

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN**

-----o0o-----



**BÁO CÁO ĐỒ ÁN
NHẬP MÔN CÔNG NGHỆ PHẦN MỀM
ĐỒ ÁN
HỆ THỐNG ĐẶT ĐỒ ĂN TRỰC TUYẾN
CHO NHÀ HÀNG**

Lớp:	E21CQCN01-N
Nhóm thực hiện:	Nhóm 10
Giảng viên hướng dẫn:	Huỳnh Lưu Quốc Linh

TP.HCM, 06/2024

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN**

-----o0o-----



**ĐỒ ÁN
HỆ THỐNG ĐẶT ĐỒ ĂN TRỰC TUYẾN
CHO NHÀ HÀNG**

Lớp:	E21CQCN01-N
Giảng viên hướng dẫn:	Huỳnh Lưu Quốc Linh
Nhóm thực hiện:	Nhóm 10
1. Nguyễn Hữu Dũng	N21DCCN018
2. Phạm Thanh Linh	N21DCCN139
3. Hà Hữu Thắng	N21DCCN174
4. Nguyễn Hoàng Thiện	N21DCCN080

This image shows a full page of a document template designed for handwritten notes or essays. It features approximately 28 evenly spaced, thin grey horizontal lines across the entire width of the page. The margins are consistent on all sides, providing a clear area for writing. There are no vertical lines, headers, footers, or other markings present.

- **Điểm chữ:**

(ký và ghi rõ họ tên)

LỜI CẢM ƠN

Để hoàn thiện được bài tiểu luận này một cách tốt nhất, em xin gửi lời cảm ơn chân thành đến Ban giám hiệu, các phòng ban, khoa và các thầy cô giảng viên tại Học viện Công nghệ Bưu chính Viễn thông cơ sở tại thành phố Hồ Chí Minh đã tận tình dạy dỗ và tạo điều kiện cho nhóm 10 trong quá nghiên cứu và học tập.

Đặc biệt, nhóm 10 muốn gửi lời cảm ơn sâu sắc đến thầy **Huỳnh Lưu Quốc Linh** - người đã trực tiếp hướng dẫn nhóm em nghiên cứu đề tài ***Hệ thống đặt đồ ăn trực tuyến cho nhà hàng*** một cách rất nhiệt tình và tận tâm.

Nhóm đã cố gắng vận dụng những kiến thức đã học được và tìm tòi thêm nhiều thông tin để hoàn thành bài tiểu luận này. Tuy nhiên, do sự hiểu biết còn hạn chế nên trong bài tiểu luận có thể không tránh khỏi còn nhiều thiếu sót. Nhóm 10 mong nhận được sự góp ý của thầy, để đồ án được hoàn thiện tốt hơn. Nhóm xin chân thành cảm ơn

MỤC LỤC

MỤC LỤC.....	1
LỜI GIỚI THIỆU	4
CHƯƠNG 1. KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU	5
1.1. Giới thiệu:	5
1.2. Khảo sát hiện trạng:	5
1.2.1. Phân tích thị trường:	5
1.2.2. Phân tích nhu cầu của khách hàng:	6
1.2.3. Phân tích yêu cầu của nhà hàng:	6
1.2.4. Phân tích hệ thống đặt đồ ăn trực tuyến hiện có:	6
1.3. Thu thập yêu cầu.....	7
1.3.1. Thu thập yêu cầu từ khách hàng:.....	7
1.3.2. Thu thập yêu cầu từ nhà hàng:	7
1.3.3. Phân tích và lập danh sách yêu cầu:	7
1.4. Kết luận:.....	7
CHƯƠNG 2. PHÂN TÍCH CÁC TRƯỜNG HỢP SỬ DỤNG (USE CASE).....	8
2.1. Use Case chung (đăng nhập, đăng ký, xem chi tiết đơn hàng).....	9
2.2. Khách Hàng	10
2.2.1. Xem thông tin khách hàng.....	10
2.2.2. Khách Hàng:	11
2.2.3. Chỉnh sửa giỏ hàng:.....	13
2.2.4. Cập nhật trạng thái đơn hàng:	15
2.3. Admin	16
2.3.1. Quản lý món ăn	16
2.3.2. Quản lý người dùng.....	18
2.3.3. Quản lý đơn hàng	19
2.3.5. Quản lý doanh thu	19
2.3.6. Quản lý đánh giá.....	20
2.4. Nhân viên.....	21
2.4.1. Cập nhật trạng thái đơn hàng	21

