

ĐẠI HỌC BÁCH KHOA THÀNH PHỐ HỒ CHÍ MINH  
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



Bài tập lớn môn Công nghệ phần mềm

Dự án:

## **RESTAURANT POS 2.0**

**Giáo viên hướng dẫn:** Quản Thành Thơ

**Lớp:** L02

**Nhóm:** 03

**Danh sách sinh viên thực hiện:**

1. Trương Hồng Hoa - 1911185
2. Lương Thị Quỳnh Hương - 1911314
3. Đặng Trần Hữu Luân - 1914052
4. Đinh Lệ Trân - 1912267
5. Nguyễn Tuấn Vinh - 1915944

# Changelog

## Giai đoạn 1

STT	Ngày	Người thực hiện	Công việc	Ghi chú
1	6/9/2021	Trân	Thành lập nhóm, tạo không gian thảo luận trên mạng xã hội	
		_____	Mỗi thành viên độc lập tìm hiểu đề tài	
2	8/9/2021	_____	Thảo luận về đề tài	
		Trân	Thắc mắc về context và scope	Đã nhận được phản hồi
		_____	Thảo luận về các feature	
3	10/9/2021	_____	Thảo luận và bổ sung feature	Đã thống nhất
4	15/9/2021	_____	Hoàn thành task 1.1	
5	17/9/2021	_____	Thảo luận về yêu cầu chức năng và phi chức năng	Đã thống nhất
6	23/9/2021	Hương	Vẽ usecase cho toàn hệ thống	
		_____	Hoàn thành task 1.2	
		Luân	Vẽ usecase cho food ordering	
		_____	Mô tả table usecase	
		_____	Hoàn thành task 1.3	
		_____	Hoàn thành bài báo cáo	
7	25/9/2021	Hoa	Hoàn thành changelog	

*Các mục người thực hiện để trống xem như toàn bộ nhóm tham gia*

## Giai đoạn 2

STT	Ngày	Người thực hiện	Công việc	Ghi chú
1	30/09/2021	Vinh	Vẽ activity diagram cho usecase “Đặt đơn”	
2	30/09/2021	Trân	Vẽ activity diagram cho usecase “Đặt bàn”	
3	30/09/2021	Hương	Vẽ activity diagram cho usecase “Thanh toán”	
4	30/09/2021	Hoa	Vẽ activity diagram cho usecase “Tư vấn khách hàng”, “Đăng nhập”	
5	02/10/2021	Luân	Vẽ activity diagram cho usecase “Quản lý thực đơn”, “Quản lý người dùng”	
6	07/10/2021	Luân, Vinh	Vẽ sequence diagram cho usecase “Đặt đơn”	
7	08/10/2021	Hương, Hoa	Vẽ class diagram cho usecase “Đặt đơn”	
8	10/10/2021	Trân	Hoàn thành báo cáo	

## Giai đoạn 3

STT	Ngày	Người thực hiện	Công việc	Ghi chú
1	14/10/2021	_____	Tìm hiểu về kiến trúc hệ thống sẽ áp dụng	
2	18/10/2021	_____	Thảo luận chọn mô hình kiến trúc và triển khai công việc	
3	22/10/2021	_____	Hoàn thành task 3.1	
4	23/10/2021	Luân	Vẽ component diagram cho hệ thống, “Quản lý thực đơn”, “Quản lý người dùng”	
5	23/10/2021	Vinh	Vẽ component diagram cho “Đặt đơn”	
6	23/10/2021	Hương	Vẽ component diagram cho “Thanh toán”	
7	23/10/2021	Trân	Vẽ component diagram cho “Đặt bàn”	
8	23/10/2021	Hoa	Vẽ component diagram cho “Đăng nhập”, “Tư vấn”	
9	24/10/2021	_____	Hoàn thành báo cáo	

*Các mục người thực hiện để trống xem như toàn bộ nhóm tham gia*

## Task 1: Requirement elicitation

**Task 1.1. Identify the context of this project. Who are relevant stakeholders? What are expected to be done? What are the scope of the project?**

### GIỚI THIỆU

- ***Bối cảnh của dự án***

Point Of Sale (POS) là thời điểm và nơi giao dịch bán lẻ được thực hiện. Trong dự án này, hệ thống Restaurant POS giúp khách hàng đặt mua hàng và thanh toán online mà không cần tiếp xúc trực tiếp với nhân viên thông qua giải pháp dựa trên web. Cụ thể, hệ thống Restaurant POS của dự án bao gồm các hoạt động đặt bàn trước, gọi thức ăn, cảnh báo, hỗ trợ thanh toán. Dự án có khả năng làm tăng trí tuệ kinh doanh, giảm thiểu nguồn lực lãng phí và cơ hội mở rộng quy mô kinh doanh, đặc biệt trong bối cảnh tình hình dịch bệnh COVID-19 diễn ra như hiện tại.

Dự án được thiết kế cho mô hình nhà hàng truyền thống, có phương thức hoạt động như sau:

- Khách hàng đi đến nhà hàng, quét mã QR vào trang web của nhà hàng để chọn món, bàn và thanh toán (đối với những bàn đã có khách đặt trước và thời điểm hẹn bắt đầu sau thời điểm hiện tại ít hơn 30 phút thì khách mới vào sẽ không thể chọn được dù hiện tại đang trống).

- Khách hàng có thể chọn dùng tại bàn hoặc mang về.

- Thức ăn được chế biến tại chỗ.

- Nhân viên sau khi nhận được hóa đơn đã thanh toán từ khách sẽ chuyển hóa đơn vào bếp và phục vụ cho khách sau khi các món chế biến xong.

- Nhà hàng chấp nhận thanh toán qua thẻ ngân hàng, tiền mặt, ví điện tử.

- ***Các bên liên quan***

- Khách hàng (Customer): dùng hệ thống để xem thực đơn, lựa chọn các món ăn đưa vào giỏ hàng, gửi các yêu cầu thêm về món ăn, thanh toán hóa đơn.

- Nhân viên bán hàng (Clerk): chuyển những yêu cầu từ khách hàng sang bộ phận nhà bếp.

- Bộ phận nhà bếp: đảm nhiệm việc chế biến đồ ăn và thông báo những nguyên liệu và loại đồ ăn đã hết cho nhân viên bán hàng để họ phản hồi lại cho khách hàng.

- Nhân viên IT: đảm nhiệm cài đặt và duy trì hệ thống.

- Kế toán: đảm nhiệm thu phí và thống kê doanh thu cho quản lý.
- Quản lý nhà hàng: quan sát và quản lý các bộ phận trên.
- Chủ cửa hàng: quyết định quy mô ứng dụng, có mở rộng hay không.
- ***Những việc muốn được thực hiện***
  - Đặt bàn.
  - Đặt thức ăn.
  - Cảnh báo.
  - Xuất hóa đơn.
  - Thanh toán sử dụng thẻ và các hình thức ví điện tử khác cũng như tiền mặt.
  - Quản lý khách hàng.
  - Bán mang về.
  - Cho phép giao tiếp gián tiếp giữa nhân viên bán hàng và khách hàng.
- ***Phạm vi của dự án***
  - Hỗ trợ đặt món qua mã QR tại nhà hàng.
  - Hỗ trợ thanh toán không tiền mặt.
  - Hỗ trợ phân quyền người dùng.
  - Không hỗ trợ ghi nợ.
  - Không hỗ trợ tích lũy điểm.
  - Không hỗ trợ thẻ thành viên.
  - Chỉ hỗ trợ đặt bàn trước, không hỗ trợ đặt món trước.

**Task 1.2. Describe all functional and non-functional requirements of the desired system. Draw a use-case diagram for the whole system**

## **CÁC YÊU CẦU CHỨC NĂNG - FUNCTIONAL REQUIREMENTS**

Sau đây là các yêu cầu chức năng mà hệ thống cần cung cấp ứng với mỗi yêu cầu của người dùng:

