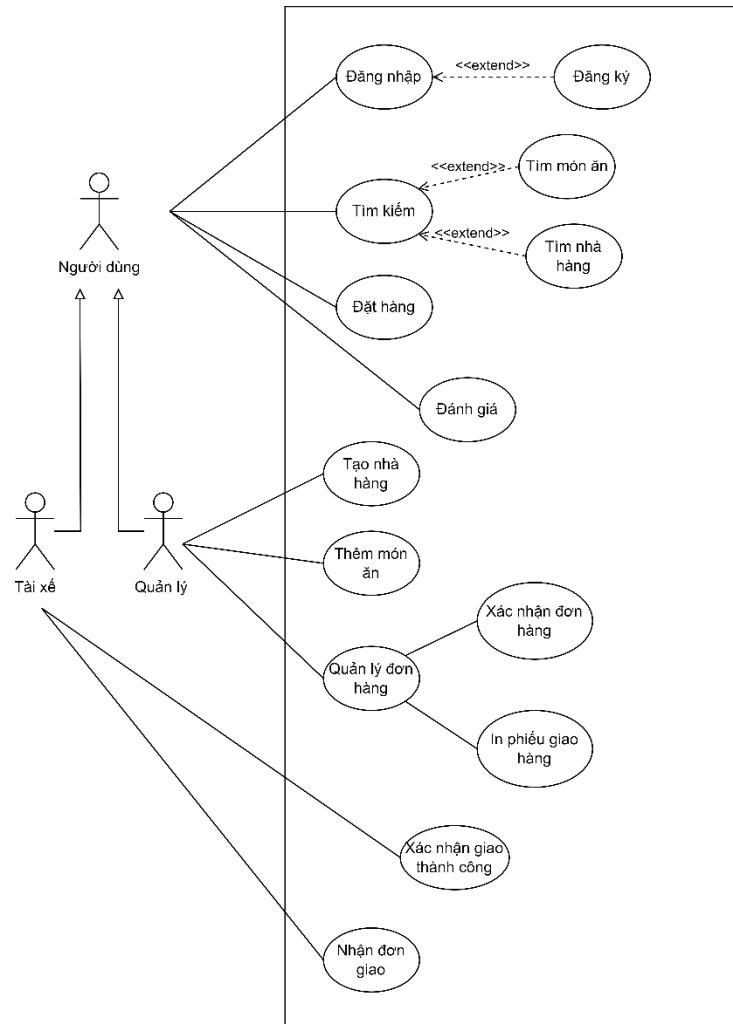
2.9. Docker

Docker là một nền tảng mã nguồn mở giúp xây dựng, vận chuyển và chạy các ứng dụng trong các container. Docker cung cấp một cách tiếp cận được tiêu chuẩn hóa để đóng gói ứng dụng và các phụ thuộc (dependencies) của nó thành một container độc lập, có thể chạy trên bất kỳ môi trường đã cài Docker

Chương 3. Phân tích thiết kế xây dựng ứng dụng

3.1. Tổng quan chức năng

3.1.1. Biểu đồ ca sử dụng tổng quan



Biểu đồ usecase tổng quan

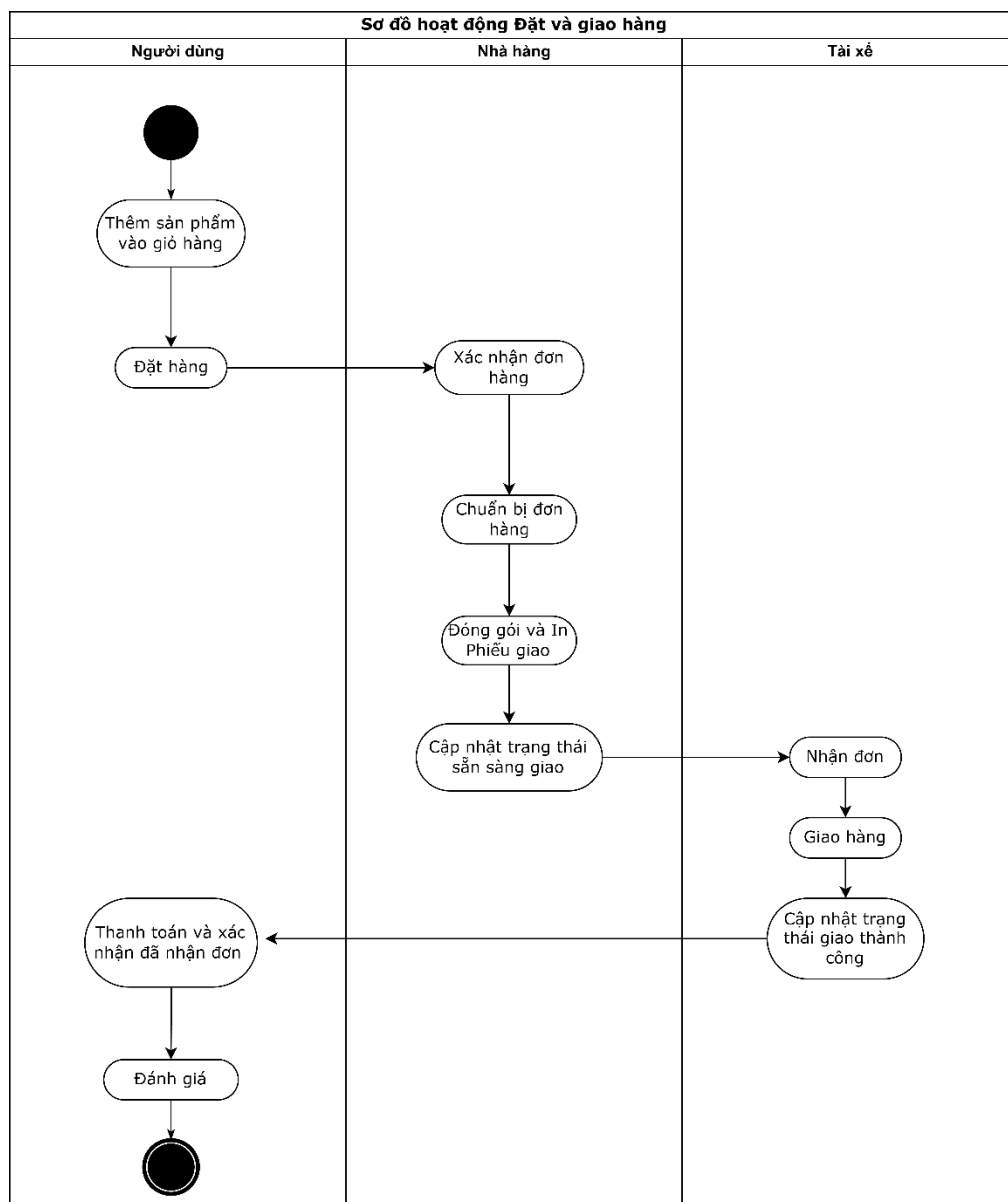
Ứng dụng hoàn chỉnh cần hỗ trợ được cho 3 đối tượng là Người dùng thông thường, Nhà hàng và Tài xế giao đồ ăn