CHƯƠNG 3. THIẾT KẾ DỮ LIỆU	22
3.1. Tổng quan	22
3.1.1. Mục tiêu.....	22
3.1.2. Phạm vi và mô tả hệ thống	22
3.2. Mô tả dữ liệu.....	23
3.2.1. Loại dữ liệu.....	23
3.2.2. Nguồn dữ liệu	24
3.3. Cấu trúc dữ liệu.....	24
3.3.1. Mô hình dữ liệu	24
3.3.2. Bảng dữ liệu	25
3.3.3. Chi tiết kỹ thuật	26
3.4. Kết luận.....	27
CHƯƠNG 4. THIẾT KẾ GIAO DIỆN	28
4.1. Customer	28
4.1.1. Thiết kế giao diện	28
4.1.2. Thiết kế chi tiết các hàm.....	35
4.2. Admin	36
4.2.1. Thiết kế giao diện	36
4.2.2. Thiết kế chi tiết các hàm.....	45
4.3. Staff.....	48
4.3.1. Thiết kế giao diện	48
4.3.2. Thiết kế chi tiết các hàm.....	49
CHƯƠNG 5. CÀI ĐẶT VÀ KẾT QUẢ THỰC NGHIỆM.....	50
5.1. Cài đặt	50
5.1.1. Môi trường phát triển	50
5.1.2. Mô hình phát triển	50
5.2. Triển khai và thực nghiệm	51
5.2.1. Báo cáo Kiểm thử.....	51
5.2.2. Đánh giá chức năng	53
CHƯƠNG 6. TỔNG KẾT.....	55

6.1. Mục tiêu và phạm vi của đề tài.....	55
6.2 Tóm tắt kết quả đạt được.....	55
6.3. Đánh giá kết quả.....	55
6.4. Học hỏi và đề xuất tương lai	55
6.5. Kết luận	55
TÀI LIỆU THAM KHẢO.....	56
PHÂN CÔNG CÔNG VIỆC.....	57

LỜI GIỚI THIỆU

Trong thời đại công nghệ số hiện nay, việc đặt đồ ăn trực tuyến đã trở thành một nhu cầu thiết yếu đối với nhiều người. Để đáp ứng nhu cầu này, nhóm 10 đã phát triển một hệ thống đặt đồ ăn trực tuyến dành cho nhà hàng. Hệ thống này không chỉ giúp khách hàng dễ dàng đặt món ăn mà còn hỗ trợ nhà hàng quản lý đơn hàng một cách hiệu quả.

Trước tiên, nhóm chúng em tiến hành khảo sát hiện trạng, bao gồm việc phân tích thị trường, nhu cầu khách hàng và yêu cầu từ nhà hàng. Nhóm cũng xem xét các hệ thống đặt đồ ăn trực tuyến hiện có để xác định tính năng và hạn chế. Từ đó, nhóm thu thập và phân tích yêu cầu từ cả khách hàng và nhà hàng, xây dựng nền tảng cho hệ thống.

Tiếp theo, nhóm phân tích các trường hợp sử dụng chính của hệ thống. Đối với khách hàng, hệ thống cung cấp các chức năng như đăng ký, đăng nhập, đặt món, thanh toán và quản lý đơn hàng. Đối với admin, hệ thống hỗ trợ quản lý món ăn, người dùng, đơn hàng, doanh thu và đánh giá nhận xét. Nhân viên có thể cập nhật trạng thái đơn hàng. Mỗi nhóm người dùng có các trường hợp sử dụng riêng biệt, đảm bảo hệ thống đáp ứng đầy đủ nhu cầu của từng đối tượng.