- **Khách hàng muốn đặt bàn trước**

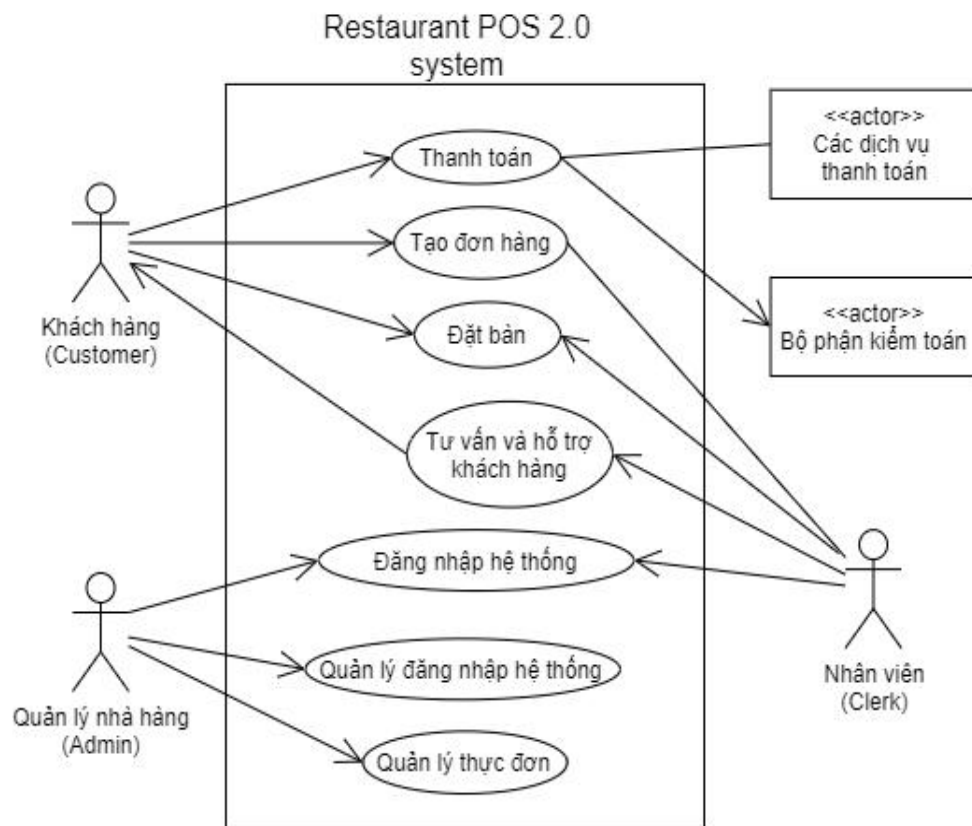
- Hệ thống phải cho phép khách hàng tạo yêu cầu đặt bàn với các lựa chọn về loại bàn, thời điểm và thời gian sử dụng.
- Hệ thống phải thông báo cho khách hàng ngay sau khi hoàn tất đặt bàn và nhắc nhở trước ngày đã đặt 2 ngày.
- **Khách hàng muốn tạo đơn hàng**
  - Hệ thống phải cho phép khách hàng tạo đơn hàng sau khi chọn món.
  - Hệ thống phải cho phép khách hàng để lại những yêu cầu riêng cho món ăn (VD: không bỏ hành, ít cay,...)
  - Hệ thống phải cho phép khách hàng lựa chọn ăn tại quán (khách hàng phải nhập mã bàn mình đang ngồi) hay mang về, không cho phép khách hàng tạo đơn hàng nếu bàn mà khách hàng đang ngồi đã có người đặt trước.
  - Hệ thống phải thông báo cho khách hàng về thông tin đơn.
  - Hệ thống phải cho phép khách hàng thanh toán thông qua các phương thức thanh toán thẻ quốc tế, thẻ quốc nội, ví điện tử, tiền mặt.
  - Hệ thống phải thông báo cho khách hàng nếu có vấn đề về thẻ tín dụng.
- **Nhân viên cần đăng nhập hệ thống**
  - Hệ thống phải cho phép nhân viên đăng nhập thông qua mã nhân viên và mật khẩu.
- **Nhân viên có nhiệm vụ duyệt xác nhận đơn của khách hàng và gửi thông báo cho khách hàng nếu món ăn không phục vụ được**
  - Hệ thống phải chuyển đơn cho nhân viên xác nhận.
  - Hệ thống phải cho phép nhân viên gửi thông báo cho khách hàng nếu món ăn không phục vụ được.
- **Nhân viên có nhiệm vụ duyệt yêu cầu đặt bàn**
  - Hệ thống phải chuyển yêu cầu cho nhân viên xác nhận và cập nhật trạng thái bàn.
- **Nhân viên có nhiệm vụ tư vấn và hỗ trợ khách hàng**
  - Hệ thống phải cho phép nhân viên tư vấn và hỗ trợ khách hàng thông qua chatbox.
- **Quản lý cần quản lý thực đơn**
  - Hệ thống phải cho phép quản lý thêm món ăn.

- Hệ thống phải cho phép quản lý cập nhật thông tin món ăn.
- **Quản lý chịu trách nhiệm quản lý đăng nhập hệ thống**
  - Chỉ quản lý được hệ thống cho phép thêm, sửa, xóa người dùng được đăng nhập vào hệ thống.
- **Bộ phận kiểm toán cần có thông tin các giao dịch, đơn hàng**
  - Hệ thống phải ghi lại các giao dịch, đơn hàng rồi gửi về bộ phận kiểm toán.

## **CÁC YÊU CẦU PHI CHỨC NĂNG - NON-FUNCTIONAL REQUIREMENTS**

- Có giao diện tương thích với các thiết bị di động, máy tính bảng hoặc máy tính/laptop thông thường.
- Tương thích với các trình duyệt phổ biến (Google Chrome, Mozilla Firefox, Microsoft Edge, Opera, Safari).
- Giao diện thân thiện dễ sử dụng, có phần onboarding hướng dẫn khách hàng đặt món và thanh toán, sau 1 giờ hướng dẫn thì các nhân viên đều sử dụng được tốt hệ thống.
- Thời gian tải trang tối đa là 1 giây.
- Lượng truy cập đồng thời tối thiểu là 300.
- Đảm bảo bảo mật thông tin khách hàng và nhân viên.
- Hệ thống luôn sẵn sàng phục vụ trong thời gian làm việc của nhà hàng.
- Các sự cố hệ thống nếu xảy ra phải được xử lý không quá 2 giờ một tuần.
- Dễ mở rộng và phát triển hệ thống trên nhiều nhà hàng.

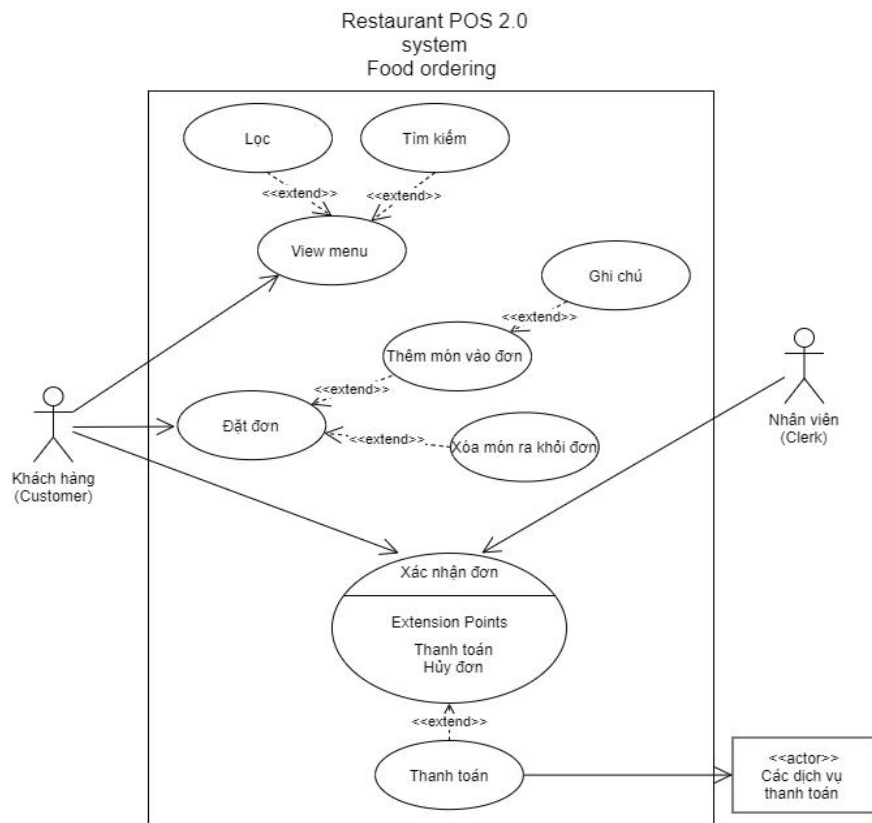
## SƠ ĐỒ USE-CASE TOÀN HỆ THỐNG





**Task 1.3. Choose one specific feature, i.e. food ordering, table reservation, customer management. Draw its use-case diagram and describe the use-case using a table format**

**SƠ ĐỒ USE-CASE MÔ TẢ TÍNH NĂNG FOOD ORDERING:**



**CÁC BẢNG MÔ TẢ USE-CASE CỦA TÍNH NĂNG FOOD ORDERING**

Name	Food ordering
Actor	Khách hàng
Description	Khách hàng đặt đơn qua web
Preconditions	Khách hàng phải quét mã QR thành công . Khách hàng đã vào web thành công.
Post-conditions	Khách hàng đặt đơn hàng thành công Hệ thống ghi nhận đơn hàng

Normal flow	1. Khách hàng truy cập vào web của nhà hàng sau khi quét mã QR. 2. Hệ thống hiển thị thực đơn. 3. Khách hàng đặt đơn. 4. Khách hàng xác nhận đơn bằng cách bấm nút Thanh toán. 5. Nhân viên xác nhận lại đơn. 6. Khách hàng thanh toán đơn. 7. Đơn hàng được thông báo cho bộ phận nhà bếp. 8. Nhà hàng phục vụ món ăn sau khi nhà bếp hoàn thành.
Exceptions	Exception 1: Tại bước 3: 3a. Khách hàng không chọn được món mà họ muốn trên trang xem thực đơn. Exception 2: Tại bước 4: 4a. Khách hàng sẽ không thể ấn nút order nếu đơn trống.
Alternative flow	Alternative 1: Tại bước 3 3a. Khách hàng có thể dùng thanh tìm kiếm để tìm món. Tiếp tục bước 3 trong normal flow Alternative 2: Tại bước 4 4a. Hệ thống đưa ra thông báo không thể order khi đơn trống 4b. Khách hàng có thể ấn nút “select all” để chọn tất cả món trong giỏ hàng. 4c. Khách hàng có thể đặt số lượng cho mỗi món. Tiếp tục bước 4 trong normal flow

### Xem thực đơn

Name	Xem thực đơn
Actor	Khách hàng
Description	Khách hàng có thể xem danh sách các món ăn của nhà hàng
Trigger	Khách hàng quét mã QR
Pre-conditions	Khách hàng phải quét mã QR và vào trang web thành công
Post-conditions	Khách hàng xem được các món ăn của nhà hàng
Normal flow	1. Hệ thống hiển thị các món ăn 2. Khách hàng có thể kéo lên hoặc kéo xuống trang để lựa chọn các món ăn
Exceptions	Exception 1: Tại bước 1: Khách hàng bị mất kết nối với trang web Exception 2: Tại bước 2: Số lượng các món ăn quá ít để thực hiện hành động này