- Người dùng có thể tìm kiếm nhà hàng và các món ăn ở khu vực lân cận và tiến hành đặt đồ
- Nhà hàng cần nhận được thông báo đặt đồ của khách hàng, chuẩn bị và đóng gói trước khi giao cho tài xế
- Tài xế nhận đơn và lấy đồ tại nhà hàng và giao cho khách

3.1.2. Biểu đồ hoạt động

Quy trình đặt đồ ăn theo chuẩn logistic bao gồm các bước từ khi khách hàng đặt hàng cho đến khi đơn hàng được giao thành công đến tay khách hàng diễn ra như sau:

- **Duyệt tìm kiếm:** Khách hàng chọn món ăn từ menu hoặc trang gợi ý trên ứng dụng
- **Đặt hàng:** Khách hàng tiến hành đặt hàng. Hệ thống xác nhận đơn hàng và thông báo cho khách hàng về tình trạng đơn hàng.
- **Xử lý đơn hàng:** Đơn hàng được gửi tới nhà hàng mà khách hàng đã chọn. Nhà hàng xác nhận đơn hàng và bắt đầu chuẩn bị món ăn.
- **Giao hàng:** Sau khi món ăn được chuẩn bị xong, nó sẽ được đóng gói và giao cho tài xế. Tài xế nhận đơn hàng từ nhà hàng và bắt đầu quá trình giao hàng đến địa chỉ của khách hàng.
- **Giao nhận và đánh giá:** Tài xế giao đơn hàng đến địa chỉ của khách hàng, xác nhận giao hàng thành công. Khách hàng nhận hàng và có thể cung cấp phản hồi về dịch vụ và chất lượng món ăn.



Biểu đồ hoạt động cho ca sử dụng Đặt đồ ăn

3.2. Thiết kế ứng dụng

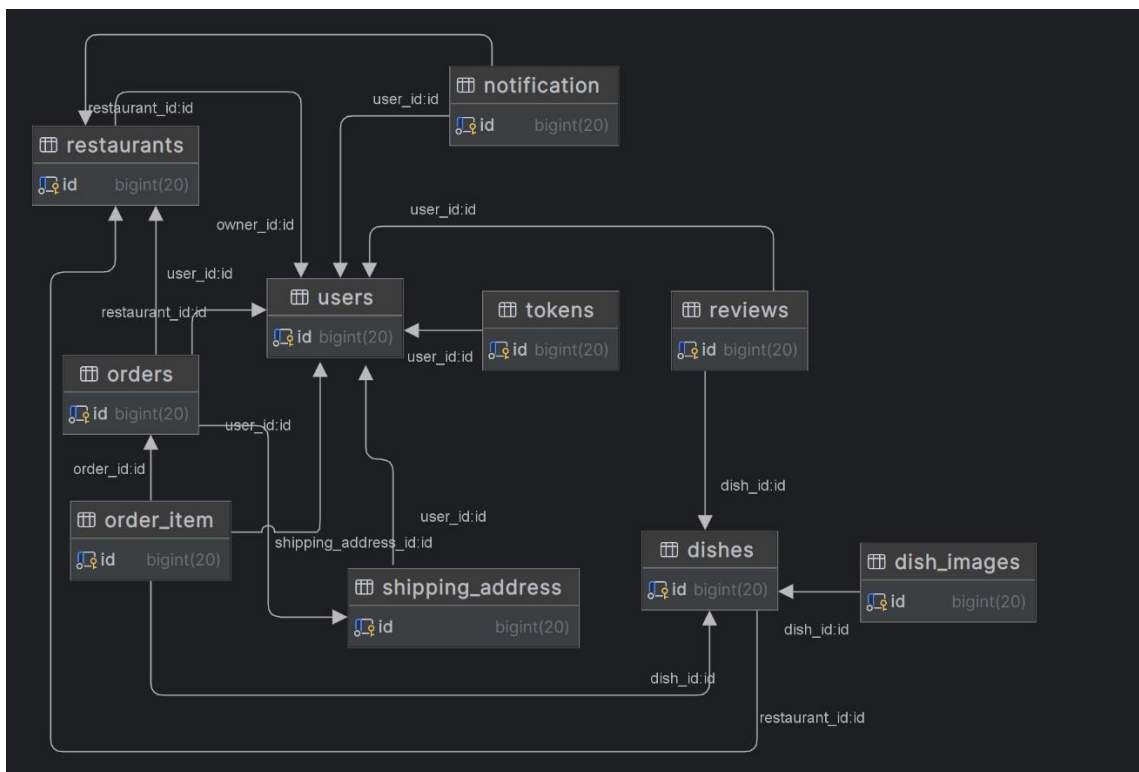
3.2.1. Lựa chọn kiến trúc

Sản phẩm áp dụng mô hình MVC (Model-View-Controller), là một mô hình được sử dụng phổ biến trong phát triển phần mềm. Mô hình MVC tách biệt các thành phần khác nhau của ứng dụng, giúp dễ dàng quản lý mã nguồn, mở rộng và bảo trì hệ thống.

Cụ thể chức năng của các thành phần:

- Thành phần Model và Controller nằm ở phía backend chịu trách nhiệm tương tác với cơ sở dữ liệu và xử lý các logic để trả dữ liệu cho View
- Thành phần View được xây dựng sử dụng ReactJS, có chức năng hiển thị giao diện người dùng và tương tác với backend thông qua các API.

3.2.2. Thiết kế cơ sở dữ liệu



Danh sách các bảng trong cơ sở dữ liệu

Thực thể	Chú thích	Thuộc tính
User	Người dùng	Họ tên, Email, Mật khẩu, URL ảnh đại diện
Shipping Address	Địa chỉ giao hàng của người dùng	Tên gọi nhớ, Địa chỉ, Tọa độ, Người nhận, SĐT người nhận
Restaurant	Nhà hàng	Tên, Địa chỉ, Tọa độ, Giờ mở/đóng, SĐT, Đánh giá, URL Logo

Dish	Món ăn	Tên, Mô tả, Giá gốc, Giá giảm, URL Ảnh bìa, URL Ảnh chi tiết, Số lượng còn lại, Số lượng đã bán, Đánh giá, Đơn vị (Hộp, Suất,...)
OrderItem	Món ăn được đặt	Món ăn, Người đặt, Số lượng, Đơn hàng (nếu không có thông tin đơn hàng thì sản phẩm đang ở trong giỏ)
Order	Danh sách các món ăn đã xác nhận đặt	Nhà hàng, Người đặt, Giá gốc, Giá giảm, Phí vận chuyển, Địa chỉ giao, Phương thức thanh toán, Thời điểm tạo đơn, Ghi chú, Trạng thái đơn, Tài xế
Review	Đánh giá của người dùng về món ăn	Đơn hàng, Món ăn, Người đánh giá, Nội dung, Đánh giá (theo số sao)
Notification	Thông báo	Nhà hàng, Người dùng, Nội dung, Đã xem (xác định người nhận đã đọc), Gửi từ người dùng (false nếu gửi từ nhà hàng tới người dùng, true nếu gửi từ người dùng)