Trong giai đoạn thiết kế dữ liệu, nhóm mô tả các loại dữ liệu, nguồn dữ liệu và cấu trúc dữ liệu của hệ thống. Nhóm xây dựng mô hình dữ liệu, các bảng dữ liệu và chi tiết kỹ thuật để đảm bảo hệ thống có nền tảng dữ liệu vững chắc và có thể mở rộng trong tương lai.

Tiếp đó, nhóm tập trung vào việc thiết kế giao diện người dùng. Với mục tiêu tạo ra một hệ thống dễ sử dụng và thân thiện với người dùng, nhóm áp dụng các nguyên tắc thiết kế giao diện hiện đại. Người dùng có thể dễ dàng thực hiện các thao tác như đặt món, thanh toán và quản lý đơn hàng. Các mẫu giao diện cụ thể được trình bày để minh họa cho thiết kế.

Quá trình cài đặt và kết quả thực nghiệm mô tả việc thiết lập môi trường phát triển và triển khai hệ thống. Nhóm thực hiện các thử nghiệm để đánh giá hiệu suất và độ ổn định của hệ thống. Kết quả thực nghiệm được phân tích để đánh giá tính hiệu quả và khả năng đáp ứng của hệ thống đối với nhu cầu của người dùng và nhà hàng.

Cuối cùng, báo cáo tổng kết mục tiêu và phạm vi của đề tài, những kết quả đạt được và đánh giá tổng quan về dự án. Nhóm chia sẻ những kinh nghiệm học hỏi được trong quá trình phát triển hệ thống và đưa ra các đề xuất cho tương lai nhằm cải thiện và mở rộng hệ thống. Kết luận khẳng định tính khả thi và tiềm năng của dự án trong việc cải thiện trải nghiệm khách hàng và nâng cao hiệu suất quản lý nhà hàng.

CHƯƠNG 1. KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU

1.1. Giới thiệu:

Mục tiêu của việc khảo sát:

- Đánh giá mức độ hiệu quả và tính ứng dụng của đồ án.
- Xác định xu hướng phát triển và khả năng áp dụng thực tế của đồ án.
- Thu thập yêu cầu và mong muốn của khách hàng cũng như yêu cầu và mong muốn của các nhà hàng hiện nay.

Phạm vi khảo sát:

- Phạm vi địa lý: Khảo sát được thực hiện trên địa bàn Thành phố Hồ Chí Minh.
- Phạm vi chức năng: Bao gồm các khía cạnh liên quan đến đặt hàng, giao hàng, thanh toán, quản lý hệ thống, chăm sóc khách hàng và dịch vụ bổ sung.

Đối tượng khảo sát:

- Về phía khách hàng: Thực hiện các khảo sát chủ yếu ở các khách hàng có độ tuổi 15 đến 30 tuổi, vì đây là nhóm khách hàng có nhu cầu và sẵn sàng thử những trải nghiệm mới như là nhà hàng trực tuyến.
- Về phía nhà hàng: Thực hiện khảo sát ở chủ yếu ở các nhà hàng không thuộc trung tâm thành phố vì đây là những nhà hàng có ít lượng khách hơn và cần cải thiện số lượng khách hàng thông qua chương trình Nhà hàng Trực tuyến.

1.2. Khảo sát hiện trạng:

1.2.1. Phân tích thị trường:

Sau quá trình phân tích thị trường thì nhóm nhận thấy thị trường giành cho nhà hàng trực tuyến khá rộng mở. Đa phần các ứng dụng đặt đồ ăn trực tuyến hiện nay chủ yếu giành cho các món ăn nhanh, chưa phổ biến ở các món ăn truyền thống.

Các đối thủ cạnh tranh chính: Là các trang thương mại điện tử, những trang này chủ yếu bán hàng điện tử, gia dụng và các sản phẩm khác. Hiện nay họ đã và đang phát triển sang các mảng bán đồ ăn trực tuyến. Điểm mạnh của các trang thương mại điện tử là họ đã có sẵn lượng khách hàng từ đầu nên không gặp khó khăn trong việc tìm kiếm người sử dụng trong thời gian đầu. Điểm yếu của các trang này là họ bán rất nhiều sản phẩm chứ không tập trung vào bán đồ ăn.