### Lọc món ăn

Name	Lọc món ăn
Actor	Khách hàng

Description	Hiển thị các món ăn phù hợp dựa trên mong muốn của khách hàng đã chọn
Trigger	Khách hàng bấm nút “Lọc”
Pre-conditions	Khách hàng phải đang ở màn hình “Xem menu”
Post-conditions	Hiển thị các món ăn phù hợp với nhu cầu đã chọn của khách hàng
Normal flow	<ol style="list-style-type: none"> <li>1. Khách hàng đánh dấu vào các lựa chọn đã có sẵn trong danh sách chọn theo mong muốn của mình.</li> <li>2. Khách hàng xác nhận bằng cách ấn vào “Lọc” ở cuối danh sách,</li> <li>3. Hệ thống quét một lần qua các lựa chọn để tìm ra các lựa chọn đã được chọn</li> <li>4. Hệ thống quét qua tất cả các món ăn để tìm ra các món ăn phù hợp với các lựa chọn</li> <li>5. Hệ thống hiển thị các món ăn tìm được cho khách hàng lựa chọn</li> </ol>
Exceptions	<p>Exception 1: Tại bước 2: Khách hàng không chọn bất cứ lựa chọn nào nhưng vẫn bấm nút “Lọc”</p> <p>Exception 2: Tại bước 3: Hệ thống không tìm thấy lựa chọn nào được chọn</p> <p>Exception 3: Tại bước 4: Hệ thống không tìm ra món ăn nào phù hợp với các lựa chọn</p>
Alternative flow	<p>Alternative 1: Tại bước 1: 1a: Khách hàng chọn một lần nữa vào lựa chọn đã chọn để hủy lựa chọn đó</p> <p>Alternative 2: Tại bước 5: 5a: Hệ thống sẽ hiển thị: “Không có món ăn nào phù hợp” khi không tìm được món ăn nào phù hợp với mong muốn của khách hàng</p>

### **Tìm kiếm món ăn**

Use-case name	Tìm kiếm món ăn
Actor	Khách hàng
Description	Khách hàng có thể tìm kiếm món ăn qua từ khóa.
Precondition	Khách hàng phải đang ở trang xem thực đơn.
Postcondition	Hệ thống hiện thông tin món ăn mà khách hàng tìm.
Normal flow	<ol style="list-style-type: none"> <li>1. Khách hàng điền vào khung tìm kiếm món ăn.</li> <li>2. Hệ thống hiển thị các món ăn có chứa từ khóa.</li> <li>3. Khách hàng chọn món ăn.</li> </ol>

	4. Hệ thống hiện thông tin món ăn.
Exceptions	Exception 1: Tại bước 2 2a. Nếu không có món ăn chứa từ khóa, hệ thống thông báo với khách hàng không có món chứa từ khóa khách đã nhập.
Alternative flow	Alternative 1: Tại bước 1 1a. Khách hàng nhập từ khóa khác. Tiếp tục bước 1 trong normal flow.

### **Đặt đơn**

Use-case name	Đặt đơn
Actor	Khách hàng
Description	Use case này cho phép khách hàng thêm, xóa món ăn trong đơn hàng
Extended point	1. Thêm món vào đơn hàng 2. Xóa món ra khỏi đơn
Precondition	Khách hàng đang ở trang chứa đơn hàng hiện tại hoặc trang chứa thông tin món ăn hiện tại
Postcondition	Các món ăn khách hàng muốn đặt được lưu trong đơn hàng hiện tại
Normal Flow	1. Extension point: Thêm món vào đơn hàng 2. Extension point: Xóa món ra khỏi đơn 3. Khách hàng vào menu và tiếp tục tìm kiếm món ăn 4. Lặp lại từ bước 1 đến bước 3 đến khi khách hàng không còn nhu cầu đặt món 5. Kết thúc flow
Alternative flow	Alternative 1: tại bước 3: 3a: Kết thúc flow (Khi khách hàng không muốn tiếp tục đặt món)

### **Thêm món vào đơn hàng**

Use-case name	Thêm món vào đơn hàng
Actor	Khách hàng

Description	Use case này cho phép khách hàng chọn món đưa vào đơn hàng
Extends use case	Đặt đơn
Extended point	Ghi chú
Precondition	Khách hàng đang ở trang chứa thông tin của món ăn muốn đặt
Postcondition	Hệ thống ghi nhận món ăn mới đặt vào đơn hàng hiện tại
Normal flow	<ol style="list-style-type: none"> <li>1. Khách hàng nhấn vào nút “Thêm vào đơn hàng”</li> <li>2. Hệ thống hiện lên một form có ô ghi số lượng (mặc định là 1) và ghi chú nếu có từ khách hàng</li> <li>3. Khách hàng nhập số lượng muốn đặt</li> <li>4. Extension point: Ghi chú</li> <li>5. Khách hàng nhấn nút “đặt món”</li> <li>6. Hệ thống ghi lại tên món ăn, số lượng và ghi chú nếu có vào đơn hàng của khách và thông báo đã chuyển thành công vào đơn hàng.</li> <li>7. Hệ thống quay về trang thực đơn.</li> </ol>
Exception	Exception 1: tại bước 5: 5a. Khách hàng không muốn tiếp tục đặt món hiện tại
Alternative flow	Alternative 1: tại bước 5: 5a: Khách hàng nhấn vào dấu x 5b: Hệ thống trở về trang thông tin món ăn 5c: Kết thúc flow

### Ghi chú

Use-case name	Ghi chú
Actor	Khách hàng
Description	Khách hàng ghi chú cho món ăn đã chọn trong đơn hàng
Extends use-case	Thêm món vào đơn
Precondition	Khách hàng đã có món ăn trong đơn hàng
Postcondition	Các ghi chú cho món ăn được lưu trong đơn hàng
Normal flow	<ol style="list-style-type: none"> <li>1. Khách hàng chọn món muốn thêm ghi chú trong đơn hàng</li> <li>2. Hệ thống hiển thị thông tin chi tiết món đó</li> <li>3. Khách hàng viết ghi chú</li> </ol>

	4. Hệ thống ghi nhận ghi chú cho món đó
--	---

#### **Xóa món ra khỏi đơn**

Use-case name	Xóa món ra khỏi đơn (Remove Food item)
Actor	Khách hàng (Customer)
Description	Use case này cho phép khách hàng xóa món ăn đã order khỏi đơn hàng
Extends use case	Đặt đơn
Precondition	Khách hàng đang ở trang xem đơn hàng hiện tại
Postcondition	Hệ thống xóa món ăn ra khỏi đơn hàng
Normal Flow	<ol style="list-style-type: none"> <li>1. Khách hàng nhấn vào dấu x bên cạnh món ăn muốn xóa trong danh sách món ăn trong đơn hàng</li> <li>2. Hệ thống hiện lên một thông báo hỏi khách hàng có chắc muốn xóa món ra khỏi đơn hàng hay không</li> <li>3. Khách nhấn vào nút “Xóa khỏi đơn hàng”</li> <li>4. Hệ thống hiện lên thông báo đã xóa món ăn ra khỏi đơn hàng</li> <li>5. Hệ thống trở lại trang xem đơn hàng hiện tại.</li> </ol>
Alternative flow	Alternative 1: tại bước 3: 3a: Khách hàng nhấn vào nút “Hủy” 3b: Normal flow chuyển đến bước 5

#### **Xác nhận đơn**

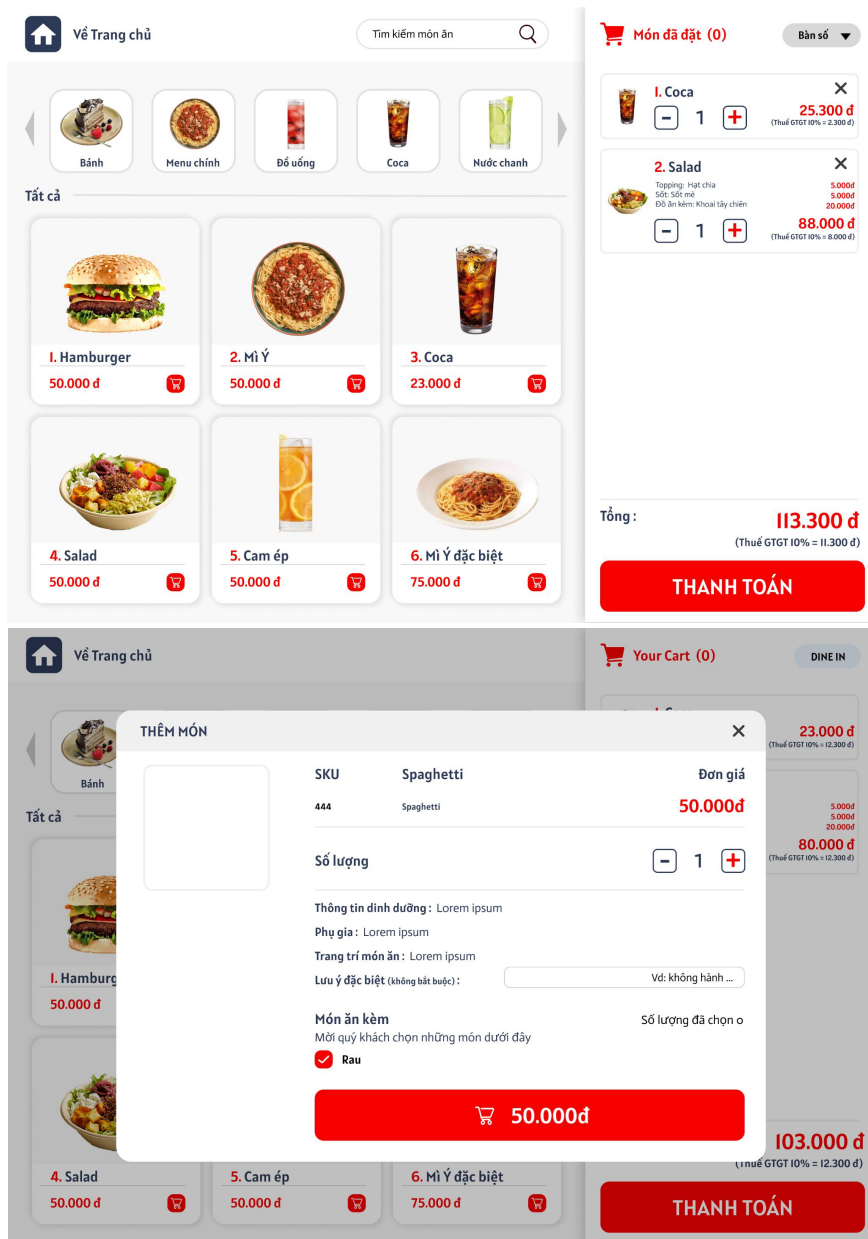
Use-case name	Xác nhận đơn
Actor	Khách hàng, Nhân viên
Description	Sau khi hoàn tất chọn món khách hàng sẽ đặt đơn
Precondition	Khách hàng phải vào được trang thực đơn và thêm món thành công vào giỏ hàng.
Postcondition	Đơn hàng được đặt thành công
Normal flow	<ol style="list-style-type: none"> <li>1. Khách hàng click vào nút thanh toán.</li> <li>2. Hệ thống sẽ ghi nhận lại đơn.</li> <li>3. Nhân viên xác nhận đơn.</li> </ol>