Xác định liên kết giữa các thực thể

- User và Restaurant: Quan hệ 1:N, một người dùng có thể tạo nhiều nhà hàng
- User và Shipping Address: Quan hệ 1:N, một người dùng có thể có nhiều địa chỉ giao hàng
- User và OrderItem: Quan hệ 1:N, một người dùng có thể có nhiều sản phẩm trong giỏ hàng/có thể đặt nhiều sản phẩm
- User và Order: Quan hệ 1:N, một người dùng có thể có nhiều đơn hàng
- User và Review: Quan hệ 1:N, một người dùng có thể có nhiều đánh giá về món ăn mỗi khi đặt hàng
- Restaurant và Order: Quan hệ 1:N, một nhà hàng có thể nhận được nhiều đơn hàng
- Restaurant và Dish: Quan hệ 1:N, một nhà hàng có thể bán nhiều món ăn
- Order và OrderItem: Quan hệ 1:N, một đơn hàng có thể bao gồm nhiều sản phẩm
- Dish và Review: Quan hệ 1:N, một món ăn có thể có nhiều đánh giá khác nhau tương ứng với các đơn hàng
- Dish và DishImage: Quan hệ 1:N, một món ăn có thể có nhiều hình ảnh quảng cáo

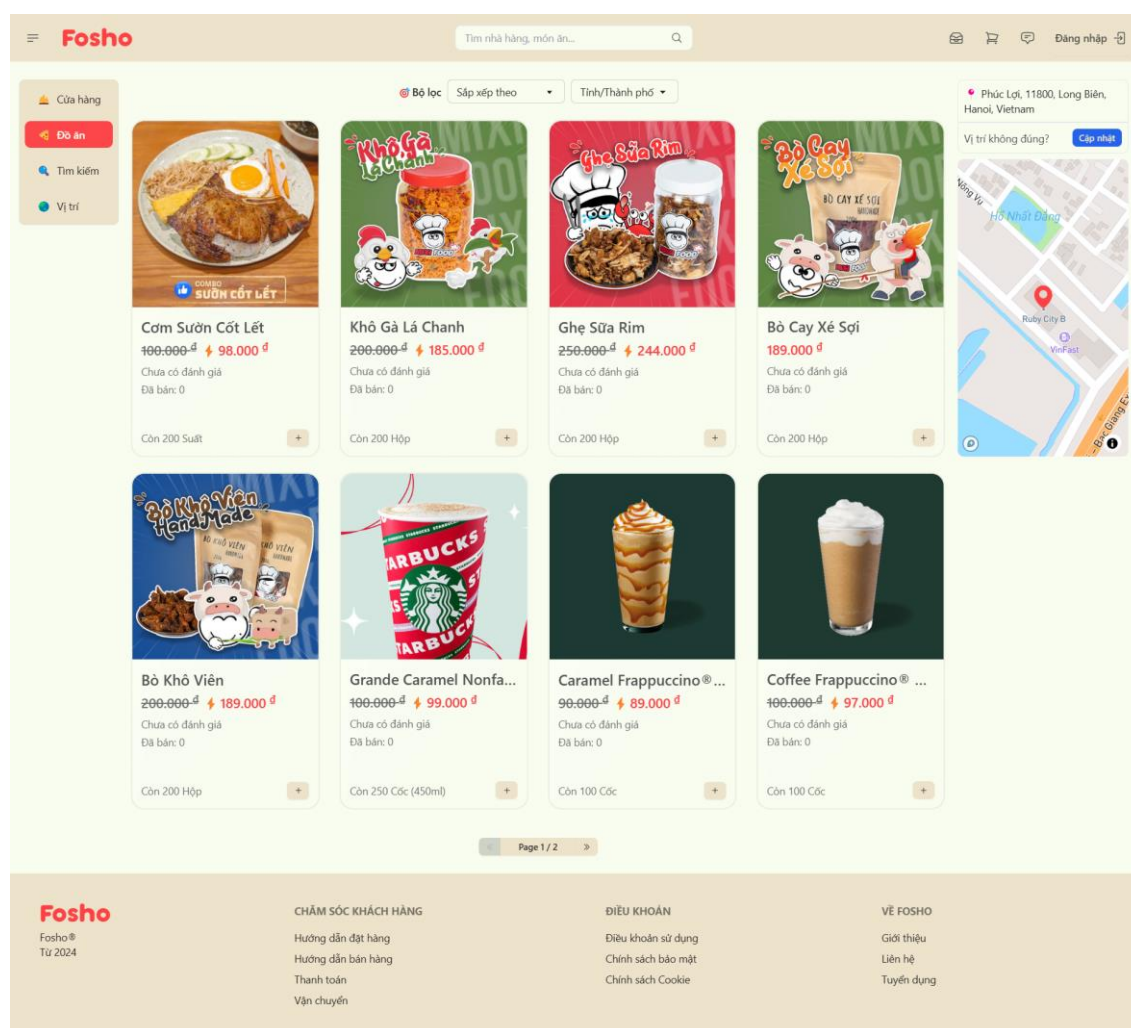
Chương 4: Triển khai cài đặt và thử nghiệm

