

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING



CÔNG NGHỆ PHẦN MỀM (CO3001)

Bài tập lớn

RESTAURANT POS 2.0 - Task 2

GVHD: Lê Đình Thuận
Nhóm sinh viên thực hiện: Nguyễn Quang Anh - 1912594
Cù Đỗ Thanh Nhân - 1914424
Nguyễn Hữu Phúc - 1914698
Hoàng Minh Tiến - 1915470
Lâm Thiện Toàn - 1915540
Ngô Đức Trí - 1915656
Nguyễn Phúc Vinh - 1915940

HO CHI MINH CITY, SEPTEMBER 2021



Mục lục

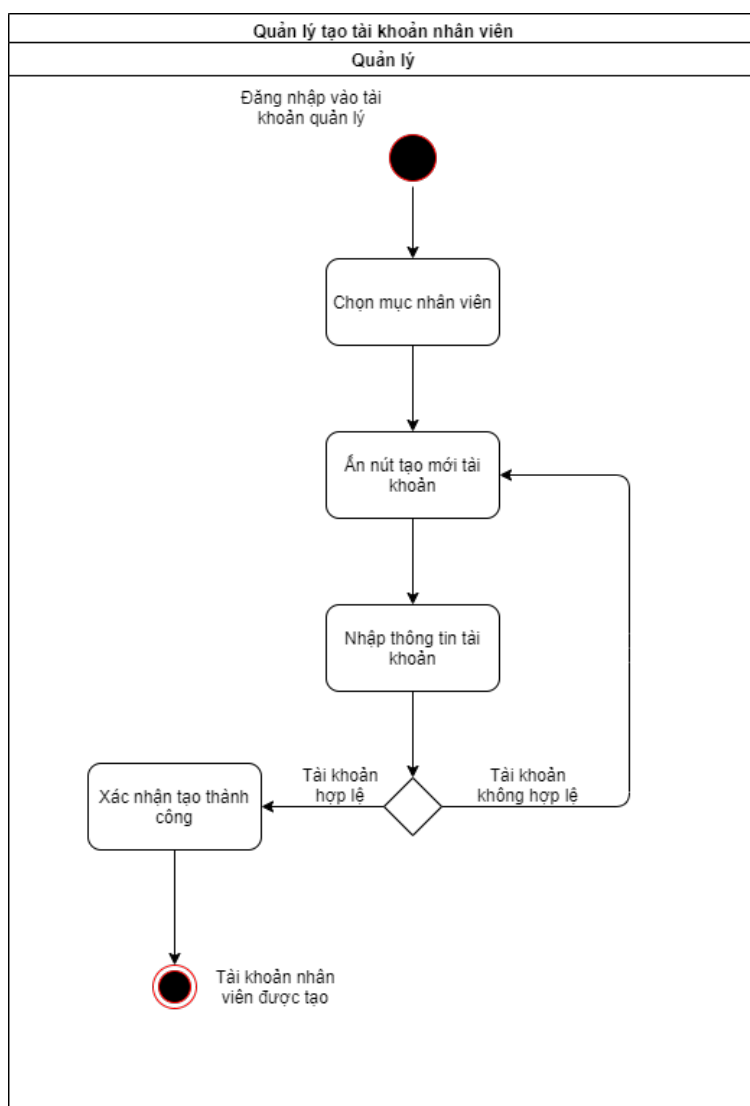
| | | |
|----------|--|-----------|
| 1 | Activity diagram | 2 |
| 1.1 | Tài khoản | 2 |
| 1.2 | Khám phá món ăn | 7 |
| 1.3 | Quản lý món ăn | 8 |
| 1.4 | Đặt đồ ăn và thanh toán | 13 |
| 2 | Sequence diagram | 19 |
| 2.1 | Quản lý tài khoản | 19 |
| 2.2 | Khám phá món ăn | 22 |
| 2.3 | Quản lý món ăn | 23 |
| 2.4 | Đặt đồ ăn và thanh toán online | 27 |
| 3 | Class diagram | 34 |

1 Activity diagram

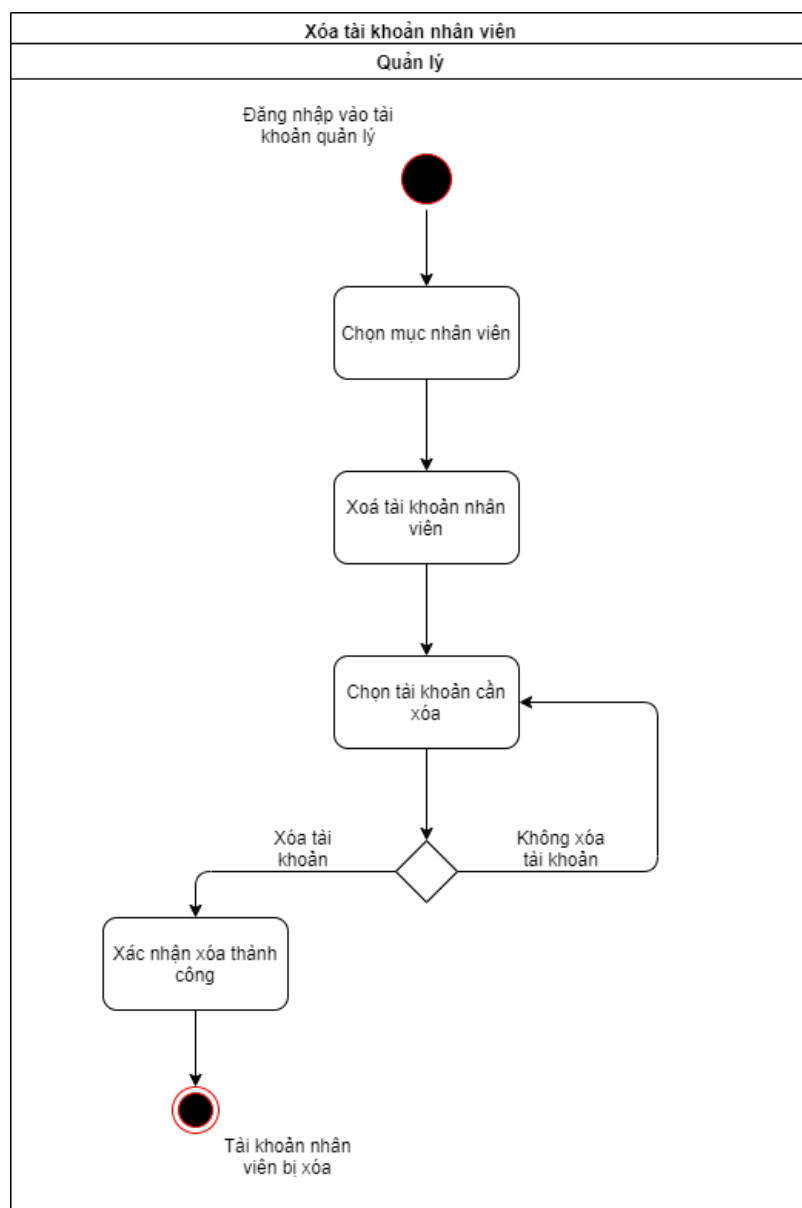
Activity diagram là biểu đồ hành vi UML thể hiện luồng điều khiển (flow of control) hoặc luồng đối tượng (object flow) với sự nhấn mạnh vào trình tự và điều kiện của luồng. Các hành động được điều phối bởi mô hình hoạt động có thể được khởi tạo bởi các hành động khác kết thúc quá trình thực thi, vì các đối tượng và dữ liệu trở nên khả dụng hoặc do một số sự kiện bên ngoài quy trình xảy ra.

1.1 Tài khoản

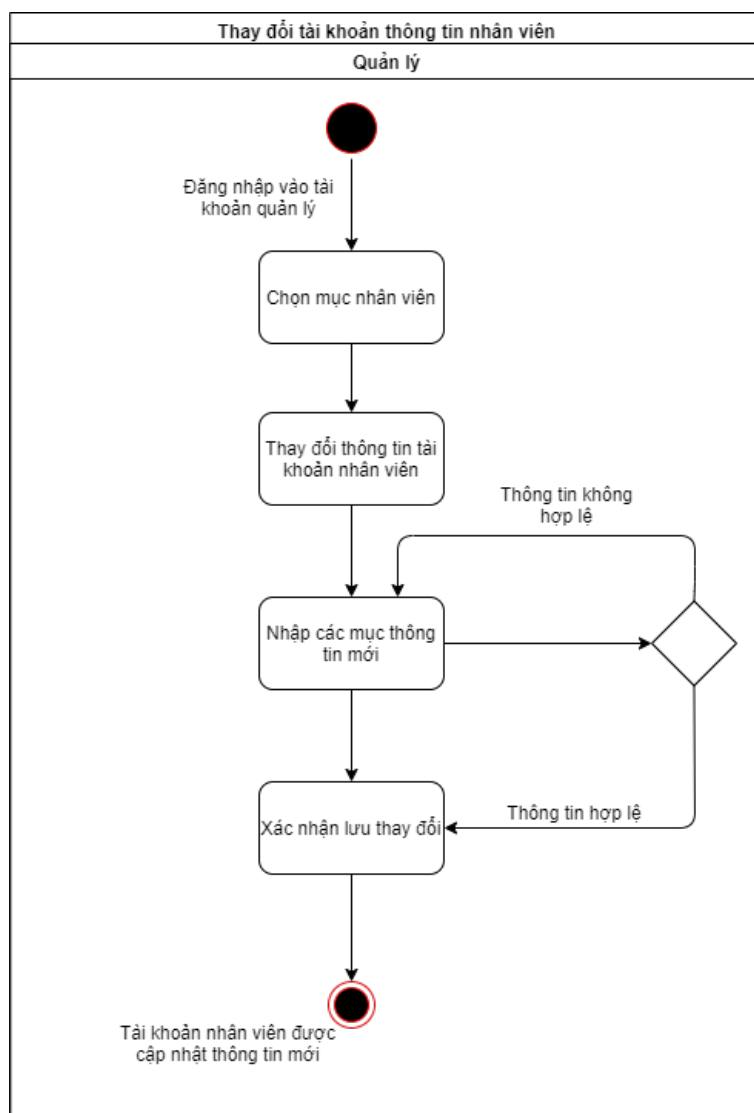
Dưới đây là các activity diagram tương ứng với năm functional requirements:



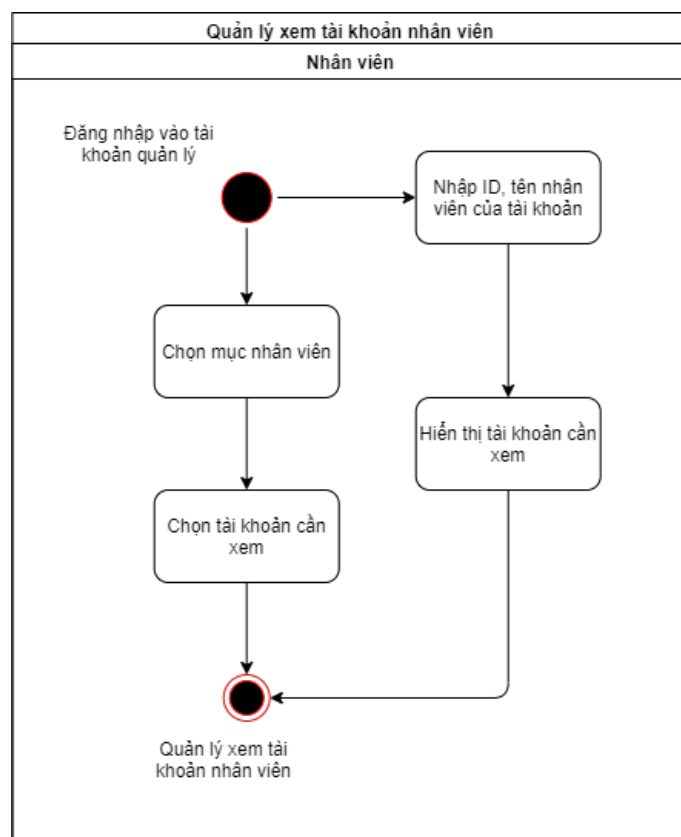
Hình 1: Activity diagram Quản lý tạo tài khoản nhân viên