	4. Hệ thống chuyển sang trang thanh toán.
Exceptions	Exception 1: Tại bước 3 Nhân viên không duyệt đơn.
Alternative flow	Alternative 1: Tại bước 3 Chuyển về trang order

### Thanh toán

Use-case name	Thanh toán
Actor	Khách hàng, Dịch vụ thanh toán
Description	Khách hàng sẽ phải thanh toán sau khi đặt đơn
Precondition	Khách hàng phải đặt đơn thành công
Postcondition	Hệ thống ghi nhận đơn hàng, thanh toán thành công
Normal flow	<ol style="list-style-type: none"> <li>1. Hệ thống đưa ra các phương thức thanh toán</li> <li>2. Khách hàng sẽ chọn phương thức thanh toán</li> <li>3. Hệ thống hiển thị thông tin thanh toán</li> <li>4. Khách hàng xác nhận thanh toán</li> <li>5. Hệ thống tương tác với dịch vụ thanh toán mà khách hàng chọn</li> <li>6. Dịch vụ thanh toán thanh toán thành công</li> <li>7. Hệ thống thông báo với khách hàng</li> <li>8. Hệ thống gửi thông tin đơn hàng đến bộ phận nhà bếp</li> </ol>
Exceptions	Exception 1: Tại bước 2 2a. Khách hàng chọn quay lại Exception 2: Tại bước 5 5a. Khách hàng bấm nút hủy thanh toán Exception 3: Tại bước 6 6a. Dịch vụ thanh toán không thành công
Alternative flow	Alternative 1: Tại bước 2 Hệ thống hiển thị trang xem thực đơn Alternative 2: Tại bước 5 Tiếp tục từ bước 1 trong normal flow Alternative 3: Tại bước 6 6a. Hệ thống thông báo thanh toán không thành công Tiếp tục từ bước 1 trong normal flow và tiếp tục

### Giao diện đặt đơn mẫu







Quay lại

THANH TOÁN

Trang chủ Thanh toán

Tổng thanh toán

50.000đ  
không VAT

PHƯƠNG THỨC THANH TOÁN

- ☒ Credit Card - tín dụng hoặc ghi nợ 
- ☐ Thẻ ngân hàng 
- ☐ Ví MOMO 
- ☐ Ví ZaloPay 
- ☐ Tiền mặt 



Điều khoản



Quay lại

THANH TOÁN

Trang chủ Thanh toán

Tổng thanh toán

50.000đ  
không VAT

PHƯƠNG THỨC THANH TOÁN

- ☒ Credit Card - credit or debit 



Card number

MM/YY

CVV



THANH TOÁN 50.000đ

Hủy thanh toán

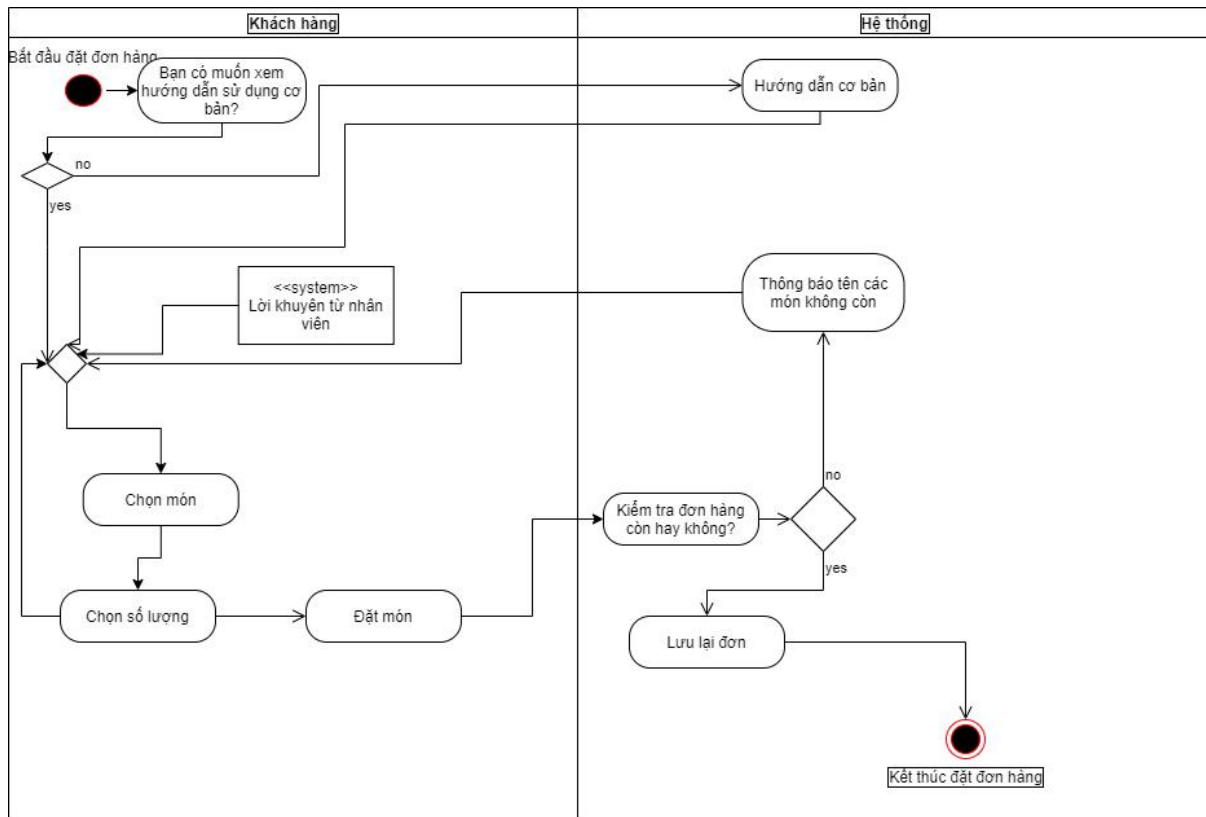


Điều khoản

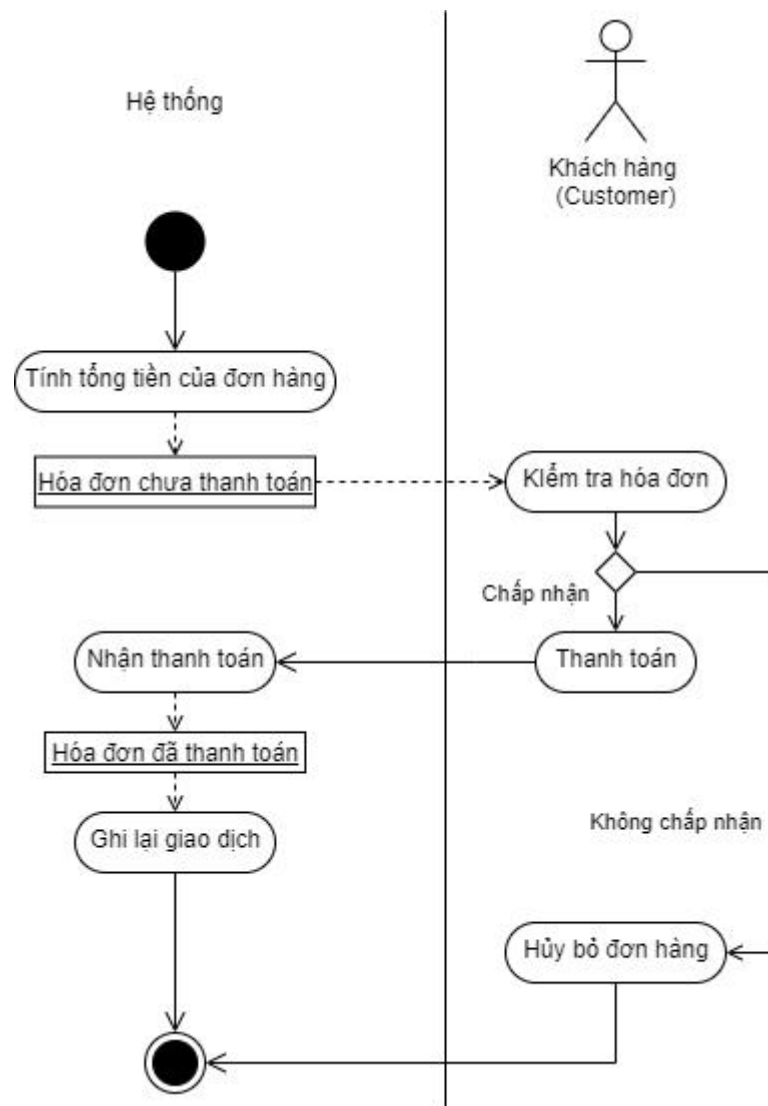
## Task 2: System modelling

Task 2.1. Draw an activity diagram to capture Major (not all) functional requirements of the desired system