4.1. Kết quả đạt được

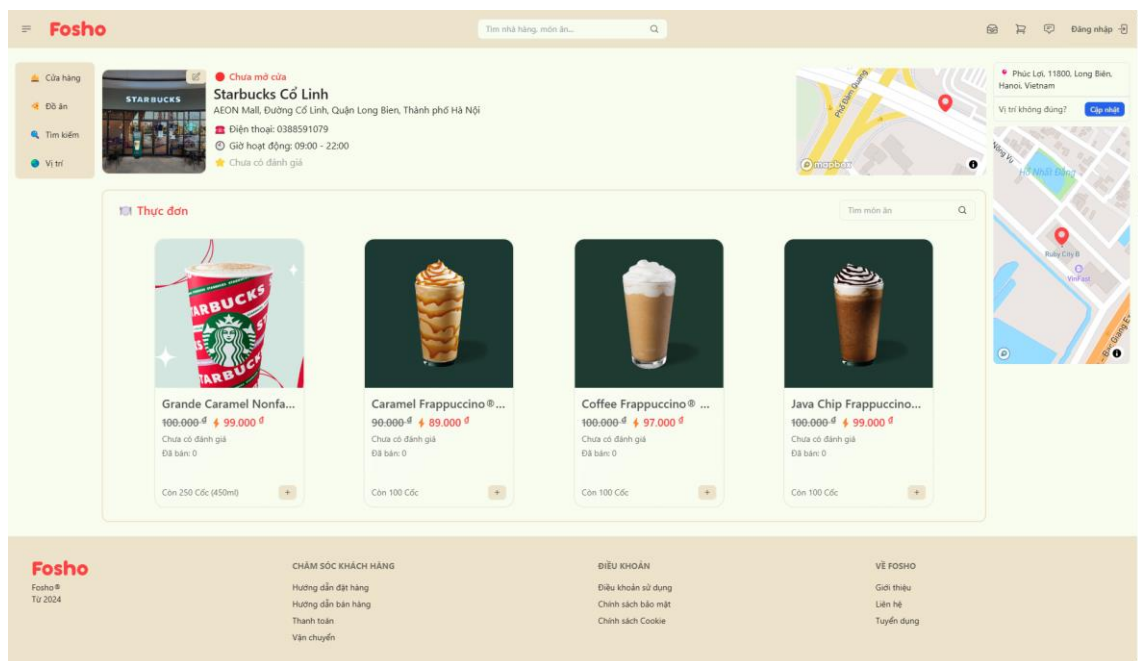
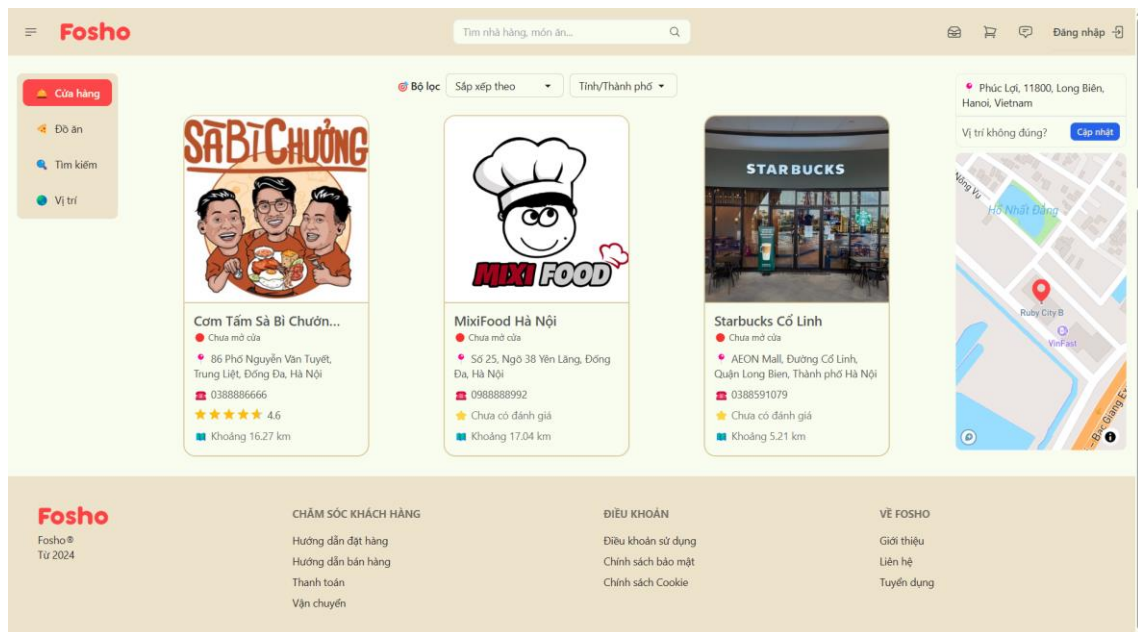
Sản phẩm đạt được là một ứng dụng đặt đồ ăn sử dụng bản đồ có hỗ trợ các chức năng như: Duyệt tìm kiếm nhà hàng, món ăn; Đăng nhập, Đăng ký; Giỏ hàng; Đặt hàng; Quản lý nhà hàng: Thực đơn, Đơn đặt,... Tuy nhiên vẫn còn có nhiều chức năng thiếu sót cần cải thiện để hỗ trợ tốt hơn cho người dùng và triển khai được trên thực tế.

4.2. Minh họa các chức năng chính

4.2.1. Giao diện danh Cửa hàng & Món ăn

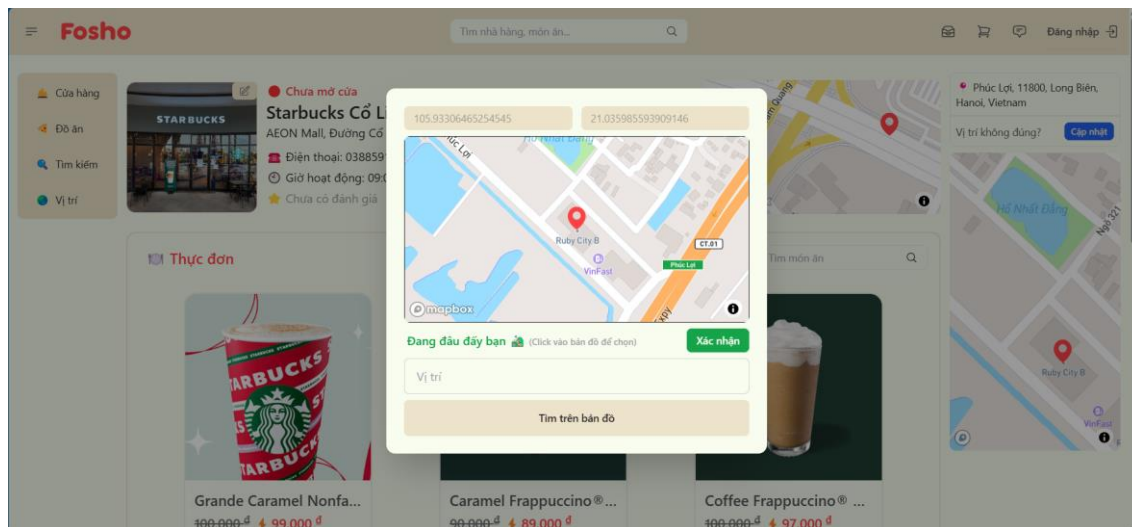


Giao diện duyệt danh sách món ăn

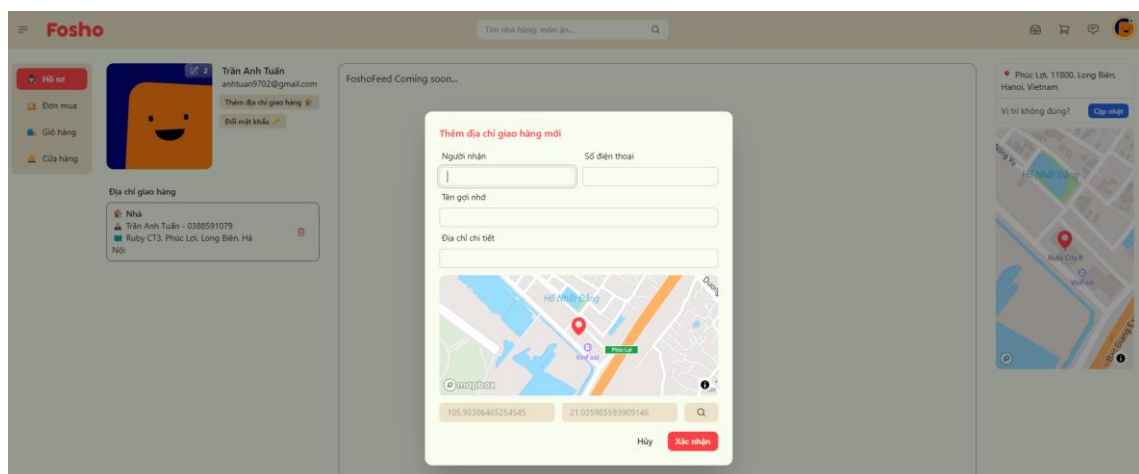


Khi chưa đăng nhập, người dùng có thể truy cập một giao diện của ứng dụng như giao diện danh sách cửa hàng, danh sách các món ăn của một cửa hàng và danh sách món ăn