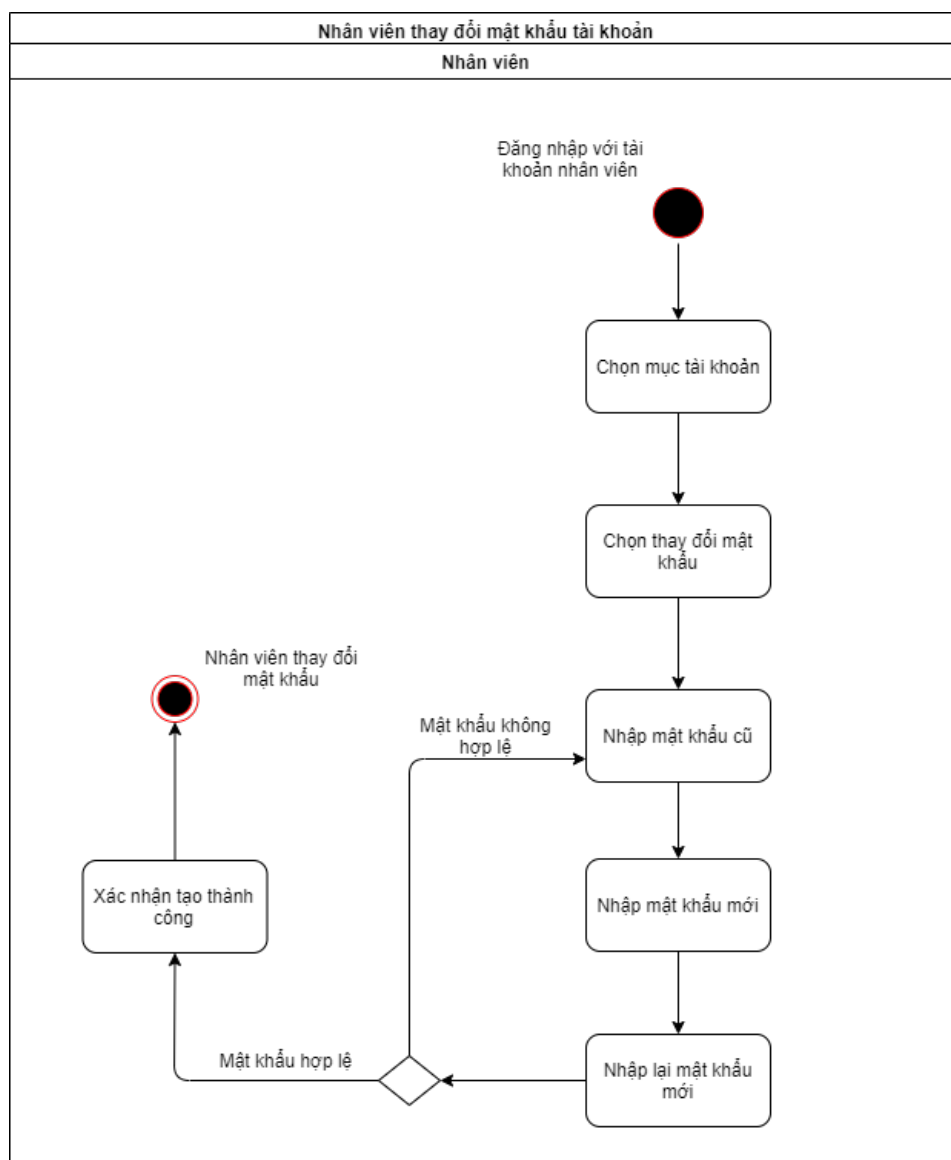
Hình 2: Activity diagram Quản lý xóa tài khoản nhân viên



Hình 3: Activity diagram Quản lý thay đổi thông tin khoản nhân viên

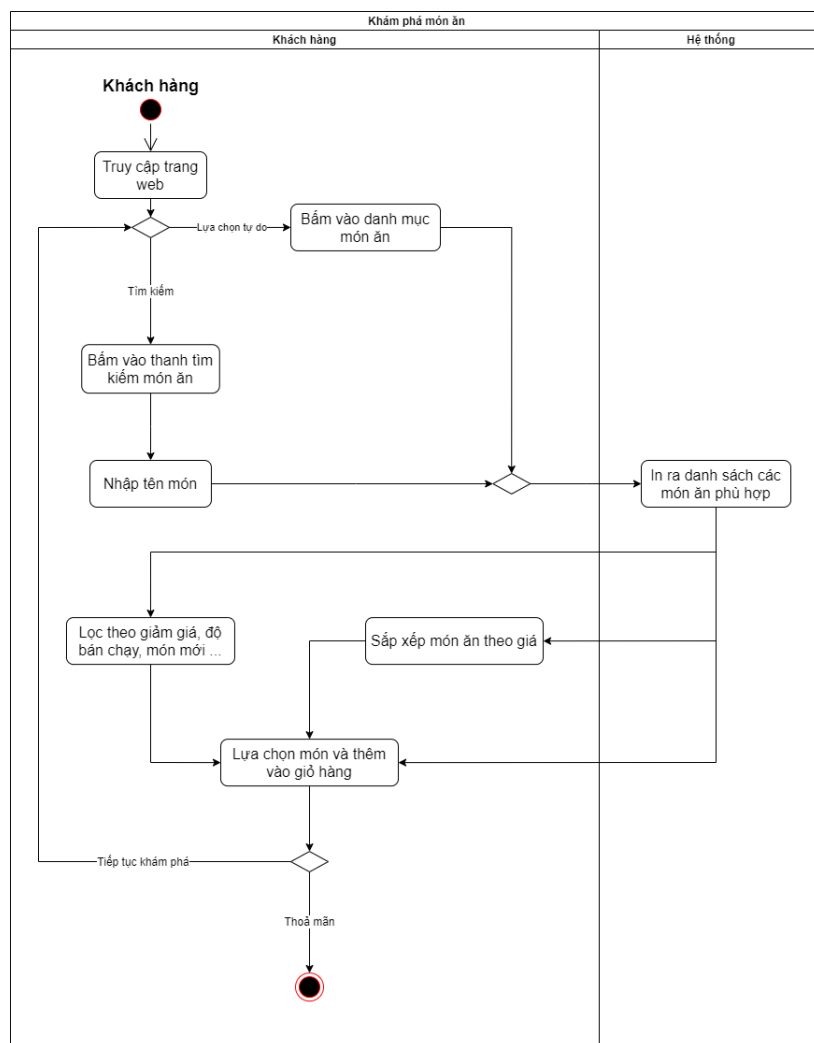


Hình 4: Activity diagram Quản lý xem tài khoản nhân viên



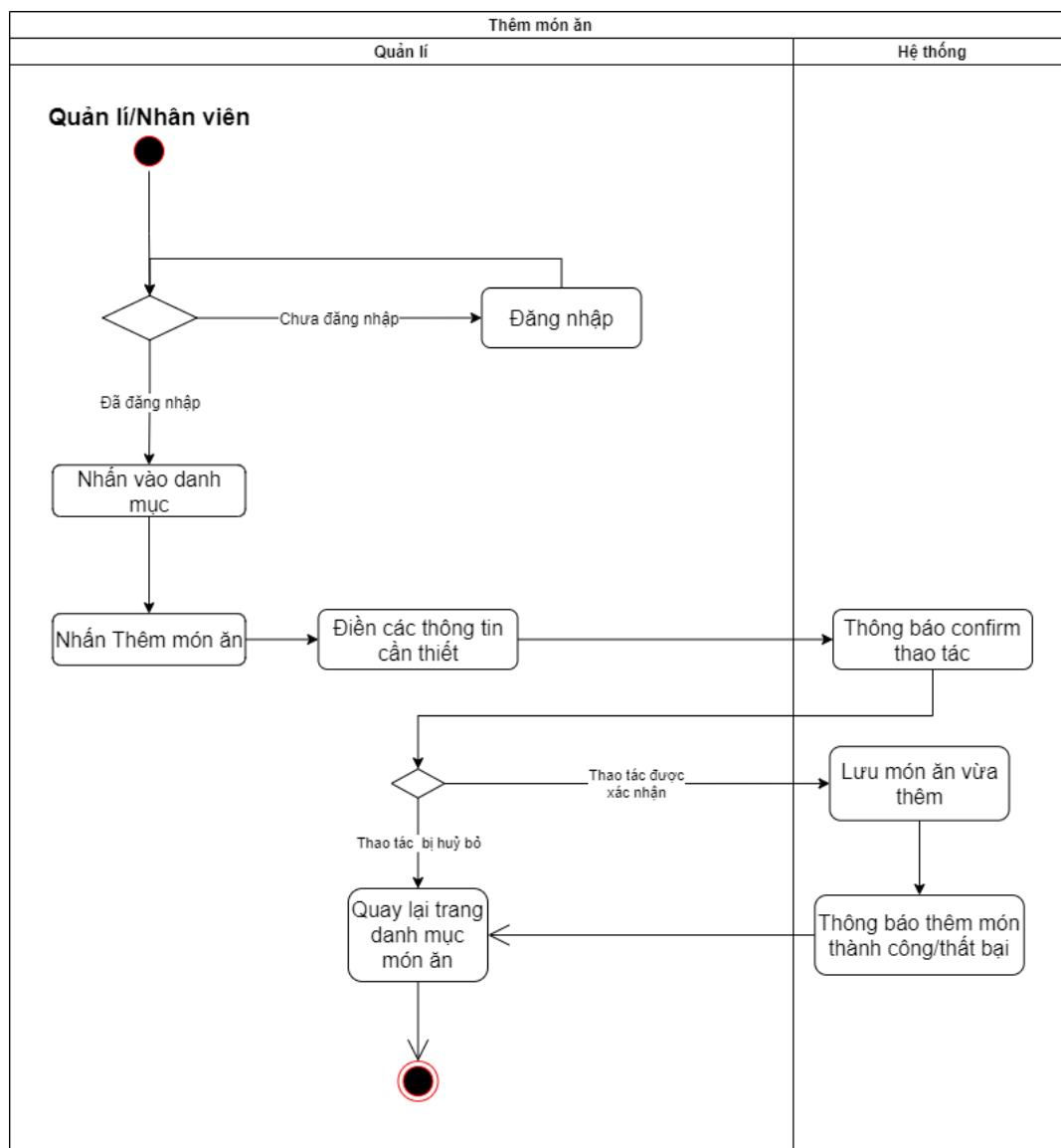
Hình 5: Activity diagram Nhân viên thay đổi mật khẩu tài khoản