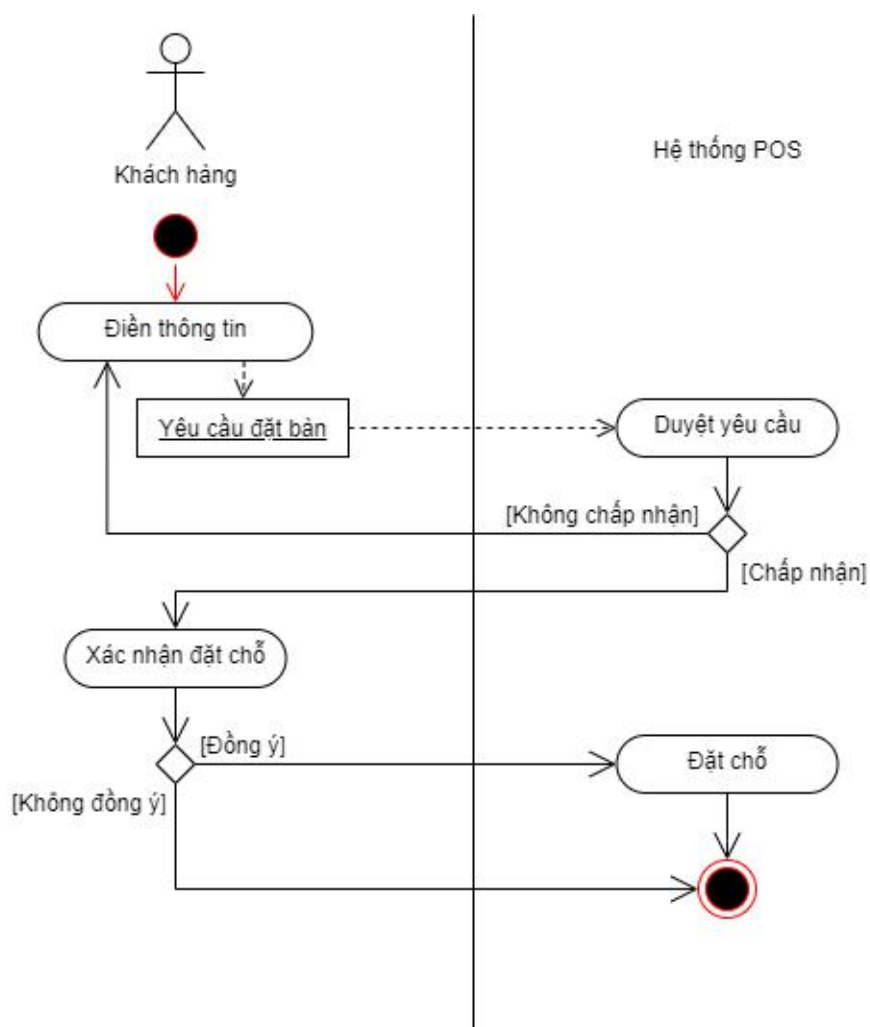
Tạo đơn hàng



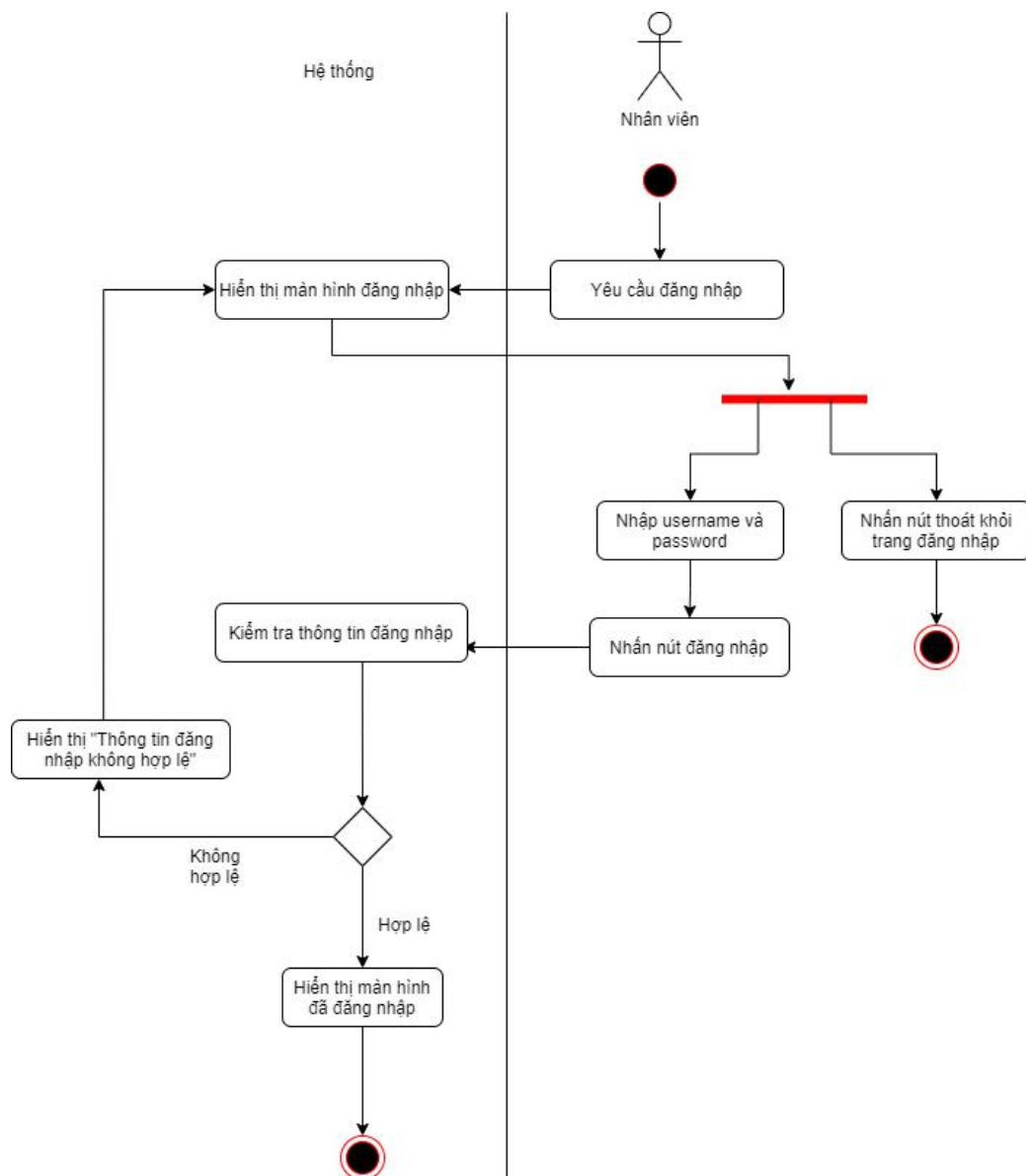
## Thanh toán



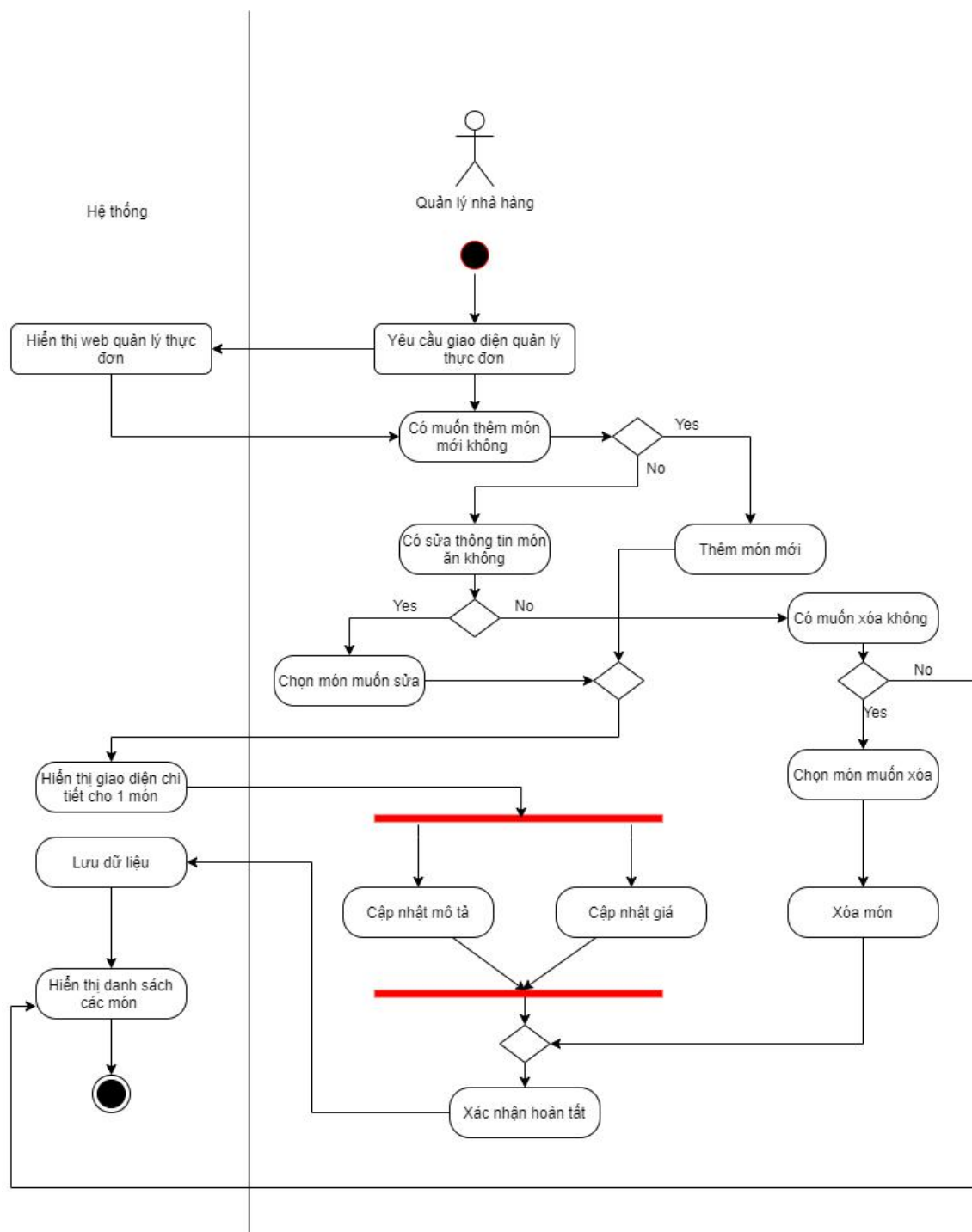