4.2.2. Chức năng cập nhật vị trí




Ứng dụng cho phép người dùng cập nhật vị trí hiện tại của mình sử dụng bản đồ để đưa ra các nhà hàng ở gần chính xác hơn



Giao diện thêm địa chỉ giao hàng mới

4.2.3. Chức năng Đăng nhập và Đăng ký

Fosho




Đăng nhập vào Fosho
Email

Mật khẩu

[Quên mật khẩu?](#)
Đăng nhập
[Bạn chưa có tài khoản? Đăng ký ngay](#)
[Hoặc](#)
[Tiếp tục với Google](#)

Fosho



Chào mừng đến Fosho
Họ và tên

Email

Mật khẩu

Nhập lại mật khẩu

Đăng ký tài khoản
[Bạn đã có tài khoản? Đăng nhập](#)
[Hoặc](#)
[Tiếp tục với Google](#)

Fosho
Fosho®
Tư 2024

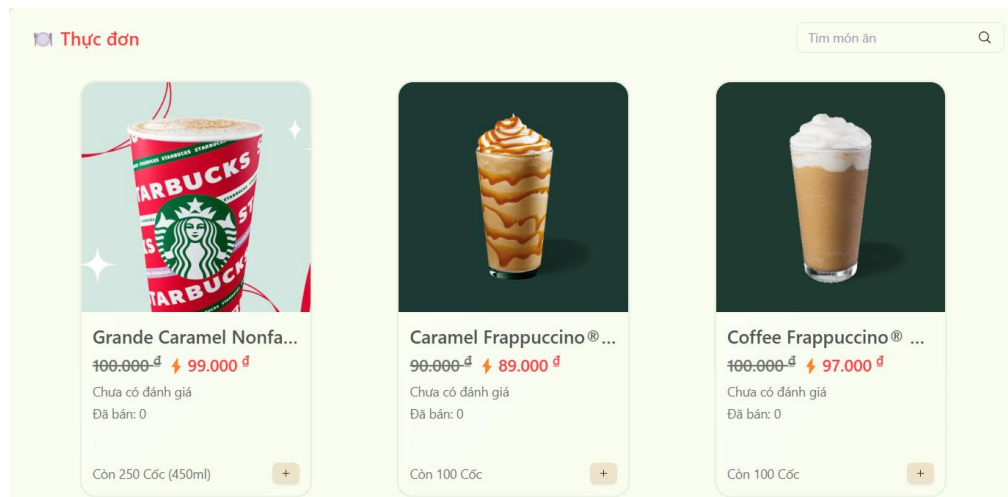
CHĂM SÓC KHÁCH HÀNG
[Hướng dẫn đặt hàng](#)
[Hướng dẫn bán hàng](#)
[Thanh toán](#)
[Vận chuyển](#)

ĐIỀU KHOẢN
[Điều khoản sử dụng](#)
[Chính sách bảo mật](#)
[Chính sách Cookie](#)

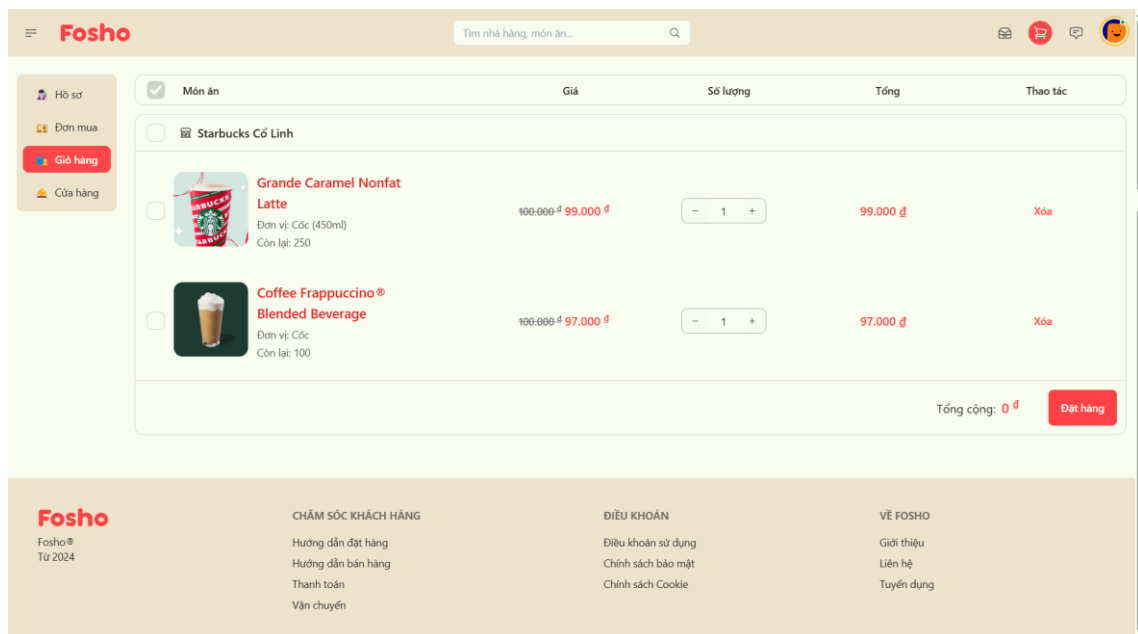
VỀ FOSHO
[Giới thiệu](#)
[Liên hệ](#)
[Tuyển dụng](#)

Để đặt hàng hay quản lý cửa hàng của mình, người dùng cần đăng ký hoặc đăng nhập khi đã có tài khoản