Xu hướng của thị trường: Hiện nay xu hướng tiêu dùng của đại đa số mọi người là mua sắm trực tuyến, do đó trong tương lai gần việc các nhà hàng kinh doanh trực tuyến sẽ là một xu hướng mới vì nó giúp cho khách hàng có thể lựa chọn được các món ăn lành mạnh, các món ăn truyền thống, phù hợp với phần lớn người dùng hơn.

1.2.2. Phân tích nhu cầu của khách hàng:

Hiện nay nhu cầu đặt đồ ăn trực tuyến đang tăng cao. Theo khảo sát và phân tích thì rất nhiều khách hàng ở độ tuổi 15 - 30 tuổi có nhu cầu đặt đồ ăn trực tuyến cao. Qua đó ta có thể thấy được tiềm năng của việc phát triển một ứng dụng chuyên về đặt đồ ăn trực tuyến.

Hành vi và thói quen đặt đồ ăn trực tuyến: Thông qua phân tích thì và đánh giá thì khách hàng hiện nay đặt đồ ăn trực tuyến thông qua các thiết bị như ứng dụng di động, website chiếm phần lớn.

Mong muốn và kỳ vọng của khách hàng đối với các nhà hàng trực tuyến: Mong muốn được trải nghiệm đa dạng các món ăn hơn mà không phải đi xa, giao hàng nhanh hơn, có nhiều chương trình khuyến mãi và nhiều tiện ích khác.

1.2.3. Phân tích yêu cầu của nhà hàng:

Yêu cầu về các chức năng: Các nhà hàng hiện nay mong muốn có các phần mềm giúp họ quản lý tốt hơn về đơn hàng, khách hàng, nhân viên.

Yêu cầu về hệ thống thanh toán, giao hàng:

- + Các nhà hàng hiện nay mong muốn hệ thống thanh toán đa dạng hơn, nhanh hơn.

- + Mong muốn có đơn vị giao hàng nhanh, giúp giảm thời gian đưa sản phẩm đến với khách hàng.

Yêu cầu về báo cáo và thống kê doanh thu: Các nhà hàng hiện nay mong muốn các chức báo cáo có thể tùy chỉnh theo ngày, theo tuần, theo tháng và hơn hết nữa họ mong muốn các báo cáo phải chi tiết và có khả năng tùy chỉnh theo nhu cầu của nhà hàng.

1.2.4. Phân tích hệ thống đặt đồ ăn trực tuyến hiện có:

1.2.4.1. Đánh giá ưu và nhược điểm của các website đặt đồ ăn trực tuyến hiện nay:

Ưu điểm: Các website đặt đồ ăn trực tuyến hiện nay có nhiều tính năng mua hàng hơn, có lượng khách hàng ổn định hơn.

Nhược điểm: Các website đặt đồ ăn này chưa đa dạng được các món ăn, thời gian giao hàng chậm do phải liên kết với đơn vị vận chuyển khác, thời gian xử lý đơn hàng chậm do có nhiều người sử dụng.

1.2.4.2 Xác định các chức năng cần thiết cho đồ án:

- Chức năng xử lý đơn hàng cần được hoàn thiện tốt hơn để có thể xử lý đơn hàng nhanh hơn.
- Chức năng thanh toán đa dạng hơn.
- Cần có chức năng nhận lại các phản hồi của khách hàng để có thể cải thiện chức năng dịch vụ tốt hơn.

1.3. Thu thập yêu cầu

1.3.1. Thu thập yêu cầu từ khách hàng:

- Phỏng vấn trực tiếp khách hàng: Nhóm đã có nhiều đợt phỏng vấn trực tiếp các khách hàng tại nhiều địa điểm khác nhau.
- Khảo sát trực tuyến: Nhóm đã đưa ra nhiều khảo sát trên các trang mạng xã hội như facebook, zalo...