1.2 Khám phá món ăn

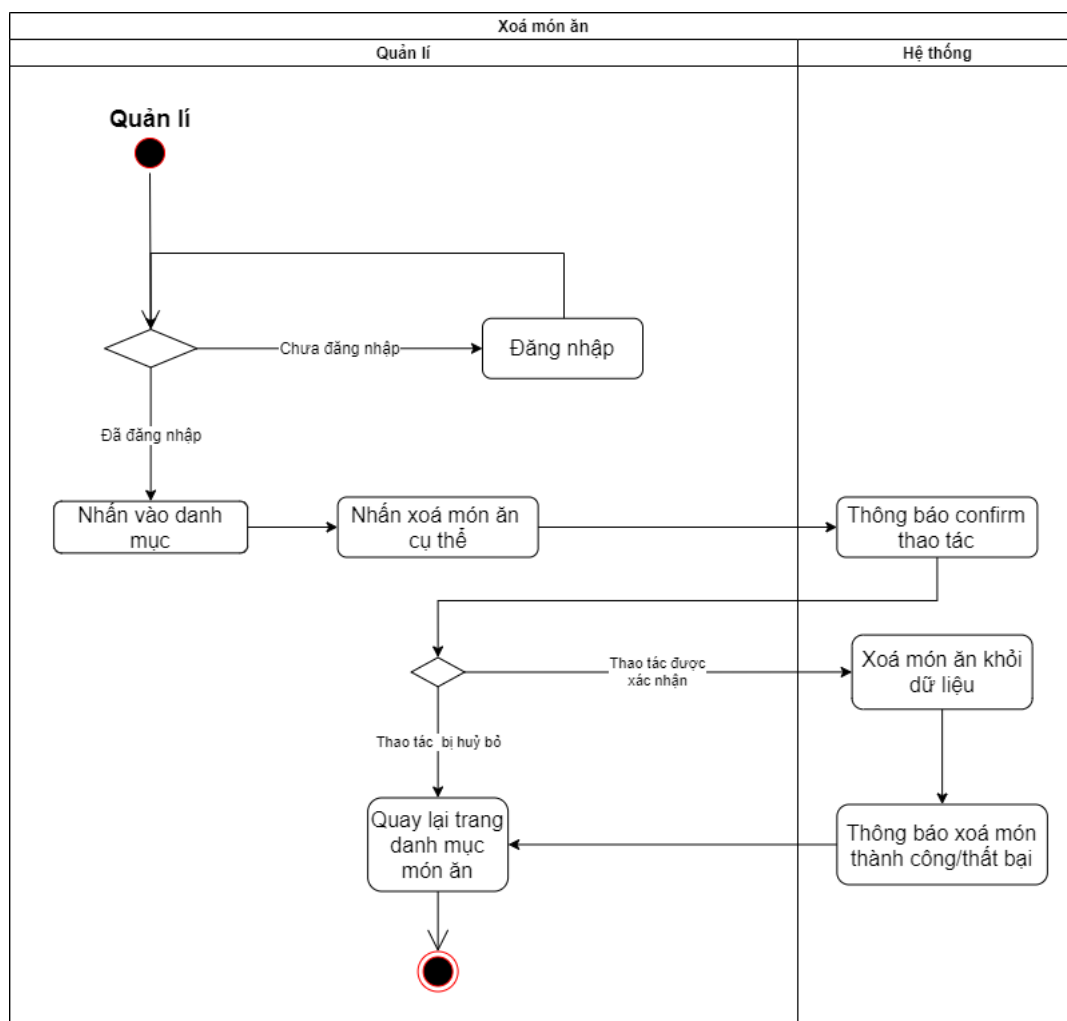


Hình 6: Activity diagram Khách hàng khám phá món ăn

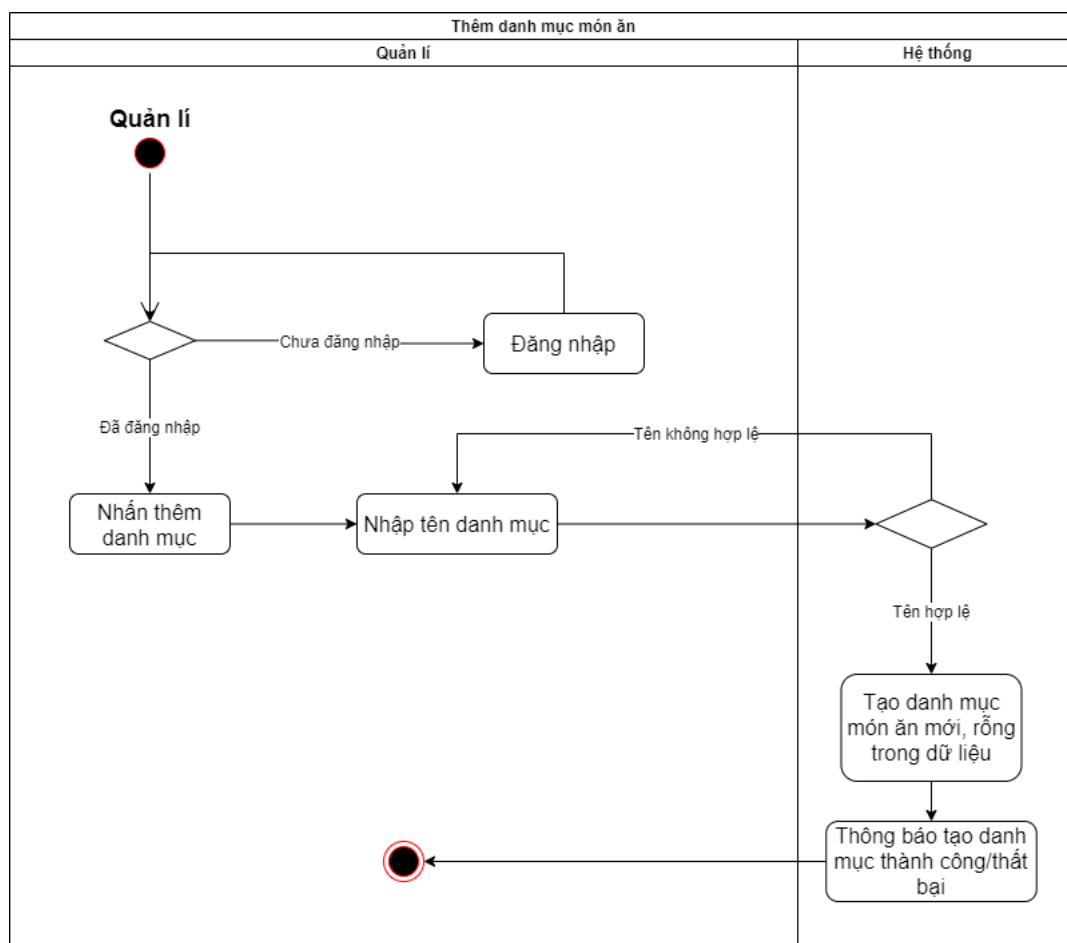
1.3 Quản lý món ăn



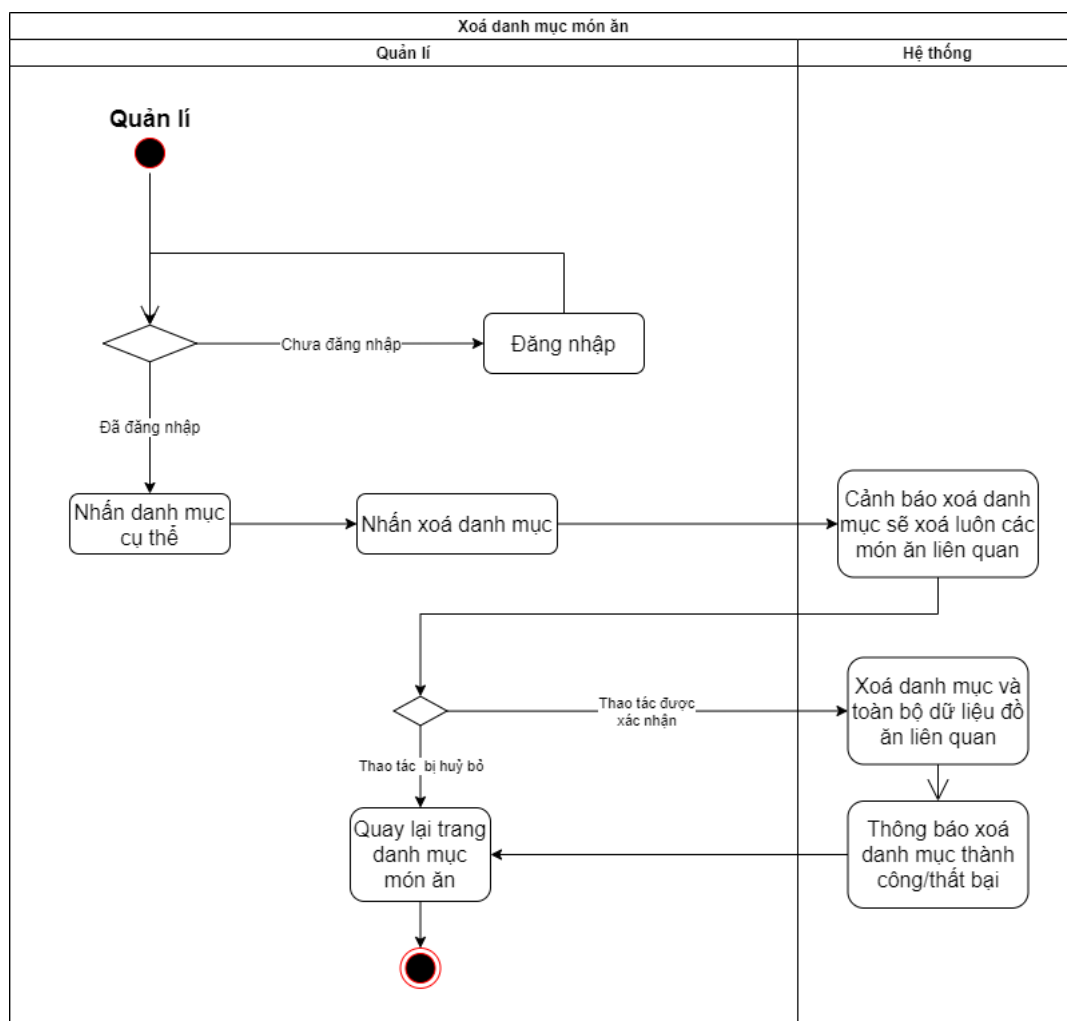
Hình 7: Activity diagram Quản lý thêm món ăn



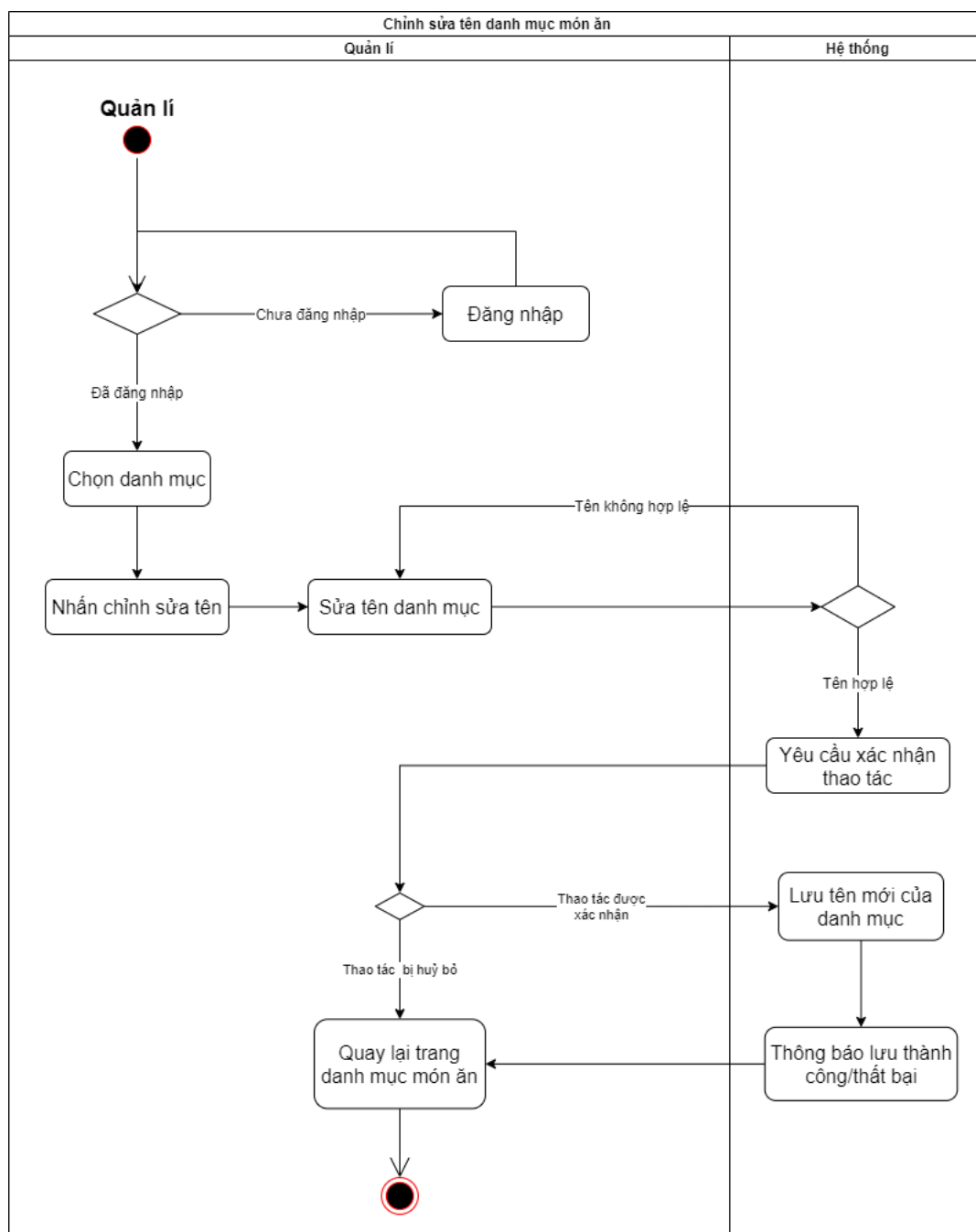
Hình 8: Activity diagram Quản lý xóa món ăn



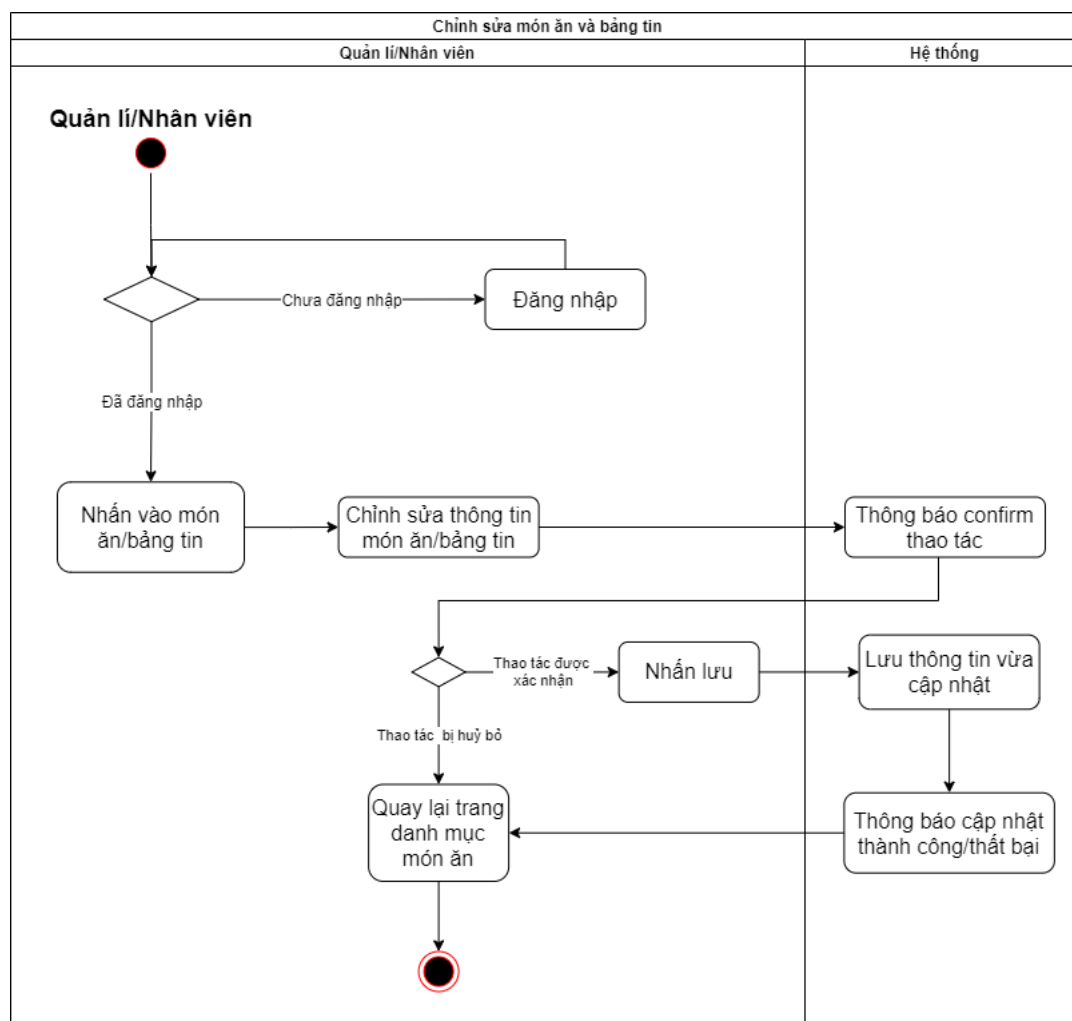
Hình 9: Activity diagram Quản lý thêm danh mục các món ăn