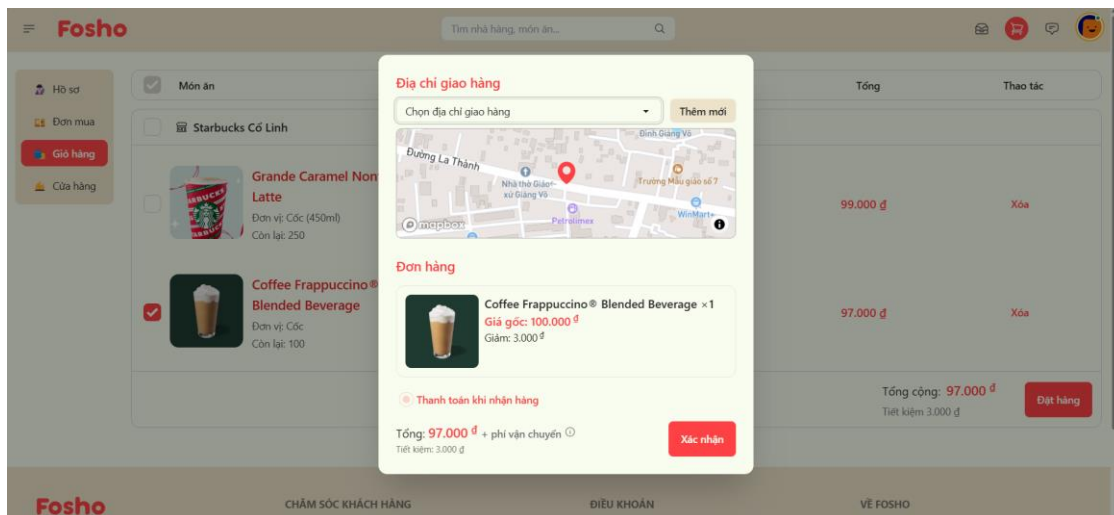
4.2.4. Chức năng Đặt hàng



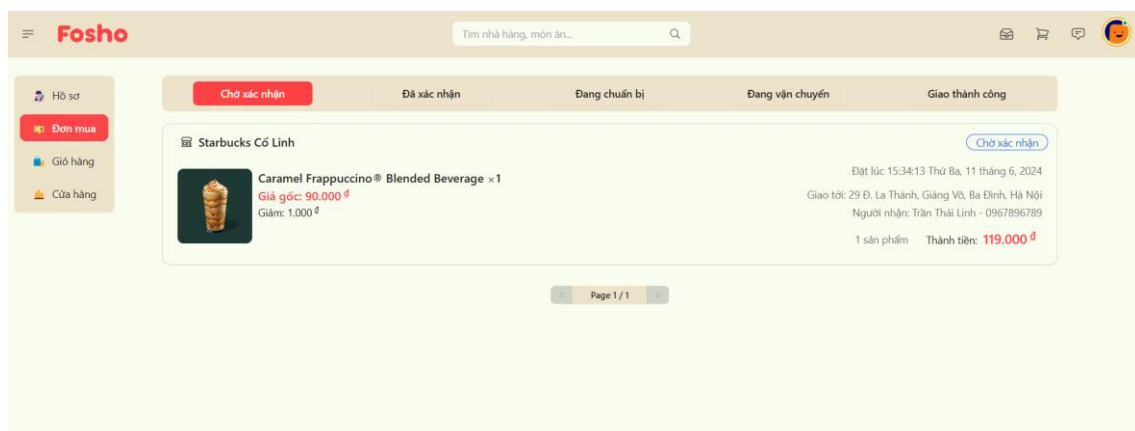
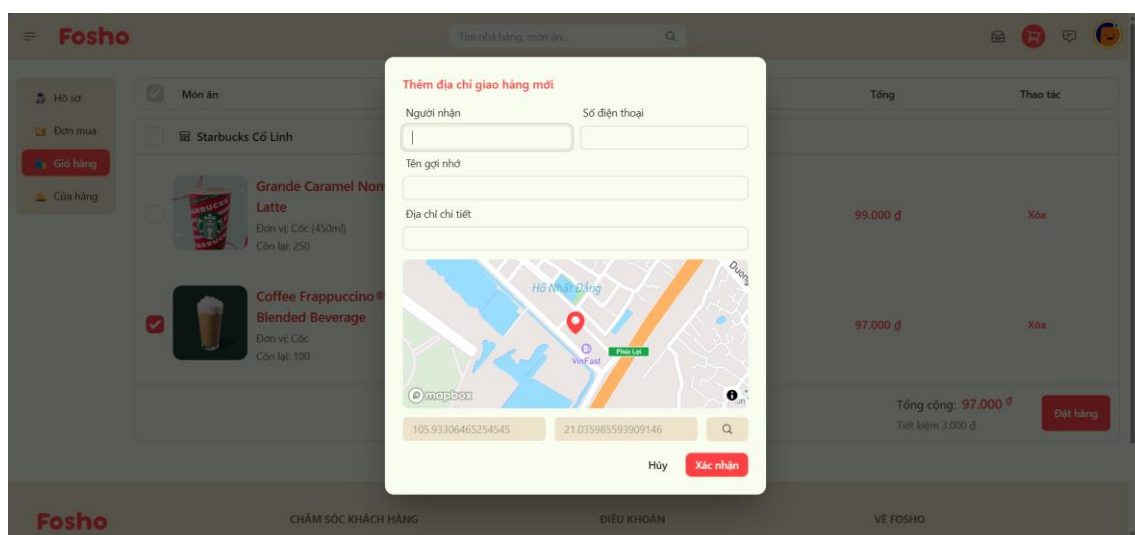
Người dùng chọn nút + để thêm sản phẩm vào giỏ hàng



Giao diện giỏ hàng

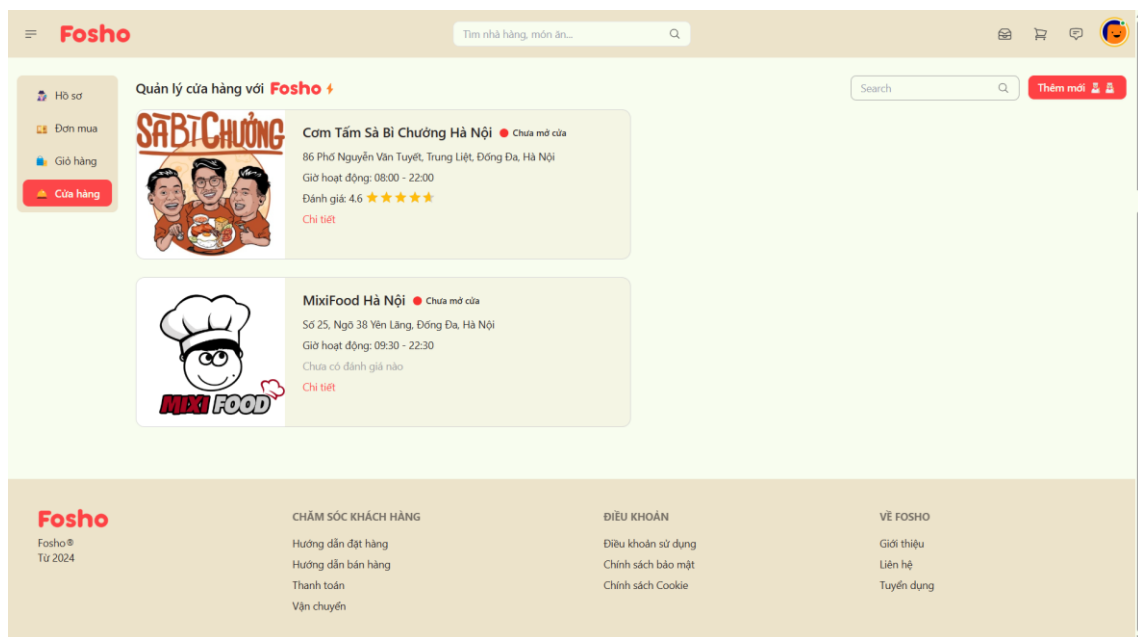


Người dùng chọn các sản phẩm muốn đặt và ấn nút Đặt hàng, tại giao diện xác nhận đơn hàng, người dùng chọn một địa chỉ giao hàng hoặc tạo một địa chỉ mới nếu chưa có