1.3.2. Thu thập yêu cầu từ nhà hàng:

- Nhóm đã phỏng vấn trực tiếp các chủ nhà hàng trong khu vực Thành phố Hồ Chí Minh từ đó tìm ra được các vấn đề dưới góc nhìn của nhà hàng.
- Nhóm đã tổng hợp được dữ liệu bán hàng của nhiều nhà hàng ở nhiều khu vực khác nhau trong Thành phố.

1.3.3. Phân tích và lập danh sách yêu cầu:

1.3.3.1 Mong muốn của khách hàng:

- Qua các thông tin nhóm đã thu thập được thì có các yêu cầu chính từ khách hàng như:
 - + Mong muốn có nhiều lựa chọn hơn cho các món ăn.
 - + Mong muốn có nhiều tiện ích hơn như: Bình luận sản phẩm, đánh giá sản phẩm.
 - + Mong muốn có đội ngũ hỗ trợ khách hàng.
 - + Mong muốn có thể lựa chọn được các nguyên liệu để nấu cho món ăn đã đặt.

1.3.3.2. Yêu cầu của nhà hàng:

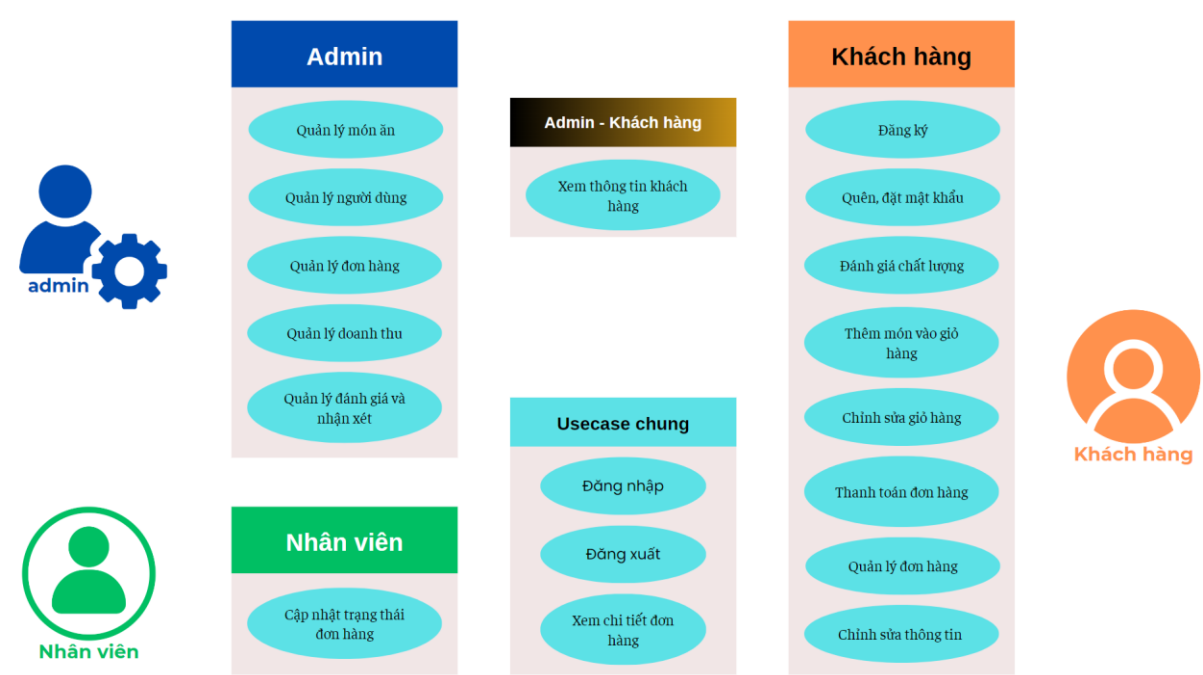
- Qua các thông tin mà nhóm đã thu thập được thì các yêu cầu chính của nhà hàng thường xoay quanh các vấn đề sau:
 - + Yêu cầu về quản lý nhân viên.
 - + Yêu cầu về báo cáo và tổng kết thông tin.
 - + Yêu cầu về xử lý hệ thống.
 - + Yêu cầu về giao diện dễ thao tác.