## Đặt bàn



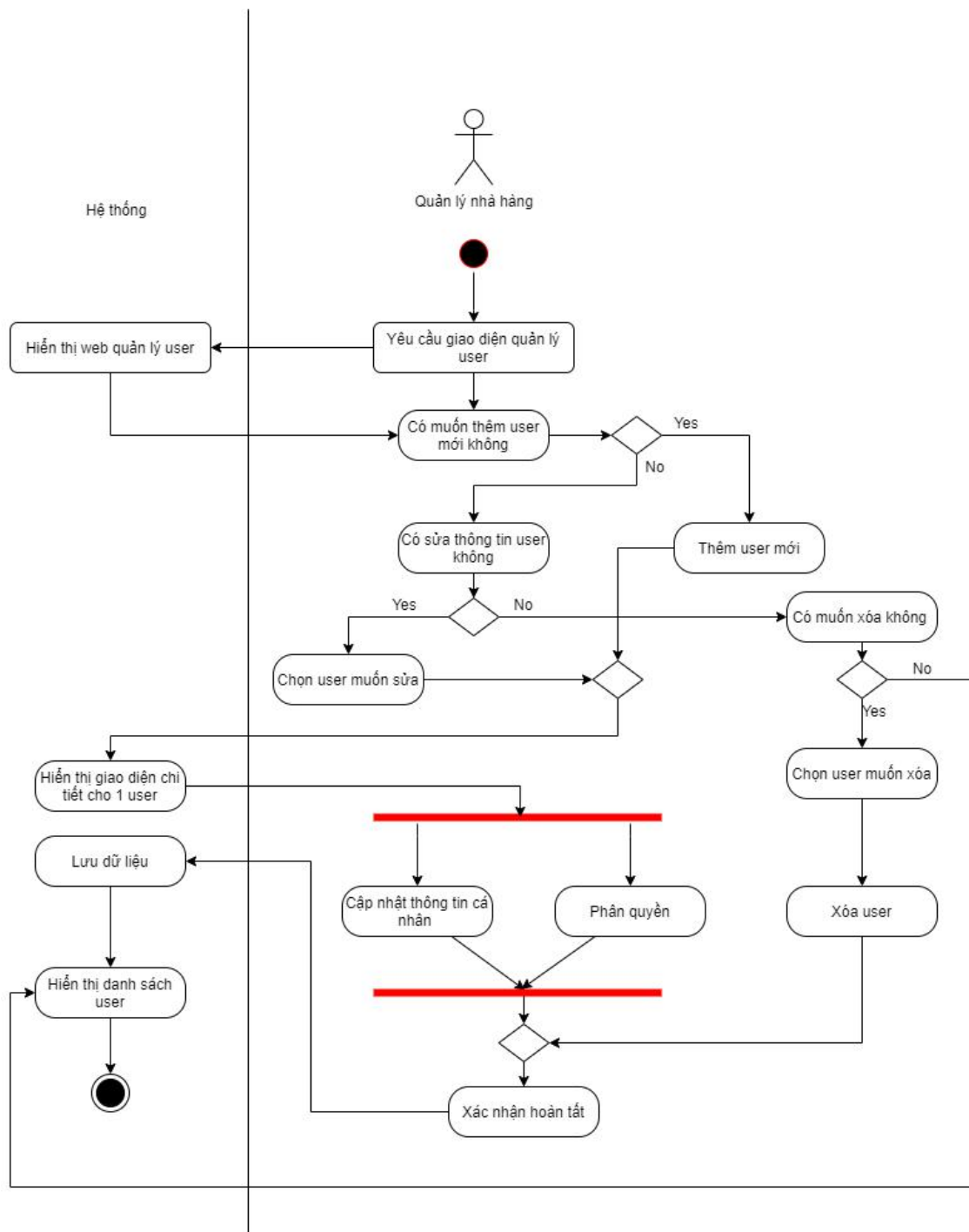
## Đăng nhập cho nhân viên (nhân viên bán hàng và quản lý)