Hình 10: Activity diagram Quản lý xoá danh mục các món ăn



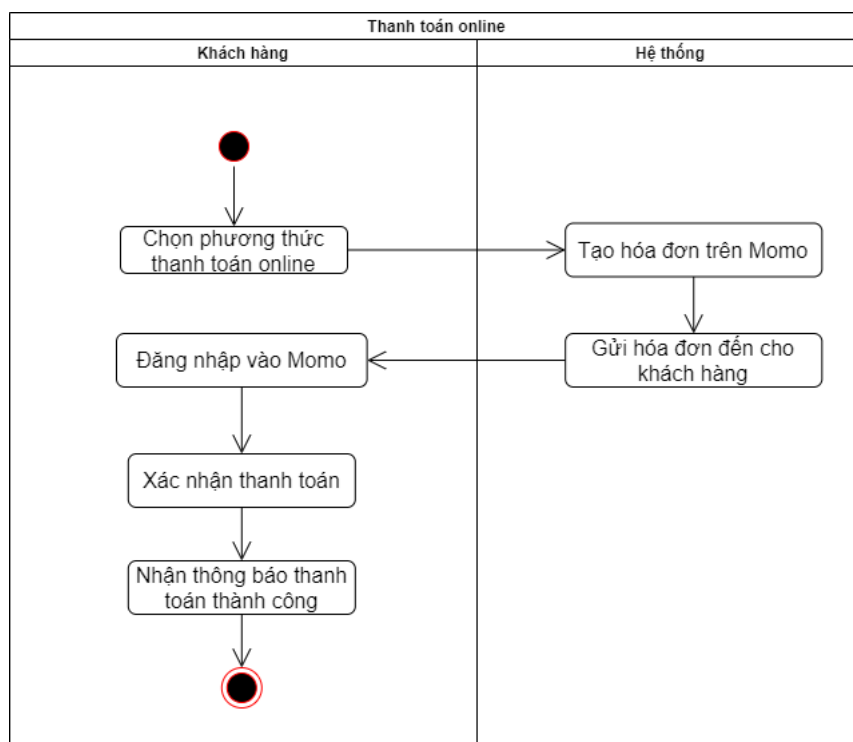
Hình 11: Activity diagram Quản lý chỉnh sửa tên danh mục các món ăn



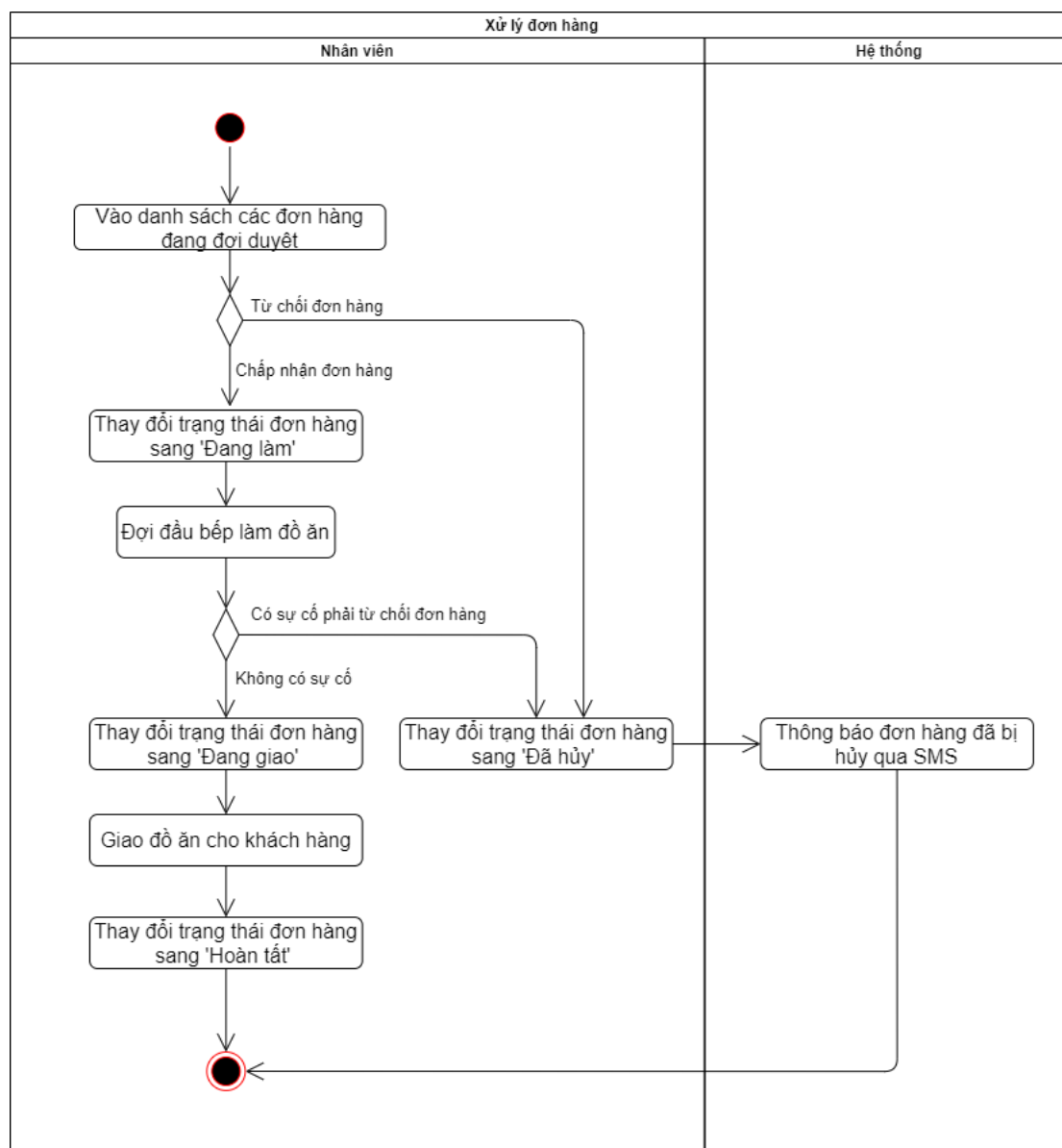
Hình 12: Activity diagram Quản lý/Nhân viên chỉnh sửa món ăn và bảng tin

1.4 Đặt đồ ăn và thanh toán

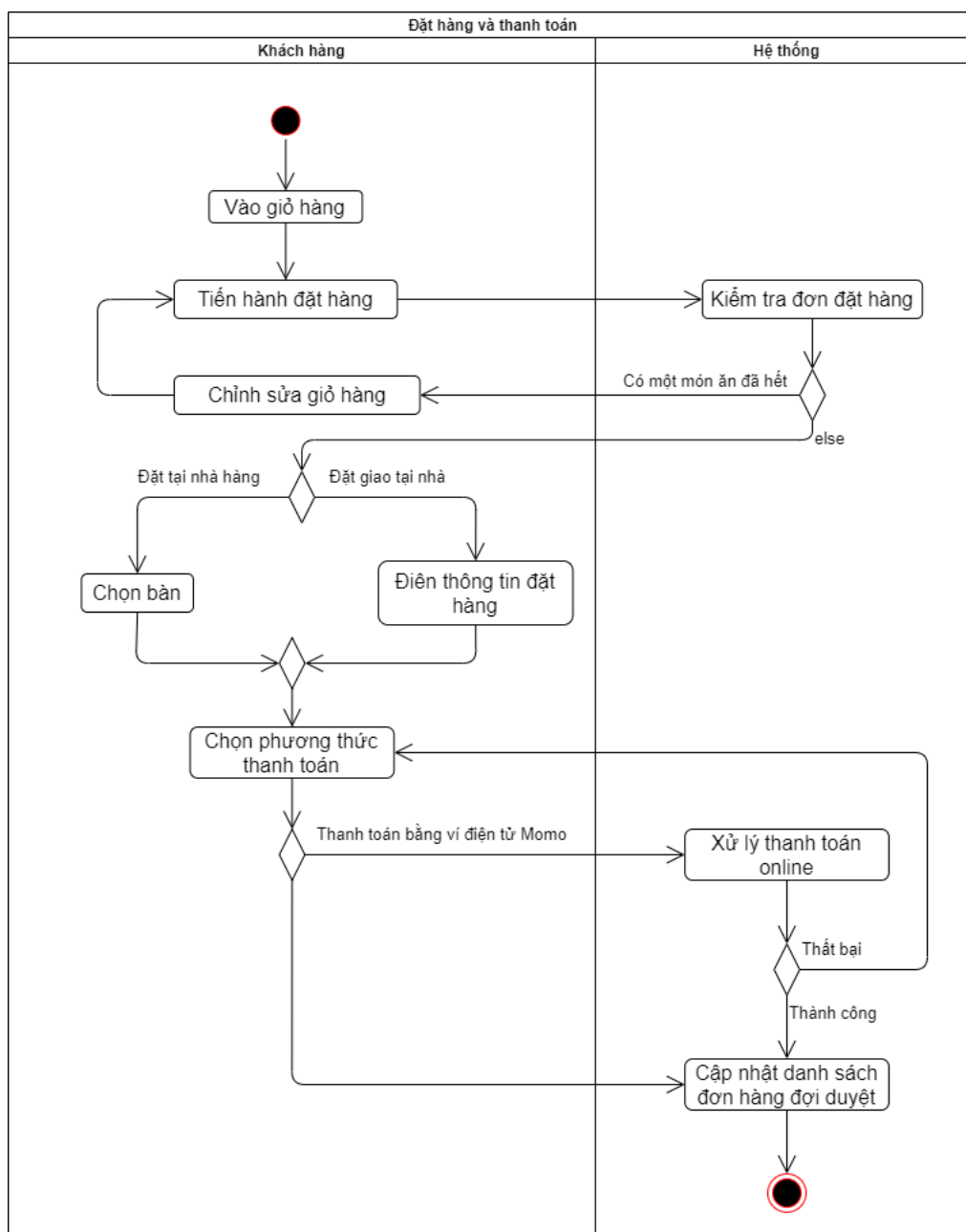
Dưới đây là các activity diagram tương ứng với năm functional requirements:



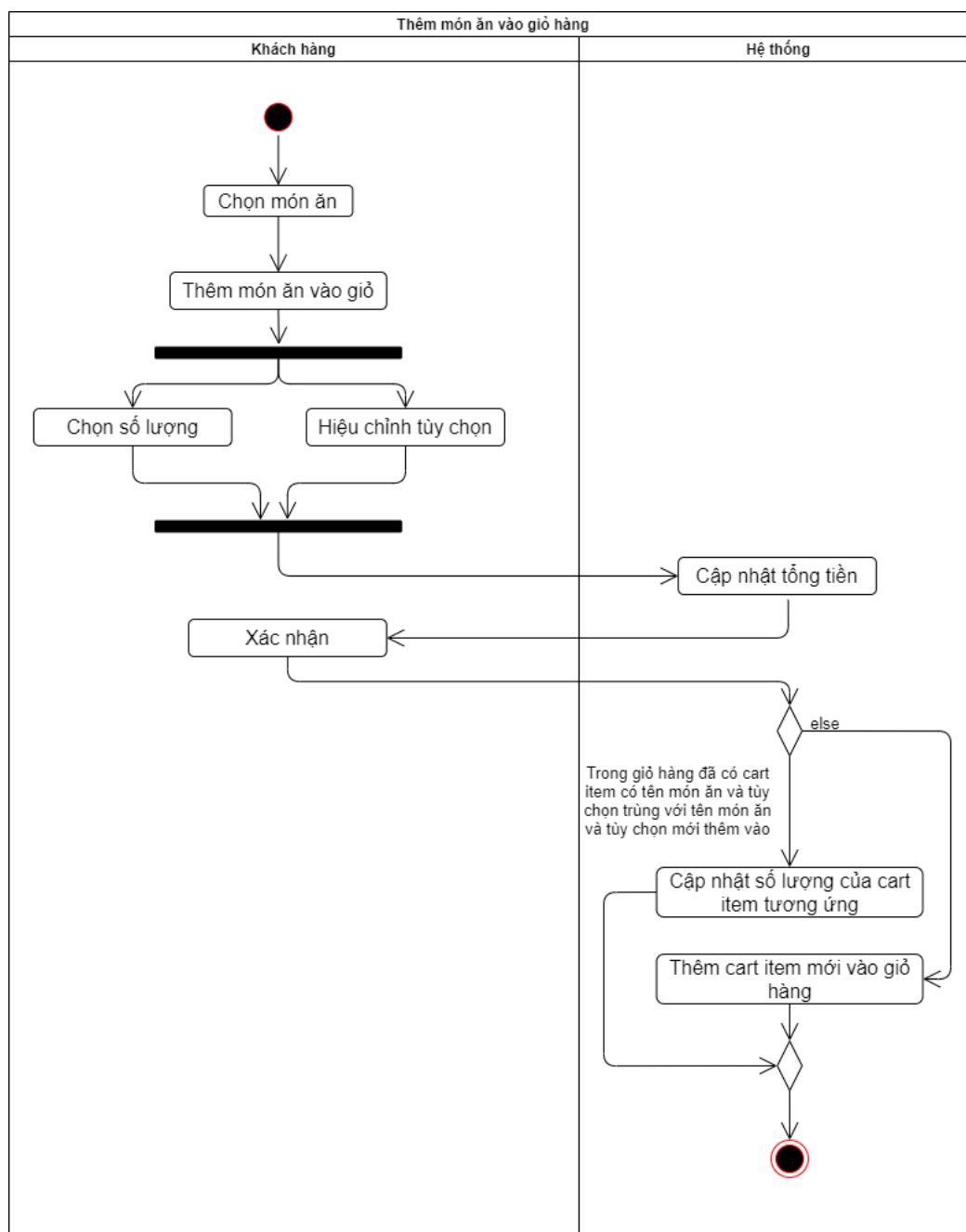
Hình 13: Activity diagram Thanh toán online