1.4. Kết luận:

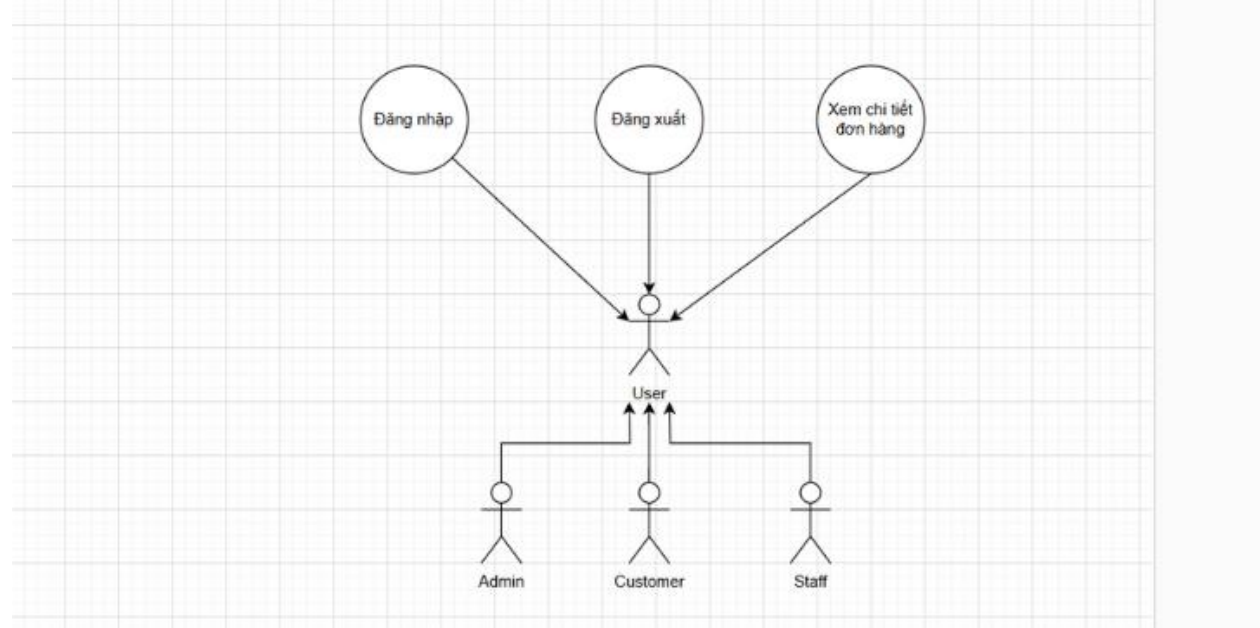
Tổng kết lại, quá trình khảo sát và thu thập yêu cầu đã giúp nhóm thu thập nhiều thông tin quan trọng về thị trường đặt đồ ăn trực tuyến, bao gồm quy mô, tiềm năng phát triển, phân tích đối thủ cạnh tranh, và nhu cầu của khách hàng cũng như nhà hàng. Những thông tin này đã dẫn đến các đề xuất phát triển thiết thực cho website. Nhóm tập trung vào việc cải thiện trải nghiệm người dùng với giao diện thân thiện, hỗ trợ phương thức thanh toán, và xây dựng hệ thống phản hồi để liên tục nâng cao chất lượng dịch vụ. Mục tiêu là tạo ra một nền tảng đặt đồ ăn trực tuyến tiện lợi, hiệu quả và đáp ứng tối đa các yêu cầu của khách hàng và nhà hàng.