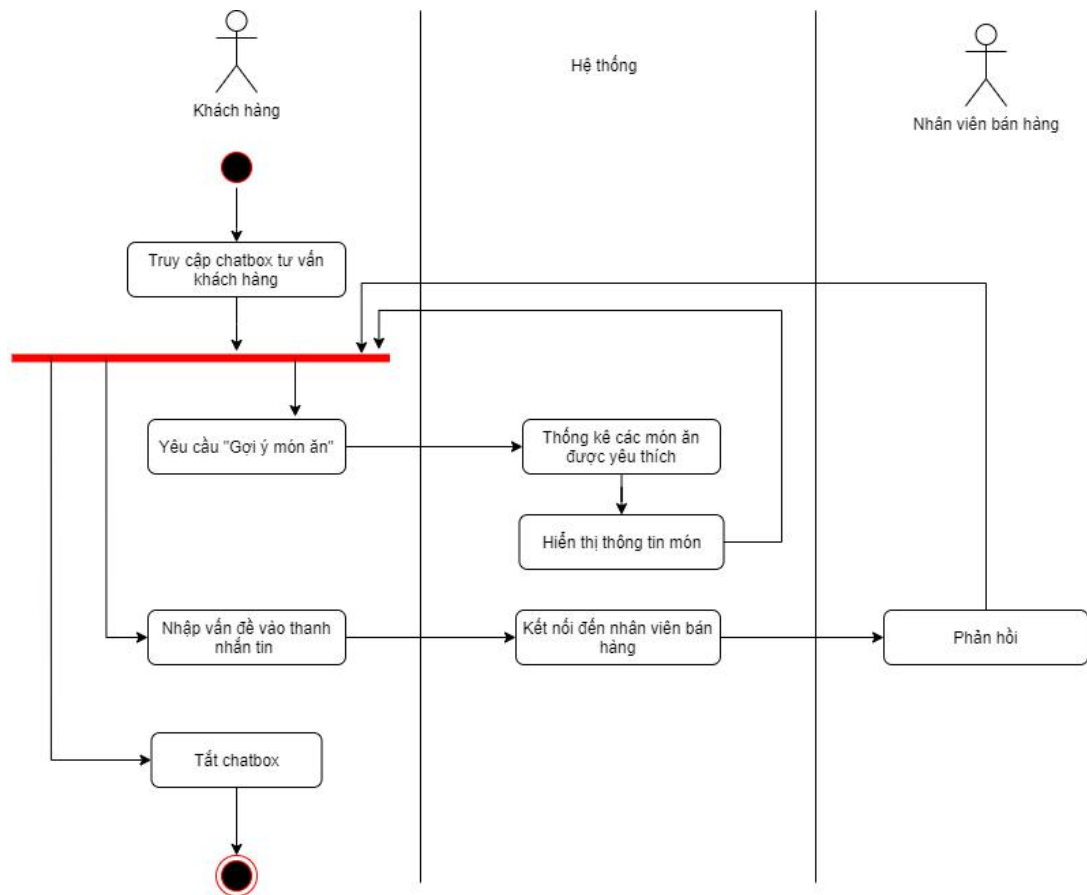
## Quản lý thực đơn



## Quản lý tài khoản người dùng



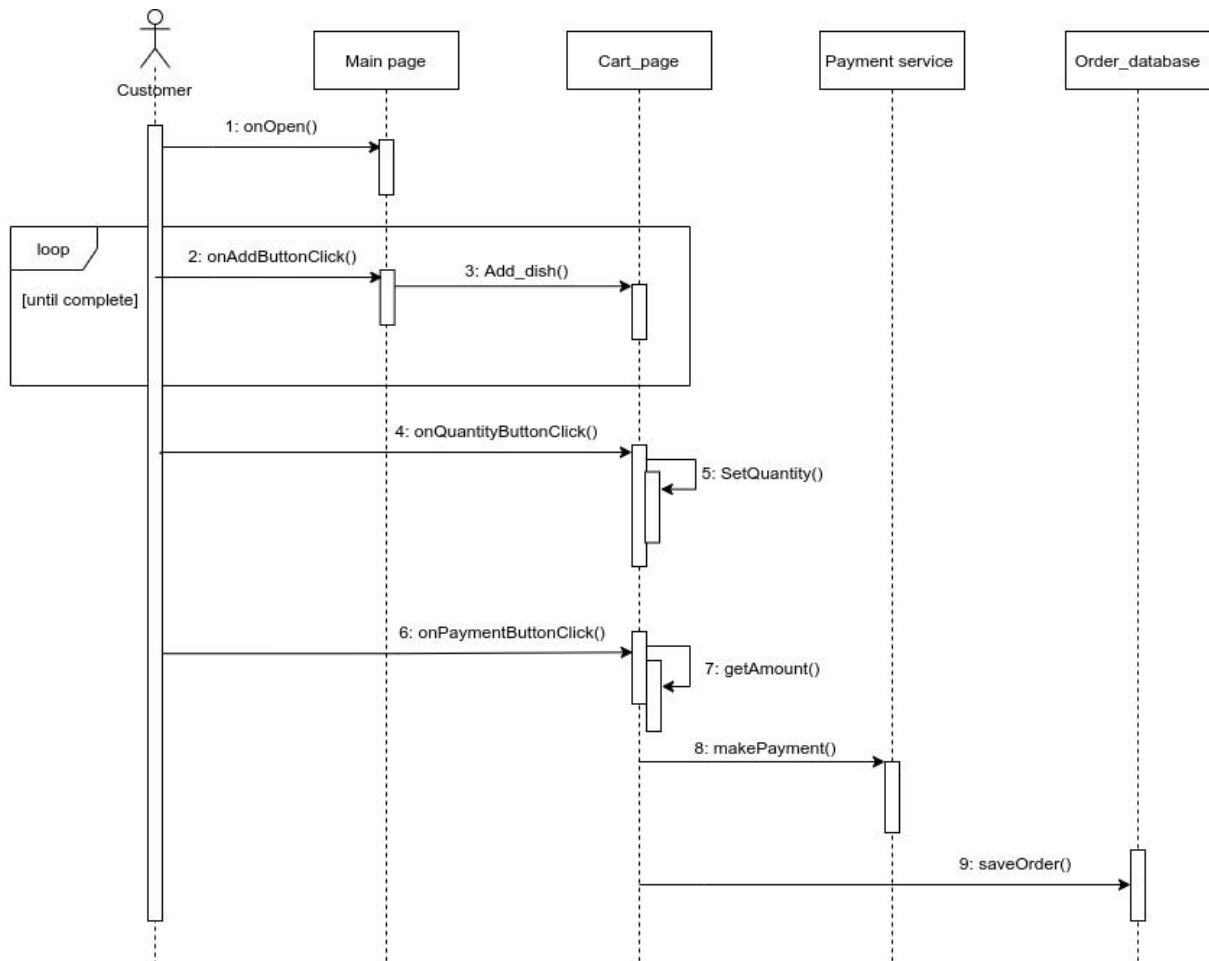
## Tư vấn khách hàng





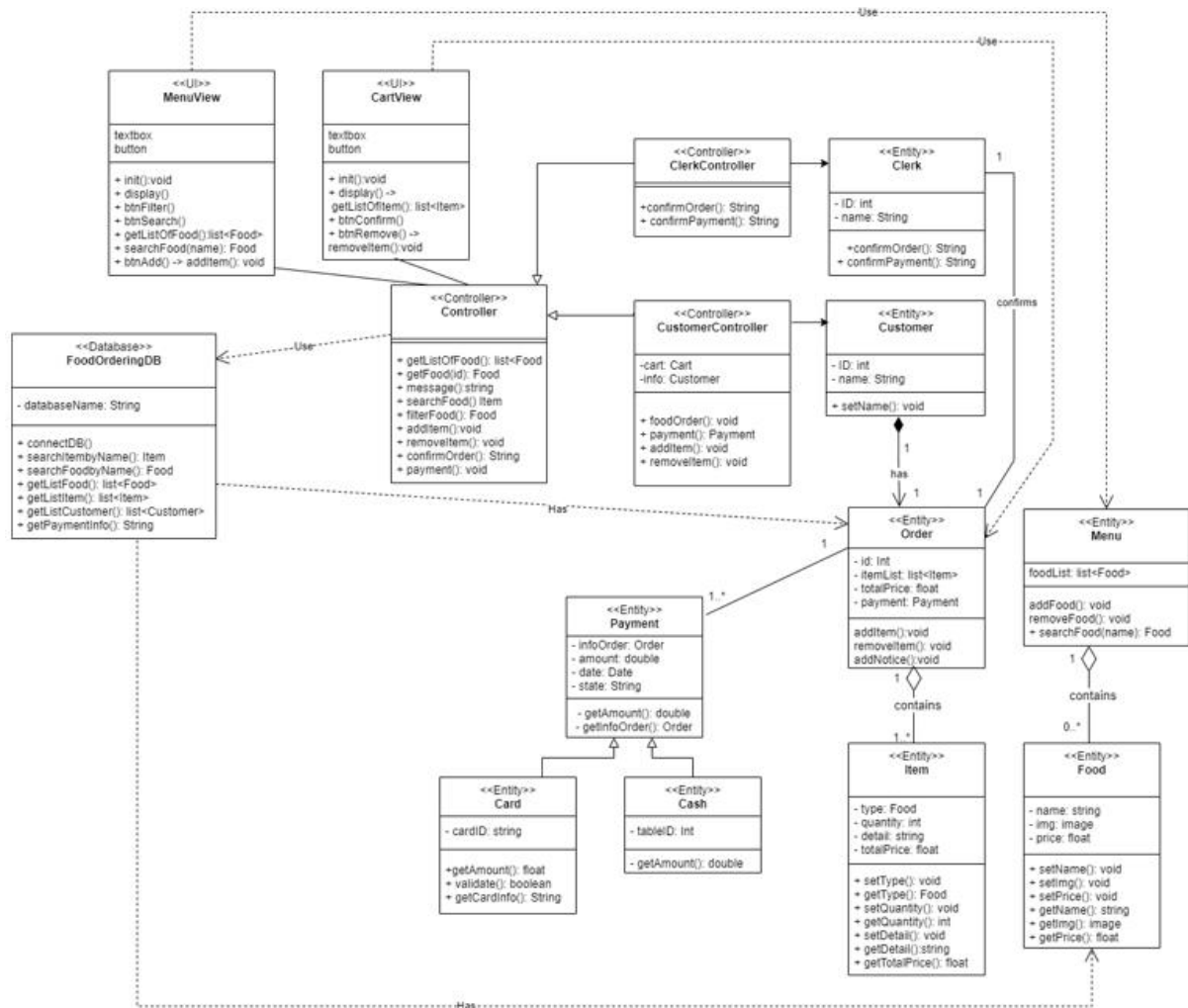
## Task 2.2. Draw a sequence diagram for use-case in Task 1.3

### Sequence diagram for Food Ordering



## Task 2.3. Draw a class diagram

Link: <https://drive.google.com/file/d/1ypRick9ZwK0QqCkbNkcGD8TnGeJU2Ae7/view?usp=sharing>



## Task 3: Architecture Design

### Task 3.1: Describe an architectural approach you will use to implement the desired system

MVC là một kiểu kiến trúc phần mềm, gồm ba thành phần riêng biệt, cho phép sửa đổi từng thành phần một cách độc lập:

- Model: lớp đại diện tương tác với cơ sở dữ liệu được sử dụng bởi chương trình. Nó có thể là cơ sở dữ liệu, file hay một đối tượng đơn giản. Lớp này có thể tái sử dụng
- View: hiển thị các đối tượng trong ứng dụng lên màn hình. Các đối tượng như cửa sổ, văn bản, nút bấm. Nó bao gồm tất cả những gì người dùng nhìn thấy
- Controller: hoạt động như thành phần trung gian. Nó nhận yêu cầu từ người dùng và thực hiện các thay đổi bên trong Model và View. Lớp Controller sẽ có thể cập nhật một Model bằng cách thay đổi giá trị thuộc tính trong Model. Ngoài ra, Controller thể sửa đổi View bằng cách hiển thị cập nhật đó.

\* Cách hoạt động trong mô hình MVC:

1. Khi một yêu cầu của từ máy khách (Client) gửi đến Server. Thì bị Controller trong MVC chặn lại để xem đó là URL request hay sự kiện.
2. Sau đó, Controller xử lý input của user rồi giao tiếp với Model trong MVC.
3. Model chuẩn bị data và gửi lại cho Controller.
4. Cuối cùng, khi xử lý xong yêu cầu thì Controller gửi dữ liệu trở lại View và hiển thị cho người dùng trên trình duyệt.

Ở đây, View không giao tiếp trực tiếp với Model. Sự tương tác giữa View và Model sẽ chỉ được xử lý bởi **Controller**.

\* Tính ưu việt của mô hình MVC:

- Tăng tốc quy trình develop:

Vì có ba phần chính nên một developer có thể làm việc trên View, một developer khác sẽ làm việc trên Controller hoặc Model. Do đó, mô hình MVC có thể hoàn thành nhanh gấp 3 lần ứng dụng mô hình khác.