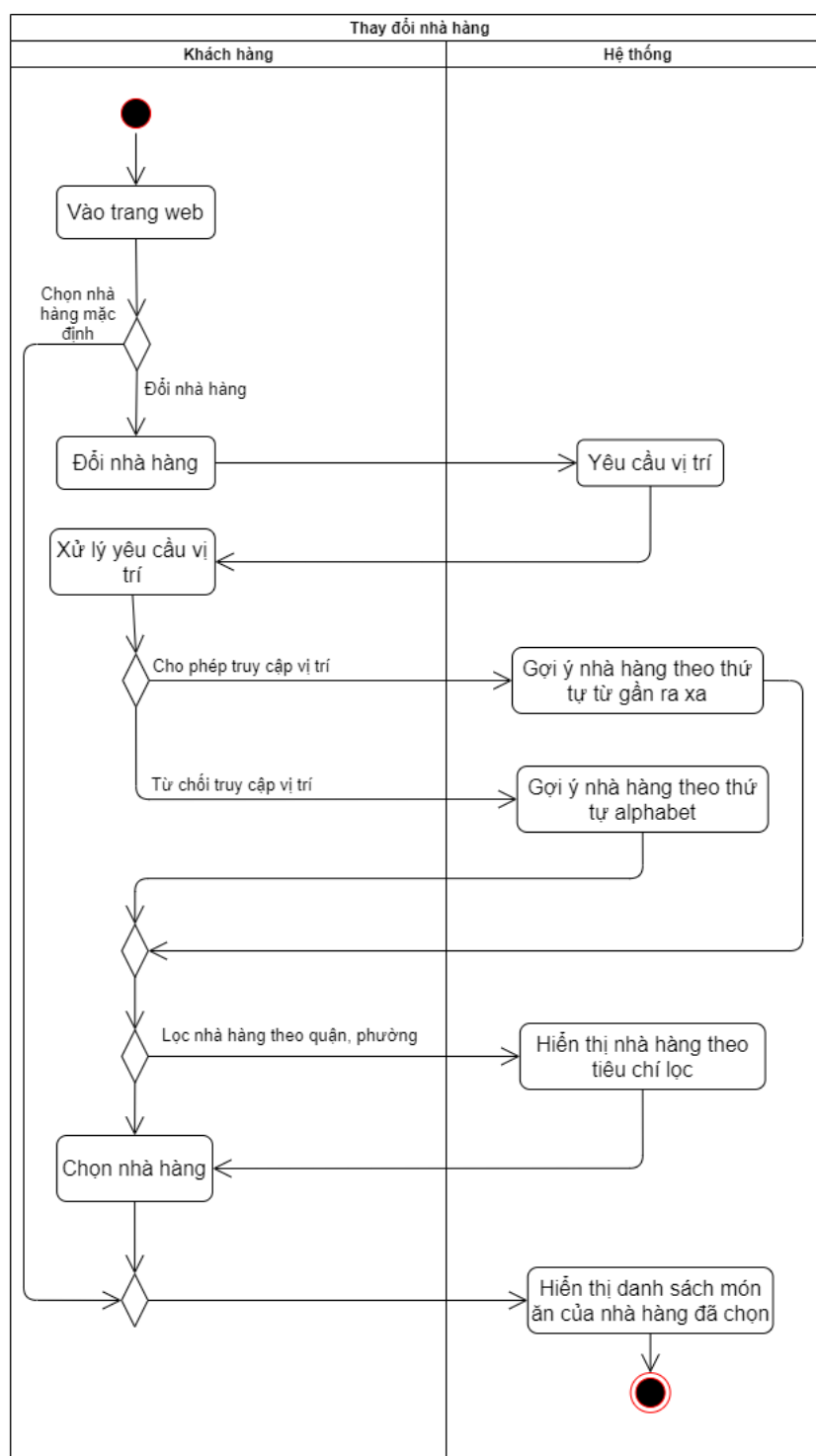
Hình 14: Activity diagram Xử lý đơn hàng



Hình 15: Activity diagram Đặt hàng và thanh toán



Hình 16: Activity diagram Thêm món ăn vào giỏ hàng



Hình 17: Activity diagram Thay đổi nhà hàng

2 Sequence diagram

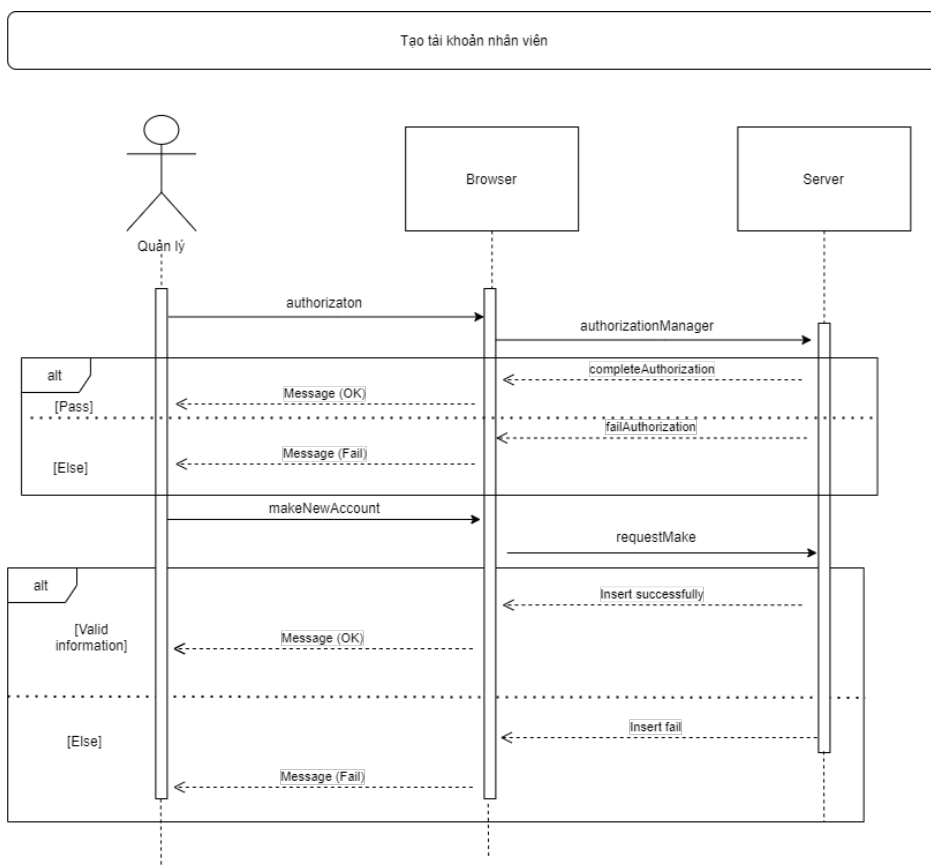
Sequence diagram là loại biểu đồ tương tác phổ biến, nó tập trung vào sự trao đổi thông điệp giữa một số huyết mạch lifeline, trong phạm vi bài tập lớn lần này là actor và các đối tượng (object).

Sequence diagram mô tả một tương tác bằng cách tập trung vào chuỗi các thông điệp được trao đổi, cùng với các đặc tả xảy ra tương ứng của chúng trên các actor và các đối tượng.

Dưới đây là tất cả các sequence diagram cho các feature mà task 1.3 nhóm đã vẽ use-case.

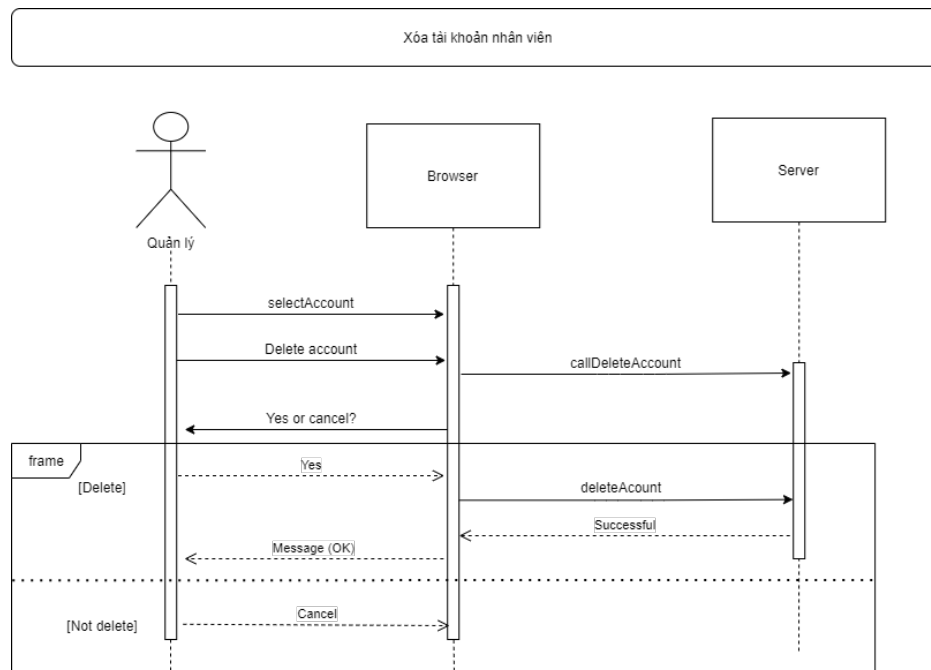
2.1 Quản lý tài khoản

Use-case Tạo tài khoản nhân viên



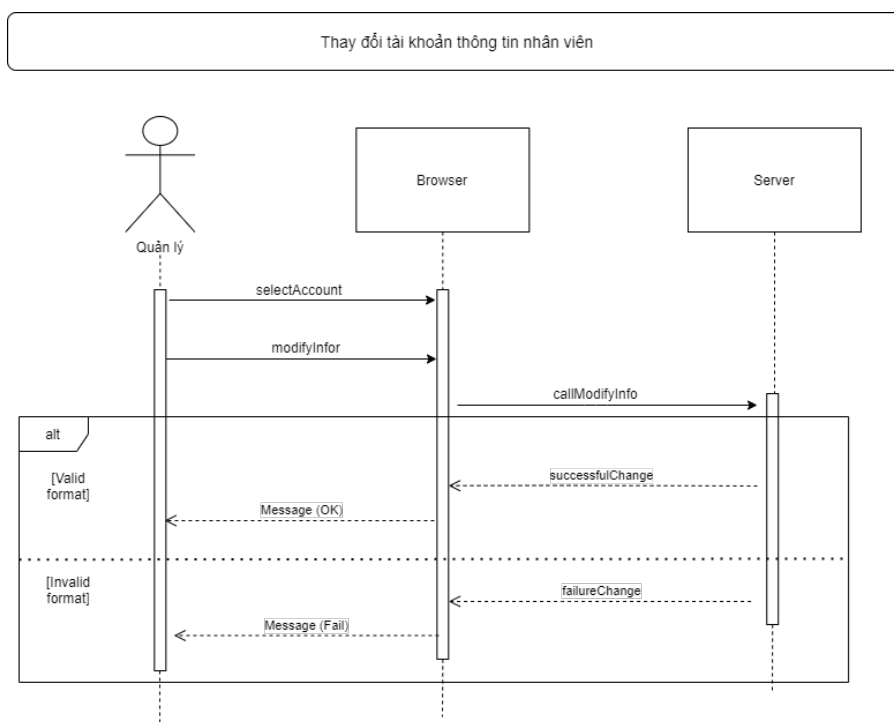
Hình 18: Sequence diagram Quản lý tạo tài khoản nhân viên

Use-case Xóa tài khoản nhân viên



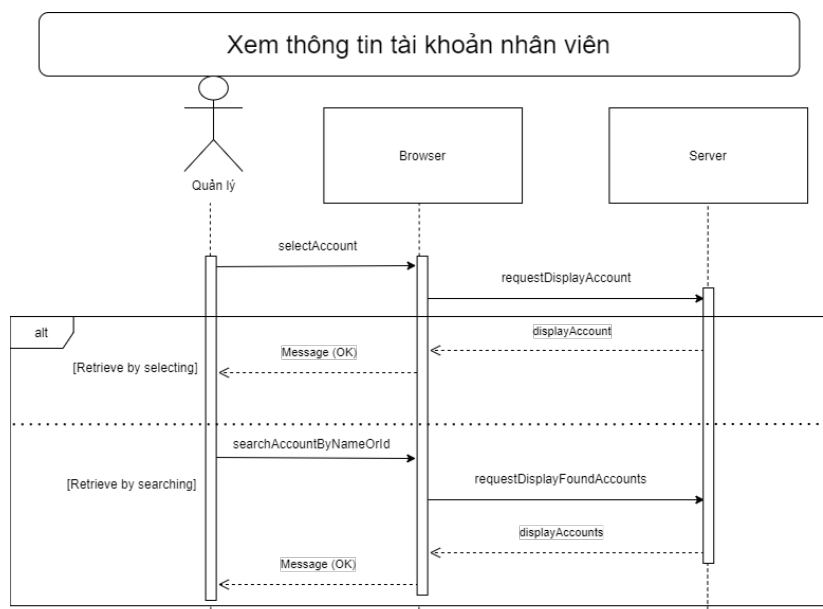
Hình 19: Sequence diagram Quản lý xóa tài khoản nhân viên