CHƯƠNG 2. PHÂN TÍCH CÁC TRƯỜNG HỢP SỬ DỤNG (USE CASE)

Trong chương này, nhóm 10 sẽ tiến hành phân tích các trường hợp sử dụng (use case) của hệ thống. Mỗi use case sẽ được mô tả chi tiết bao gồm tên use case, các bên liên quan (actors), mô tả, tiền điều kiện (preconditions) và kết quả (postconditions). Trước khi đi vào chi tiết từng use case, hình ảnh tổng quan dưới đây sẽ cung cấp một cái nhìn khái quát về tất cả các use case trong hệ thống:



2.1. Use Case chung (đăng nhập, đăng ký, xem chi tiết đơn hàng)

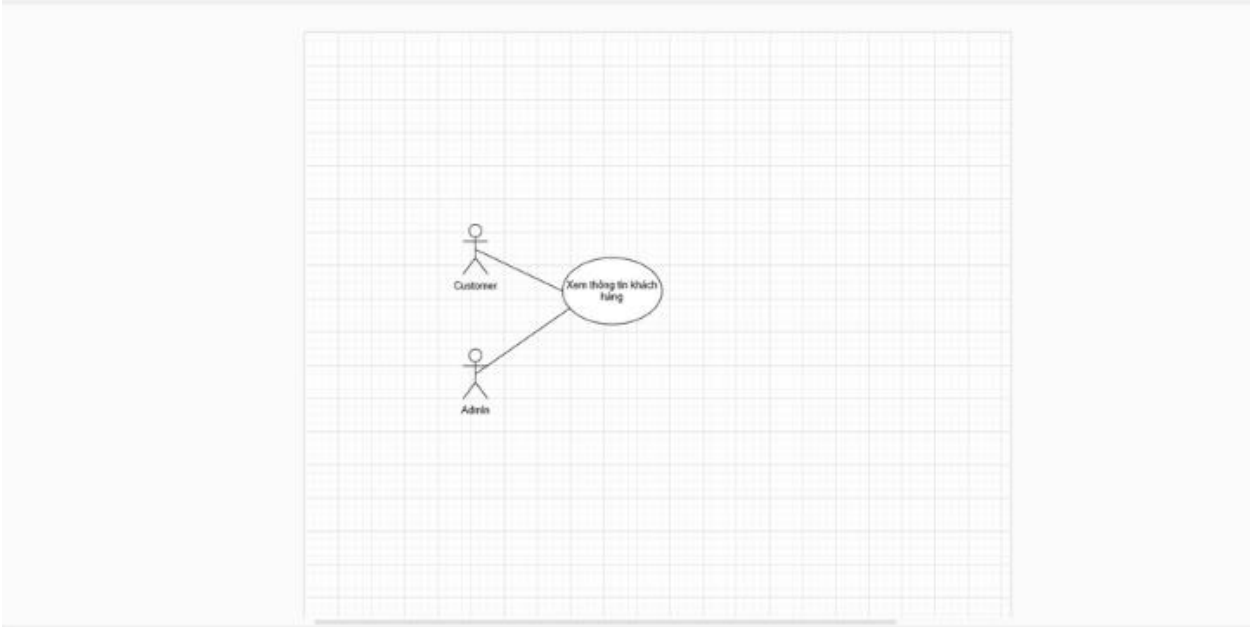


Tên use case	Chung
Bên liên quan (actor)	Admin, Customer, Staff
Mô tả	<div><div>- Đối với Đăng nhập: Khách hàng đăng nhập vào hệ thống bằng tài khoản đã đăng ký, admin, nhân viên đăng nhập bằng tài khoản được tạo. Các bước thực hiện:</div><div><div>• Người dùng nhập tên đăng nhập và mật khẩu vào form đăng nhập.</div><div>• Hệ thống xác thực thông tin đăng nhập.</div><div>• Nếu thông tin chính xác, người dùng được đăng nhập thành công.</div><div>• Nếu thông tin sai, hệ thống báo lỗi và yêu cầu nhập lại.</div></div><div>- Đối với Đăng xuất: Người dùng đăng xuất khỏi hệ thống. Các bước thực hiện:</div><div><div>• Người dùng nhấn vào nút đăng xuất.</div><div>• Hệ thống thực hiện quy trình đăng xuất và trả về trang chủ.</div></div><div>- Đối với Xem chi tiết đơn hàng: Người dùng theo dõi chi tiết đơn hàng đã đặt. Các bước thực hiện:</div><div><div>• Người dùng vào trang chi tiết đơn hàng.</div><div>• Hệ thống hiển thị chi tiết đơn hàng.</div></div></div>