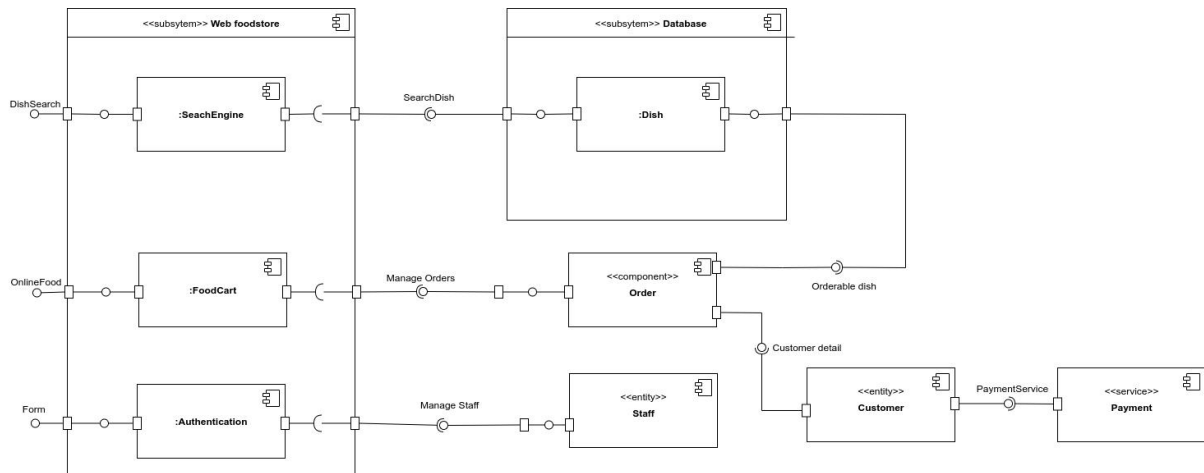
- Cung cấp nhiều chế độ view: Trong mô hình MVC, ta có thể tạo nhiều view chỉ cho một mô hình.

- Các sửa đổi không ảnh hưởng đến toàn bộ mô hình

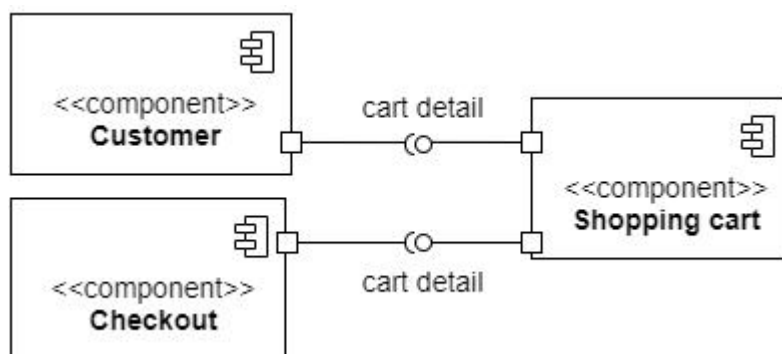
Hiện nay, với việc interface có xu hướng thay đổi thường xuyên cụ thể là font chữ, màu sắc hay bố cục màn hình sẽ được giải quyết dễ dàng với mô hình MVC.

### Task 3.2: · Draw an implementation diagram for Major (not all) functional requirements

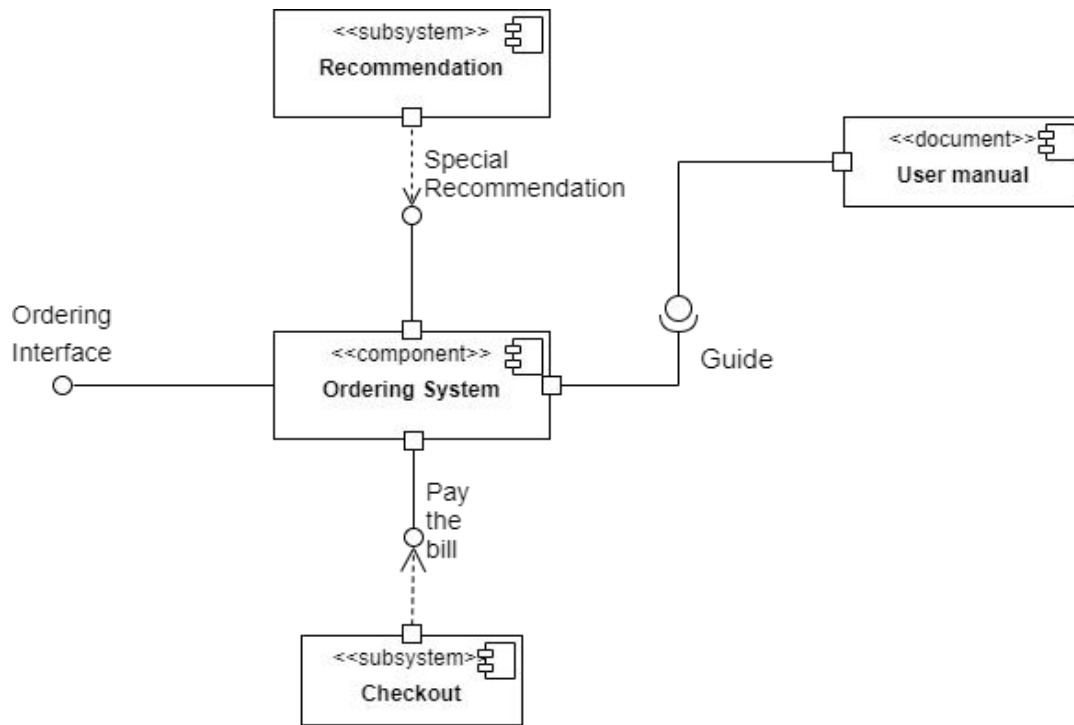
Hệ thống:



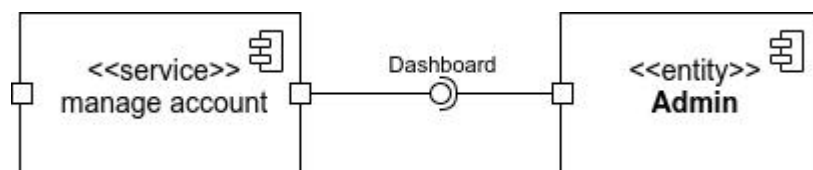
- Thanh toán: Khách hàng xem chi tiết giỏ hàng từ Shopping cart và bộ phận checkout lấy chi tiết giỏ hàng từ Shopping cart:



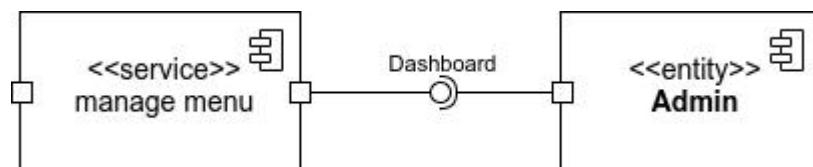
- Tạo đơn hàng: Mọi hoạt động chọn món của khách hàng được Ordering system xử lý. Ngoài ra, khách hàng còn có thể nhận lời khuyên thông qua một hệ thống khác là Recommendation và có thể xem một số hướng dẫn thông qua User manual. Sau đó khách hàng sẽ được chuyển sang hệ thống Checkout để thanh toán.



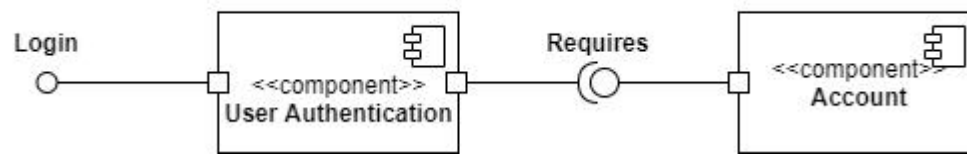
- Quản lý tài khoản: Việc quản lý tài khoản người dùng do Manage Account xử lý và Admin tham gia vào việc quản lý tài khoản



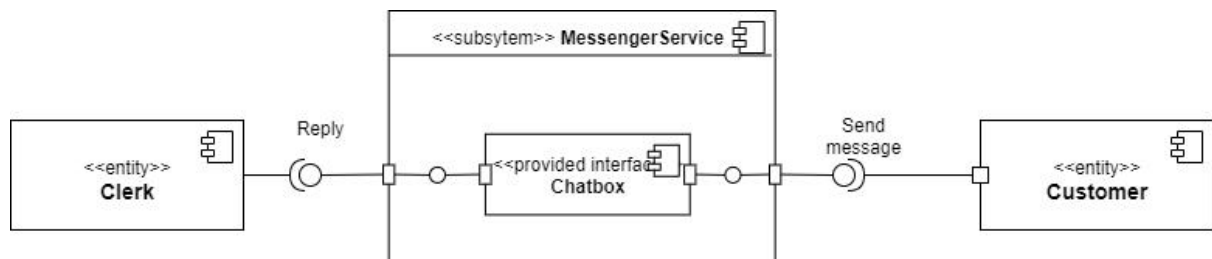
- Quản lý thực đơn: Việc quản lý thực đơn do Manage Account xử lý và Admin tham gia vào việc quản lý thực đơn



- Đăng nhập: Nhân viên nhà hàng thực hiện đăng nhập, có một component User Authentication kiểm tra việc đăng nhập hợp lệ và yêu cầu thông tin tài khoản đăng nhập từ Account



- Tư vấn: Hệ thống Messenger System quản lý việc tư vấn khách hàng, ghi nhận và hiển thị tin nhắn đến từ Customer và Clerk thông qua giao diện Chatbox



- Đặt bàn: Nhà hàng cung cấp dịch vụ đặt bàn và quản lí bằng cách lưu lại Booking Record trong cơ sở dữ liệu của hệ thống.