Use-case Thay đổi thông tin tài khoản nhân viên



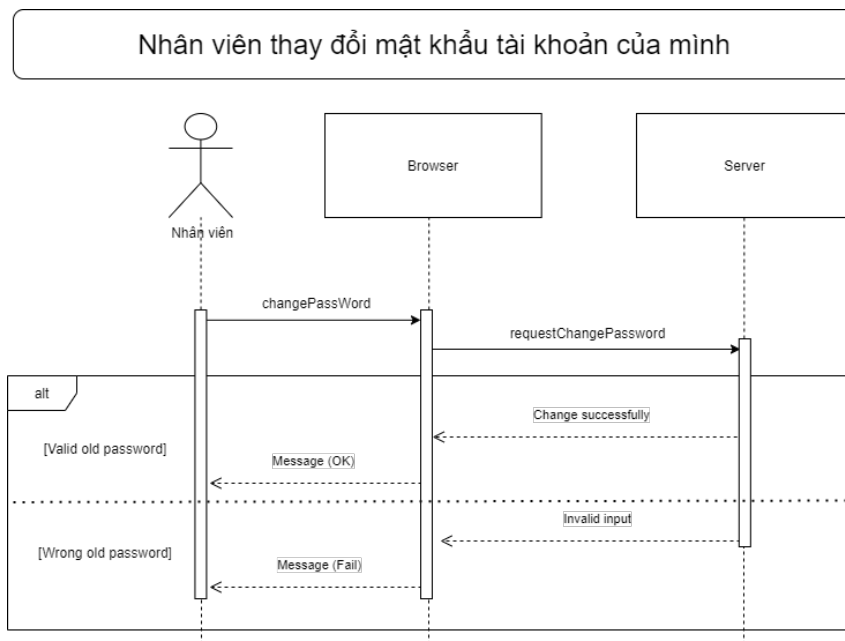
Hình 20: Sequence diagram Quản lý thay đổi thông tin tài khoản nhân viên

Use-case Xem thông tin tài khoản nhân viên



Hình 21: Sequence diagram Quản lý xem thông tin tài khoản nhân viên

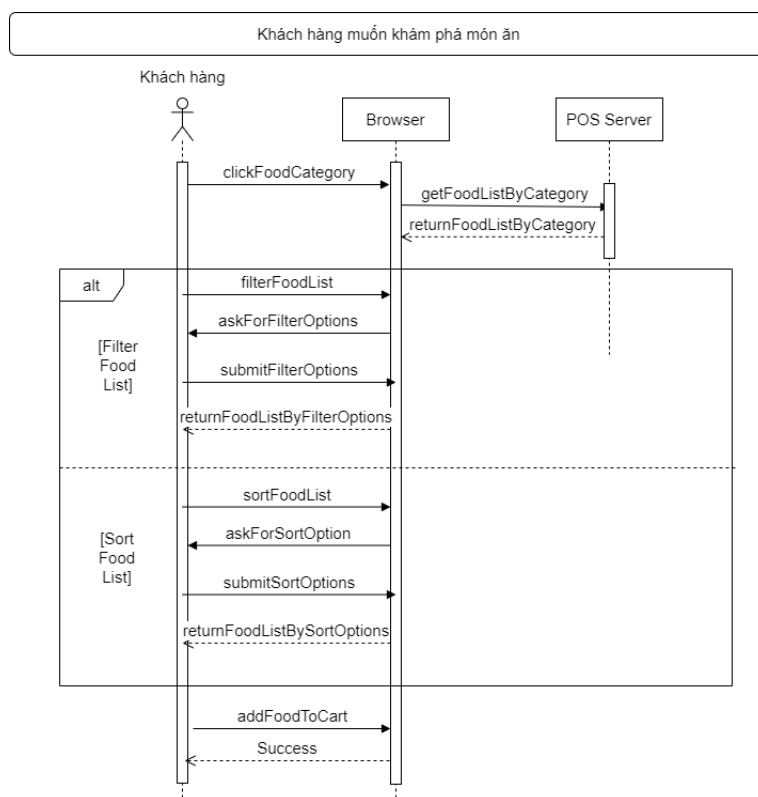
Use-case Thay đổi mật khẩu tài khoản



Hình 22: Sequence diagram Nhân viên thay đổi mật khẩu tài khoản

2.2 Khám phá món ăn

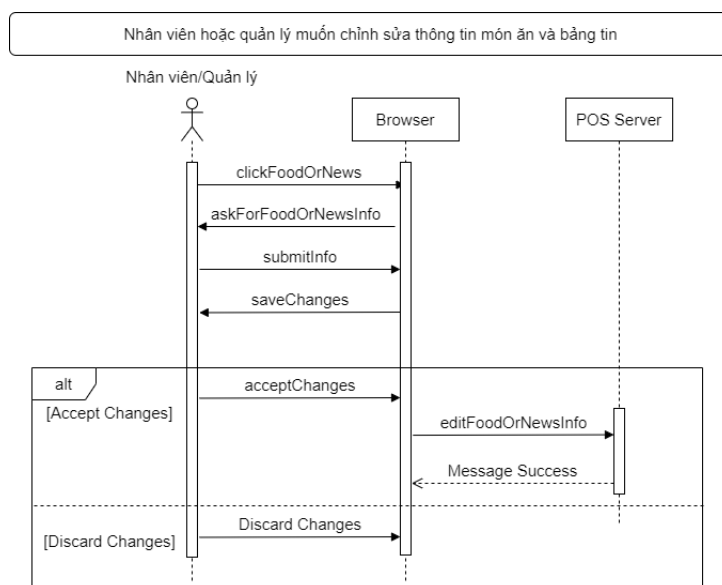
Use-case Khám phá món ăn



Hình 23: Sequence diagram Khách hàng khám phá món ăn

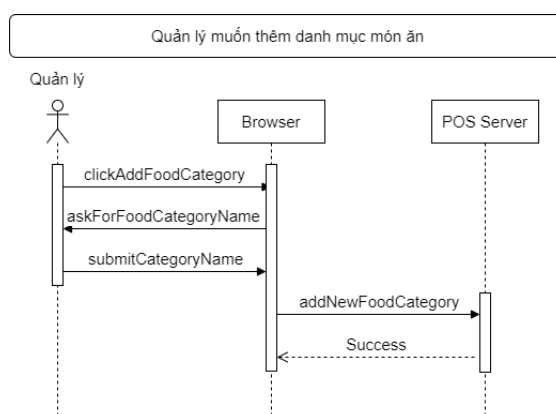
2.3 Quản lý món ăn

Use-case chỉnh sửa thông tin món ăn và bảng tin



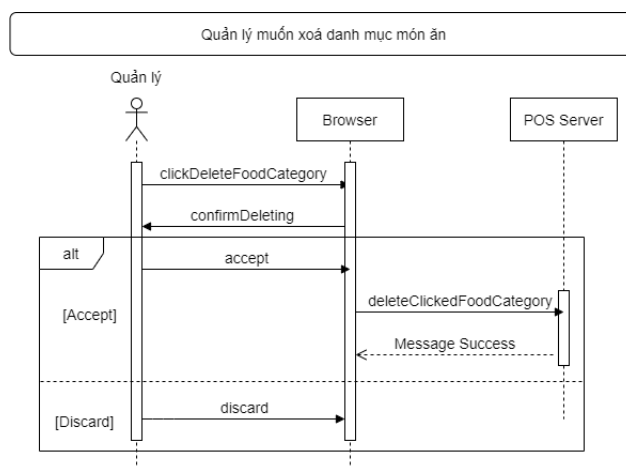
Hình 24: Sequence diagram Quản lý hoặc nhân viên chỉnh sửa thông tin món ăn và bảng tin

Use-case Thêm danh mục món ăn



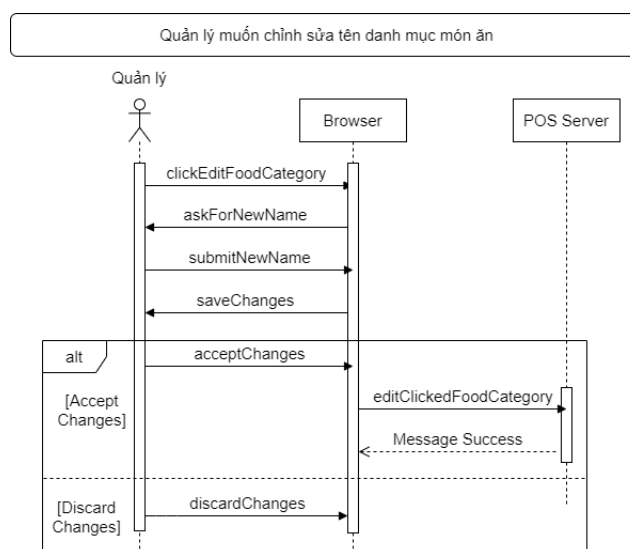
Hình 25: Sequence diagram Quản lý thêm danh mục món ăn

Use-case Xóa danh mục món ăn



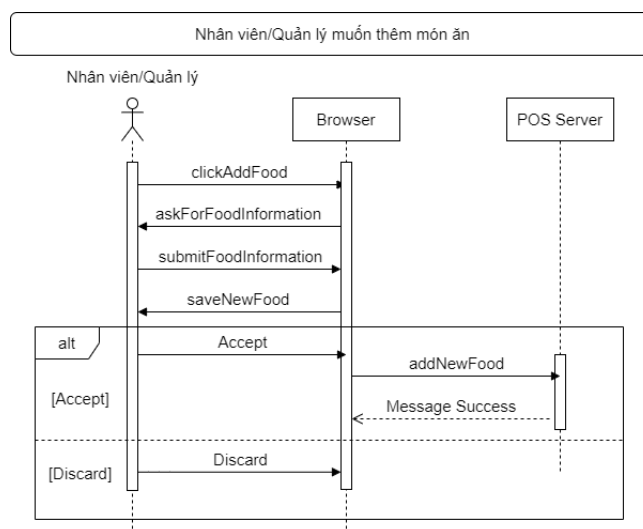
Hình 26: Sequence diagram Quản lý xóa danh mục món ăn

Use-case **Chỉnh sửa danh mục món ăn**



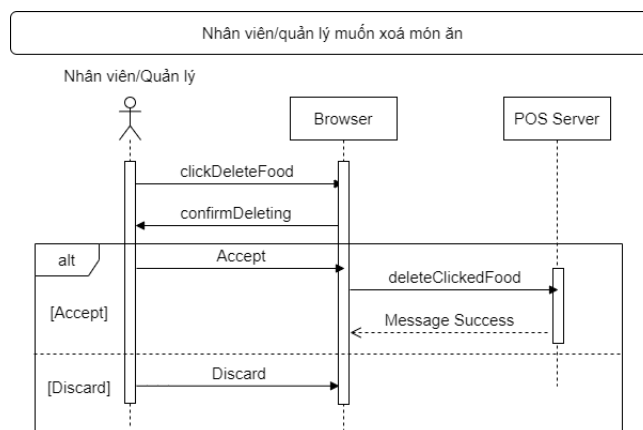
Hình 27: Sequence diagram Quản lý chỉnh sửa danh mục món ăn