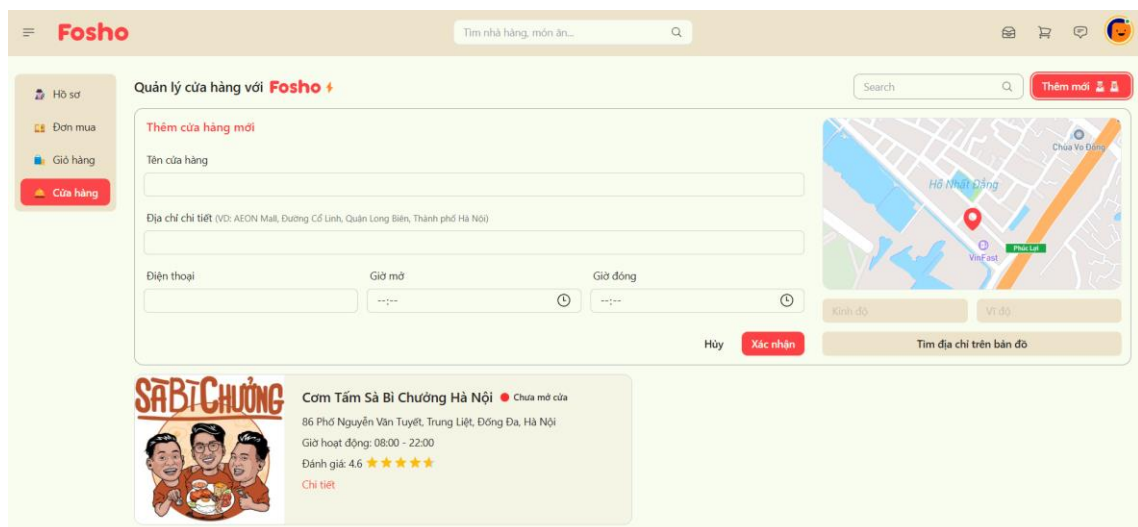


Giao diện đơn đặt hàng

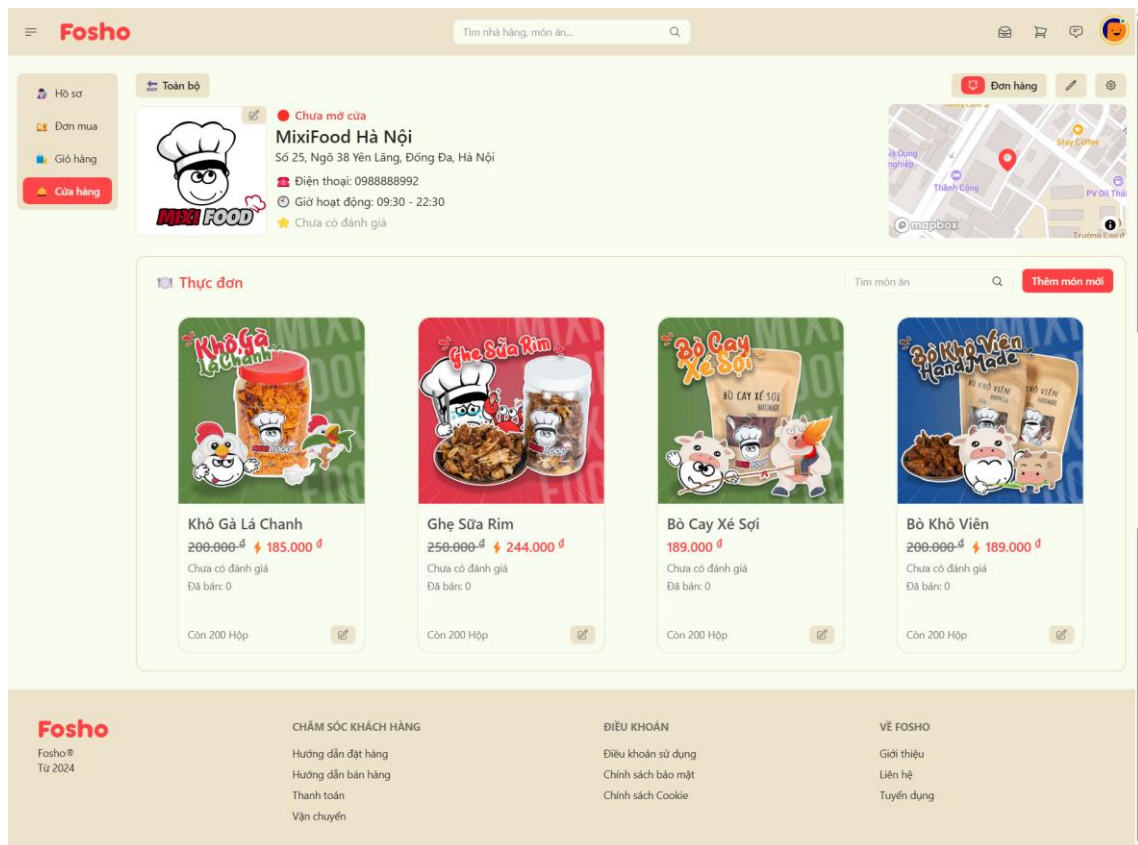
4.2.5. Chức năng Quản lý cửa hàng



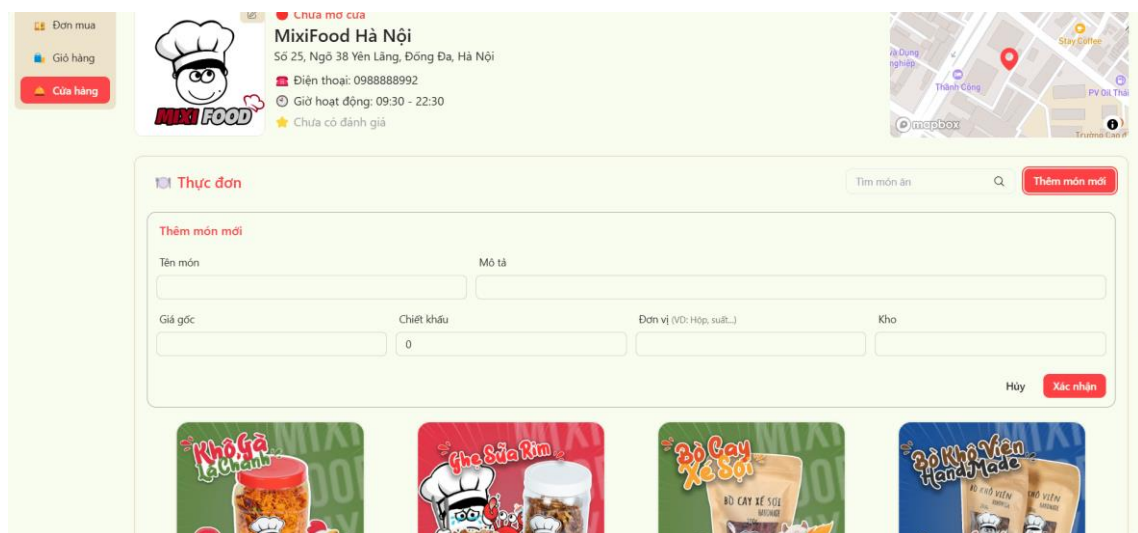
Giao diện danh sách cửa hàng



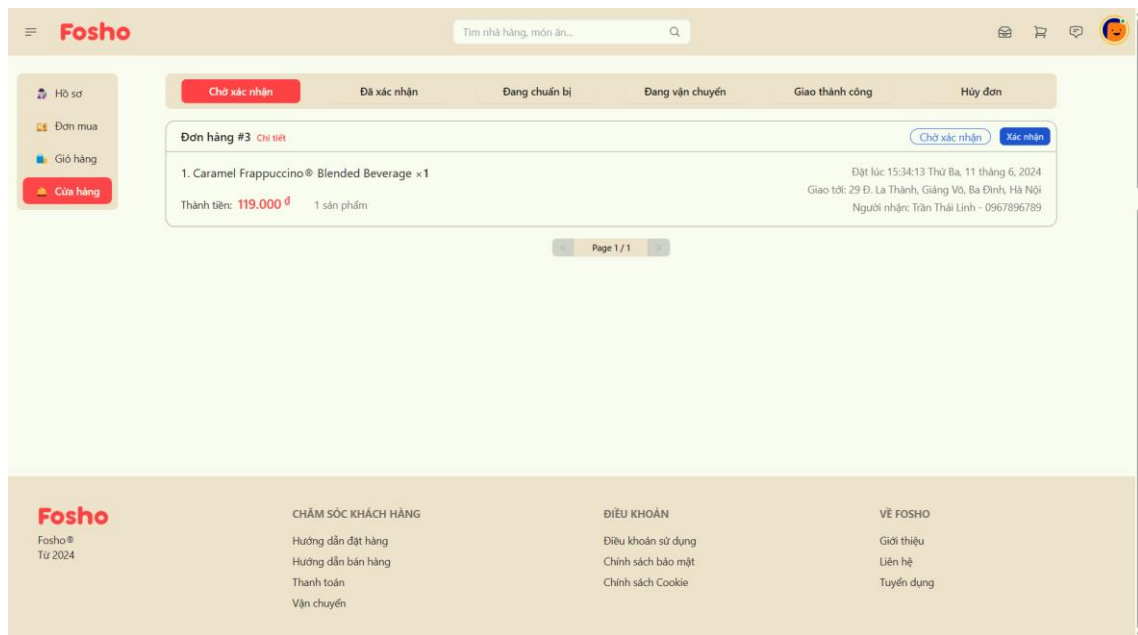
Người dùng có thể tạo cửa hàng mới để bắt đầu bán hàng



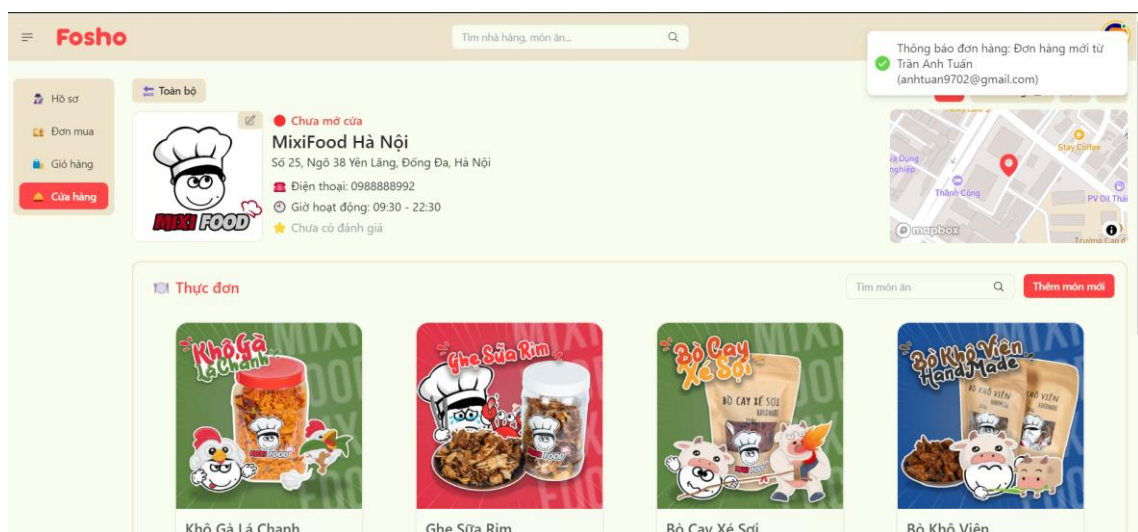
Giao diện quản lý của một cửa hàng, bao gồm danh sách các món đang bán



Giao diện thêm món ăn mới



Giao diện quản lý đơn hàng của cửa hàng



Giao diện thông báo và thông báo đơn hàng mới sử dụng WebSocket

Kết luận và hướng phát triển

Kết luận: Qua đồ án này, em đã tìm hiểu sâu hơn về quy trình đặt đồ ăn của một số ứng dụng đang hoạt động trên thực tế và từ đó xây dựng được một ứng dụng mô phỏng lại một số chức năng chính quan trọng

Hướng phát triển:

Với kinh nghiệm của bản thân còn nhiều hạn chế và điều kiện thời gian hữu hạn nên đồ án vẫn còn những tính năng chưa được cài đặt. Ứng dụng hiện chỉ dừng ở việc giúp người dùng đặt đồ ăn trực tuyến và giúp nhà hàng quản lý các đơn hàng. Một số chức năng cần bổ sung trong tương lai bao gồm:

- Thanh toán online
- Thông báo đẩy (Push Notification)
- Tính phí vận chuyển
- Các tính năng dành cho tài xế giao đồ ăn như nhận đơn, bản đồ định tuyến, xác nhận giao thành công
- Đánh giá đồ ăn của người dùng
- Bảng tin
- Caching dữ liệu sử dụng Redis

Tài liệu tham khảo

1. baeldung, W. by: (2024) Introduction to debezium, Baeldung. Available at: <https://www.baeldung.com/debezium-intro> (Accessed: 14 June 2024).
2. Debezium connector for mysql (no date) Debezium connector for MySQL :: Debezium Documentation. Available at: <https://debezium.io/documentation/reference/2.6/connectors/mysql.html> (Accessed: 14 June 2024).
3. Install Elasticsearch with Docker: Elasticsearch Guide [8.14] (no date) Elastic. Available at: <https://www.elastic.co/guide/en/elasticsearch/reference/current/docker.html> (Accessed: 14 June 2024).
4. [Spring Boot 3 + Spring Security 6 - JWT Authentication and Authorisation \[NEW\] \[2023\] - YouTube](#)
5. [Image Upload with Cloudinary using Spring Boot, Angular, and MySQL - YouTube](#)