Use-case **Thêm món ăn**



Hình 28: Sequence diagram Quản lý thêm món ăn

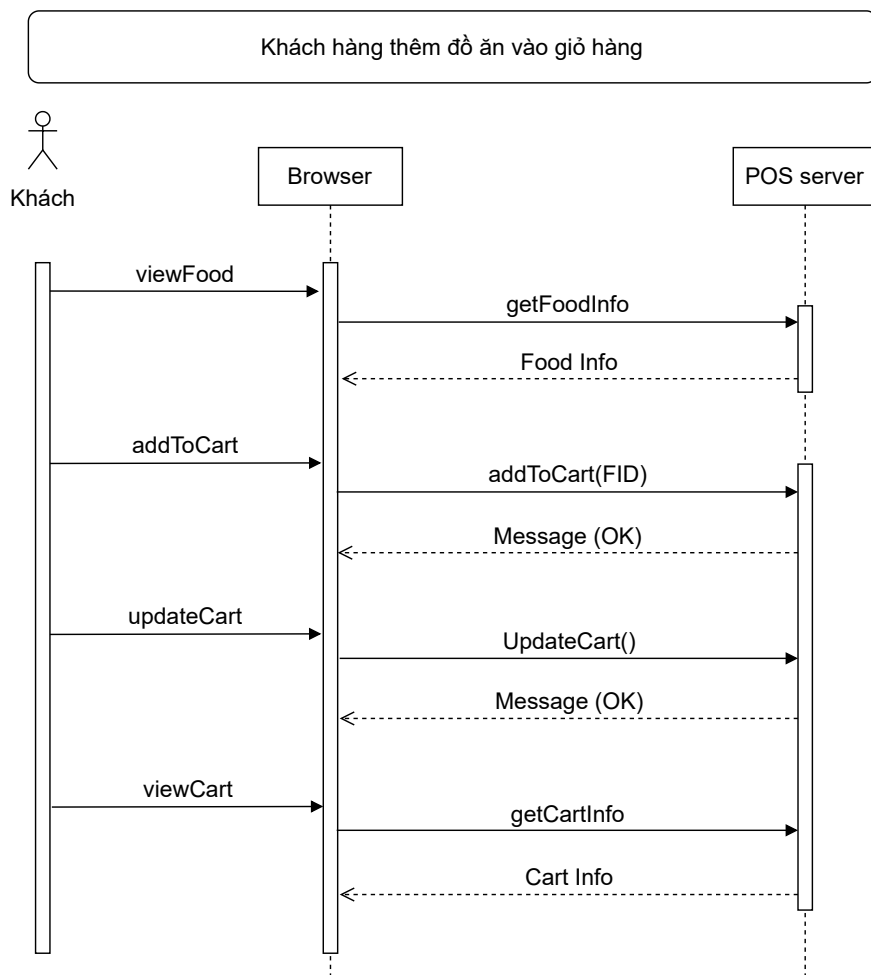
Use-case Xóa món ăn



Hình 29: Sequence diagram Quản lý xóa món ăn

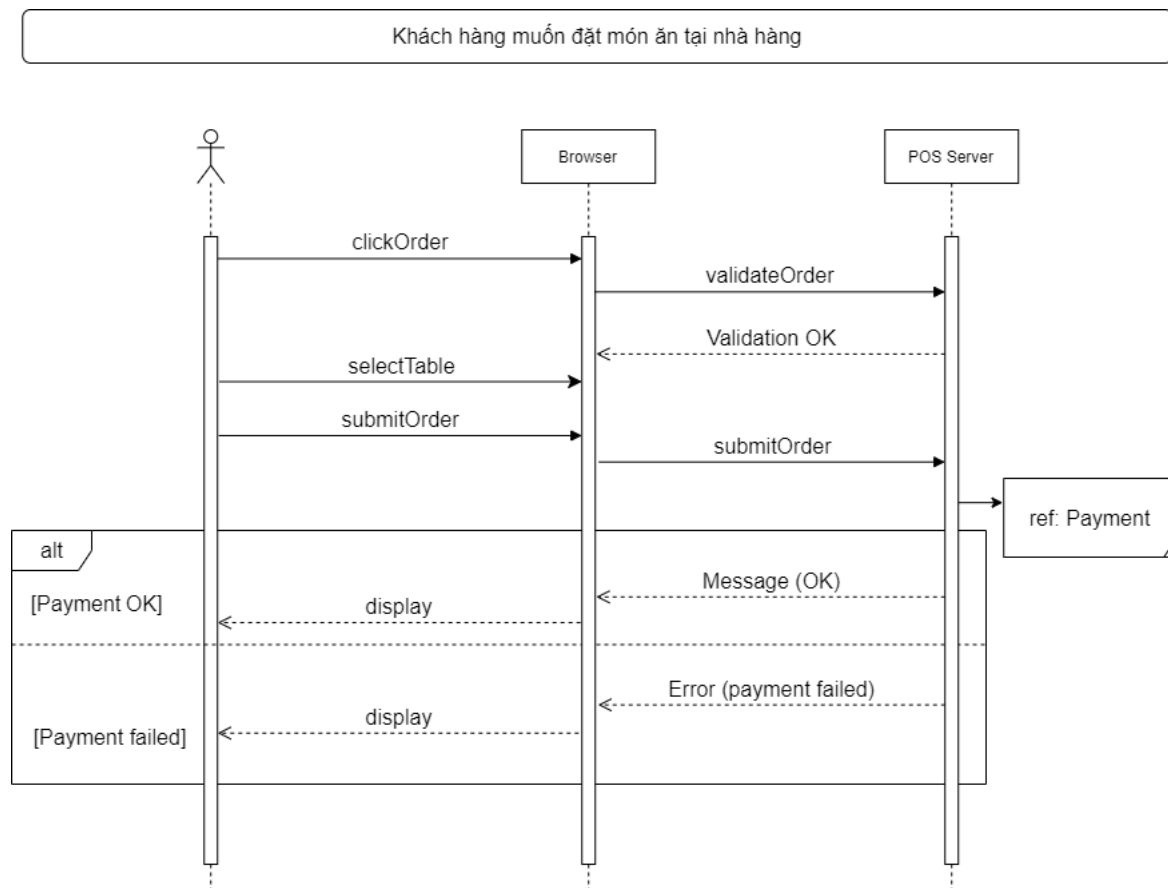
2.4 Đặt đồ ăn và thanh toán online

Use-case Thêm đồ ăn vào giỏ hàng



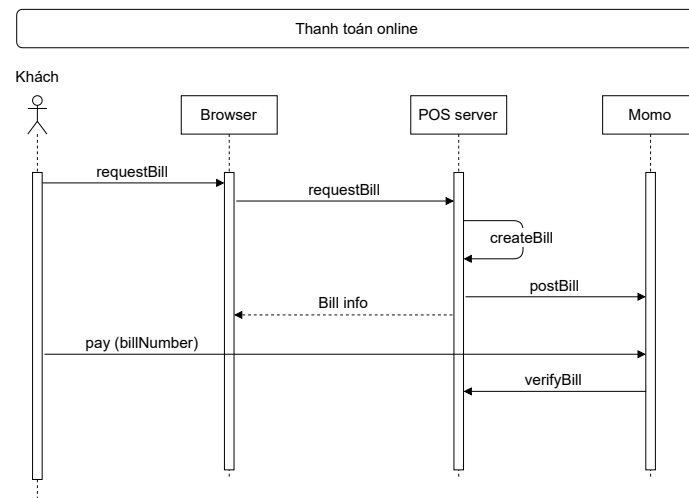
Hình 30: Sequence diagram Thêm đồ ăn vào giỏ hàng

Use-case Đặt đồ ăn tại nhà hàng



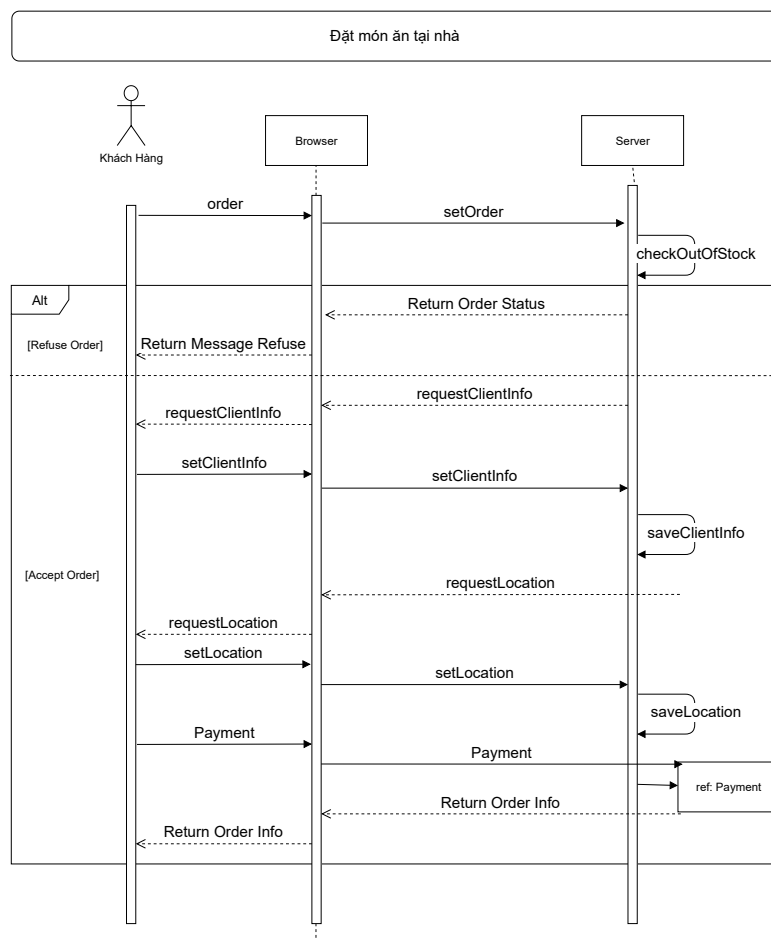
Hình 31: Sequence diagram Đặt đồ ăn tại nhà hàng

Use-case Thanh toán online



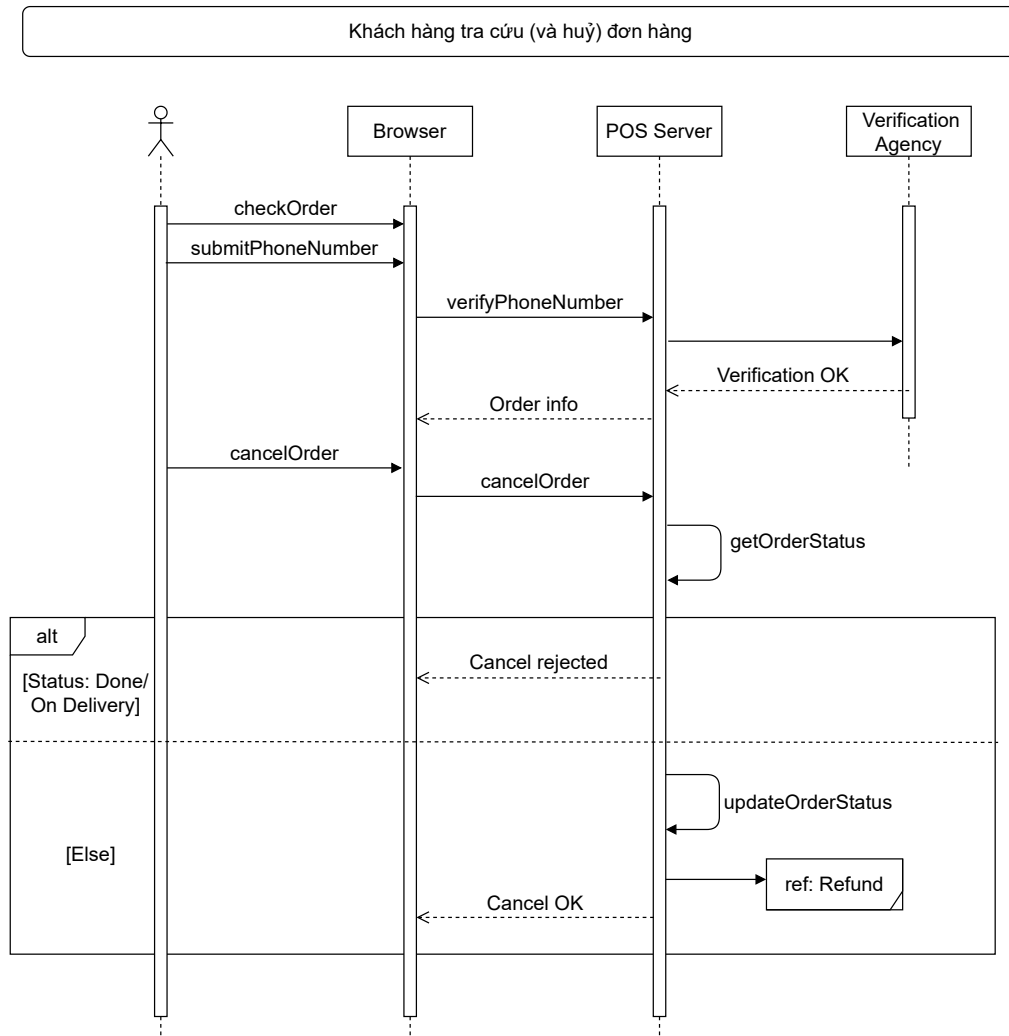
Hình 32: Sequence diagram Thanh toán online

Use-case Đặt đồ ăn tại nhà



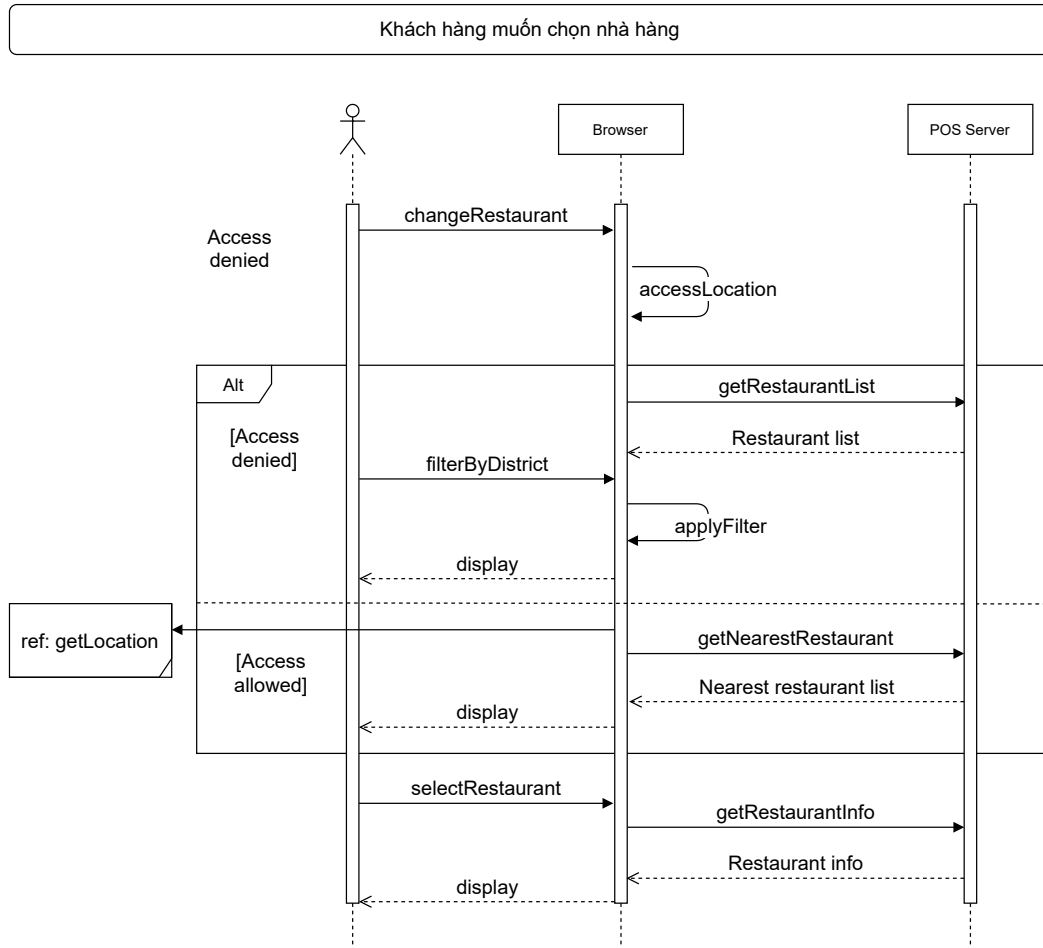
Hình 33: Sequence diagram Đặt đồ ăn tại nhà

Use-case Tra cứu và huỷ đơn đặt hàng



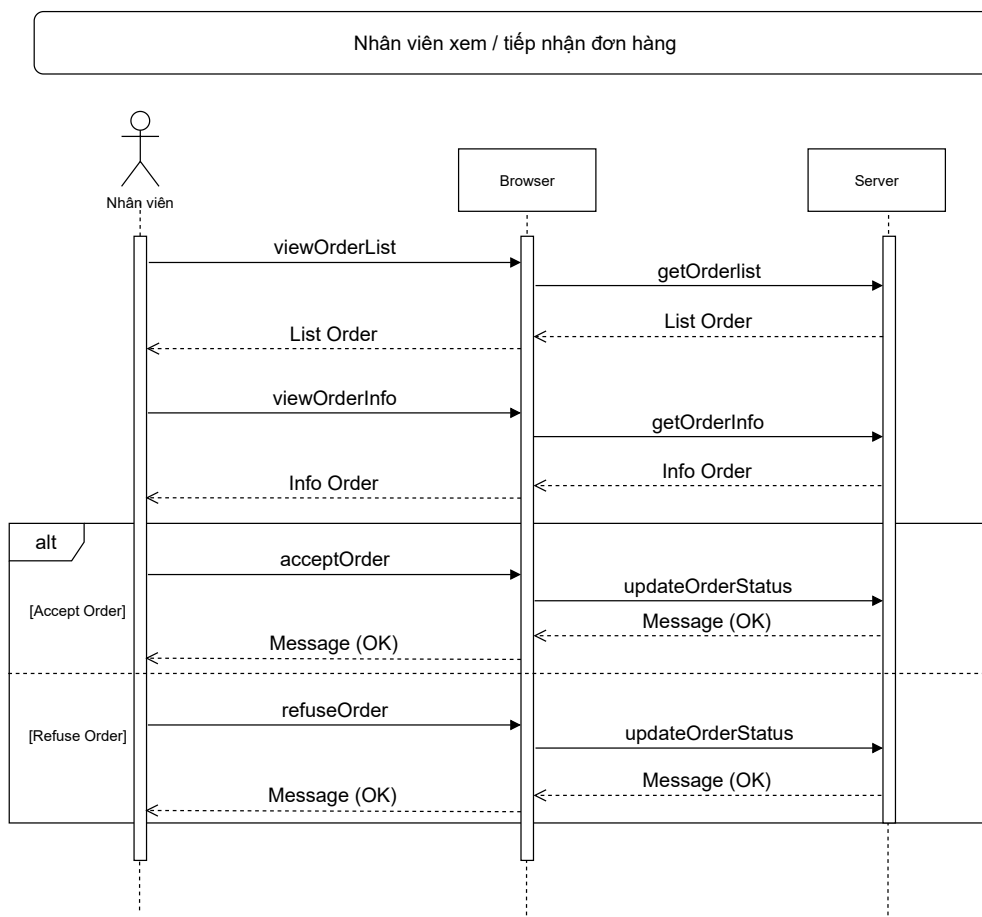
Hình 34: Sequence diagram Tra cứu và huỷ đơn đặt hàng

Use-case Chọn nhà hàng



Hình 35: Sequence diagram Chọn nhà hàng

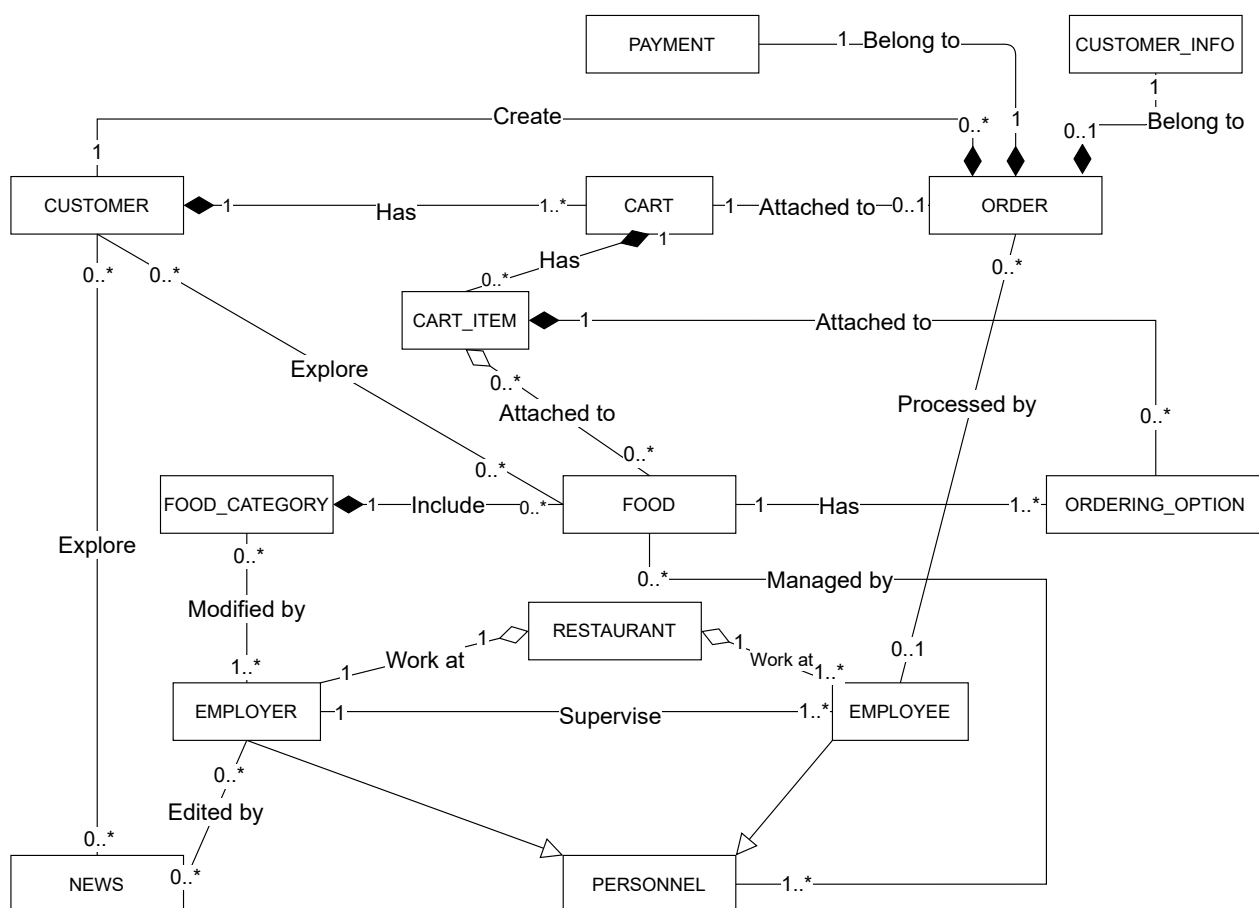
Use-case Nhân viên duyệt đơn hàng



Hình 36: Sequence diagram Nhân viên duyệt đơn hàng

3 Class diagram

Trong task 2: Mô hình hóa hệ thống, nhóm sẽ vẽ class diagram cho toàn hệ thống ở mức độ ý niệm. Các class diagram phục vụ cho việc thực thi và triển khai code sẽ được biểu diễn và mô tả ở task 3, cụ thể phần 3.2. Class diagram của toàn hệ thống được thể hiện ở hình 37.



Hình 37: Class diagram của toàn hệ thống ở mức độ ý niệm

Mô tả class diagram ở hình 37:

1. Nhân sự (PERSONNEL) của một nhà hàng (RESTAURANT) được chia thành hai vai trò là quản lý (EMPLOYER) và nhân viên (EMPLOYEE). Mỗi nhà hàng chỉ có duy nhất một quản lý và có thể có nhiều nhân viên làm việc chung trong một nhà hàng. Ngoài ra, một quản lý có thể quản lý nhiều nhân viên.
2. Quản lý và nhân viên đều có thể chỉnh sửa và cập nhật thông tin các món ăn (FOOD). Mỗi món ăn đều thuộc về một danh mục (FOOD_CATEGORY) nào đó và chỉ người quản lý mới chỉnh sửa được danh mục của các món ăn.
3. Trang web sẽ có các bản tin (NEWS) để quảng cáo và đưa các thông tin về chương trình khuyến mãi của nhà hàng. Và các thông tin trên các bản tin chỉ có thể được chỉnh sửa bởi người quản lý.

4. Mỗi khách hàng sẽ có nhiều giỏ hàng (CART) để thêm món ăn muốn đặt hàng vào. Một giỏ hàng có thể chứa nhiều mặt hàng (CART_ITEM). Mỗi mặt hàng sẽ ứng với một món ăn, và lựa chọn (ORDERING_OPTION) đi kèm với món ăn đó. Ví dụ đặt cùng món Pizza nhưng có thể có hai mặt hàng: một mặt hàng là một bánh pizza đế dày và một mặt hàng là hai bánh pizza đế mỏng.
5. Mỗi khách hàng sẽ có nhiều đơn đặt hàng. Mỗi đơn đặt hàng sẽ bao gồm thông tin thanh toán (PAYMENT), thông tin khách hàng (CUSTOMER_INFO) và giỏ hàng tương ứng với đơn hàng đó.
6. Chỉ có nhân viên mới xử lý được đơn đặt hàng.

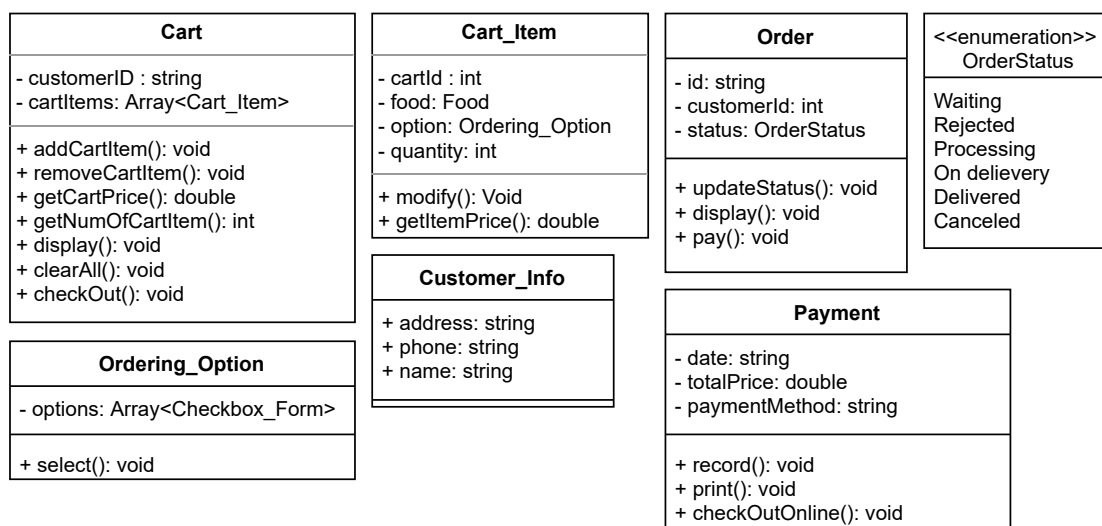
Như đã thấy ở hình 37, nhằm đảm bảo tính gọn gàng và dễ đọc, các classes sẽ không có các thuộc tính (attributes) cũng như các phương thức (methods). Thay vào đó các thuộc tính và phương thức của các classes sẽ được thể hiện chi tiết ở các hình 38, 39 và 40.

| FOOD | FOOD_CATEGORY | NEWS | Restaurant |
|--|--|---|---|
| <ul style="list-style-type: none"> - name: string - image: file - price: int - isOutOfStock: bool - category: Food_Category | <ul style="list-style-type: none"> - name: string - foods: Array<Food> | <ul style="list-style-type: none"> - name: string - content: file | <ul style="list-style-type: none"> - ID : string - address: string - numOfEmployee: int |
| <ul style="list-style-type: none"> + display(): void + modify(): void + isOutOfStock(): bool | <ul style="list-style-type: none"> + display(): void + addFood(): void + removeFood(): void + getListFood(): Array of Food | <ul style="list-style-type: none"> + display(): void + edit(): void | <ul style="list-style-type: none"> + getNumOfEmployee(): int + isOpen() : bool + getFoodsStatus(): Array<bool> + getAddress(): string |

Hình 38: Các thuộc tính và phương thức của các class Food, Food_Category, News và Restaurant

| Personnel | Employee | Employer | Customer |
|--|--|---|---|
| <ul style="list-style-type: none"> - ID: string - name: string - dOb: string - user: string - password: string - address: string - phone: string - Email: string - workRestaurant: Restaurant | <ul style="list-style-type: none"> - ID : string - Salary: double - Shift: Enum { 'Morning' , 'Afternoon' , 'Evening' } - Branch: string | <ul style="list-style-type: none"> + makeAccount(): void + deleteAccount(): void + changeInfoAccount(): void | <ul style="list-style-type: none"> - id: string - cart: Cart |
| <ul style="list-style-type: none"> + editNews(): void + processOrder(): void + updateFoodStatus(): void | <ul style="list-style-type: none"> + changePassword(): void + changeShift(): void | | <ul style="list-style-type: none"> + addToCart(): void + viewFood() : Food + updateCart: void + viewCart() : cart + selectRestaurant() : void + order() : void + pay(): void + setInfo() : void + setLocation() : void |

Hình 39: Các thuộc tính và phương thức của các class Personnel, Employer, Employee và Customer



Hình 40: Các thuộc tính và phương thức của các class Cart, Cart_Item, Order, Customer_Info, Ordering_Option và Payment