Tiền điều kiện (Preconditions)	<ul style="list-style-type: none">- Đối với Đăng nhập: Khách hàng, admin, nhân viên đã có tài khoản.- Đối với Đăng xuất: Người dùng đã đăng nhập vào hệ thống.- Đối với Xem chi tiết đơn hàng: Đơn hàng đã được đặt
Kết quả (Postconditions)	<ul style="list-style-type: none">- Đối với Đăng nhập:<ul style="list-style-type: none">• Khách hàng, admin, nhân viên được xác thực và đăng nhập thành công, chuyển tới giao diện của vai trò.• Nếu thông tin đăng nhập sai, hệ thống báo lỗi và yêu cầu nhập lại.- Đối với Đăng xuất: Người dùng được đăng xuất khỏi hệ thống.- Đối với Xem chi tiết đơn hàng: Chi tiết đơn hàng được hiển thị.

2.2. Khách Hàng

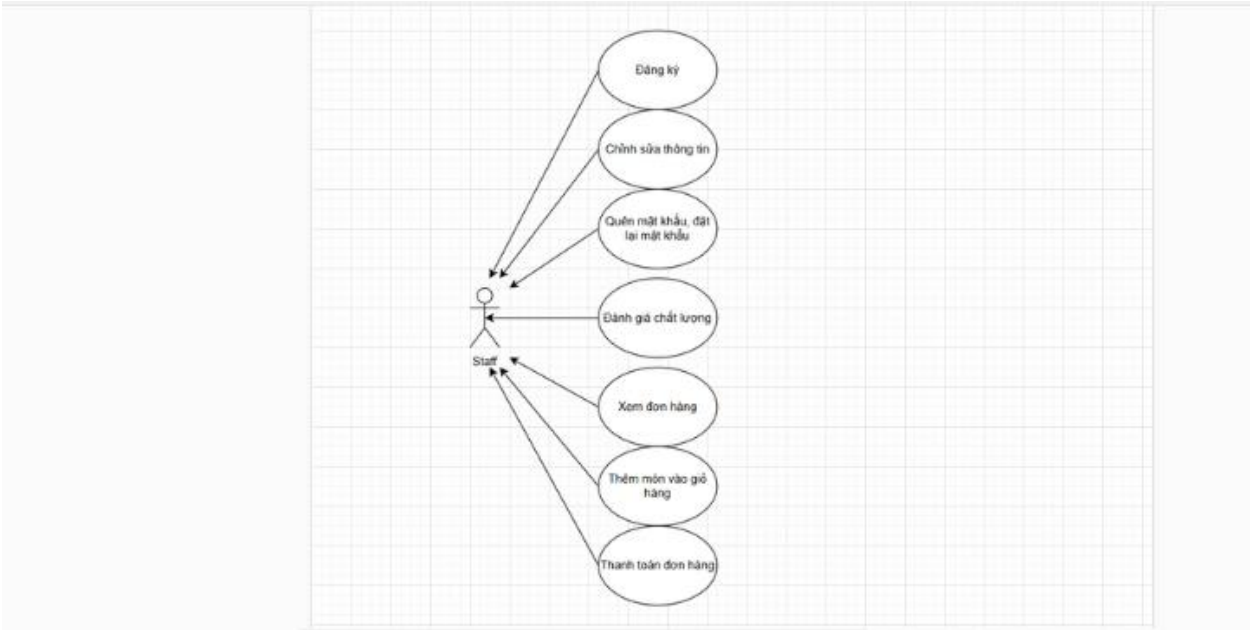
2.2.1. Xem thông tin khách hàng



Tên use case	Xem thông tin khách hàng
Bên liên quan (actor)	Admin, Customer

Mô tả	Admin và Customer được truy cập vào trang thông tin của các user trong hệ thống. Với customer chỉ là thông tin của chính customer đó.
Tiền điều kiện (Preconditions)	Đăng nhập dưới quyền admin hoặc customer
Kết quả (Postconditions)	Hiển thị thông tin user tương ứng với vai trò của người đăng nhập

2.2.2. Khách Hàng:

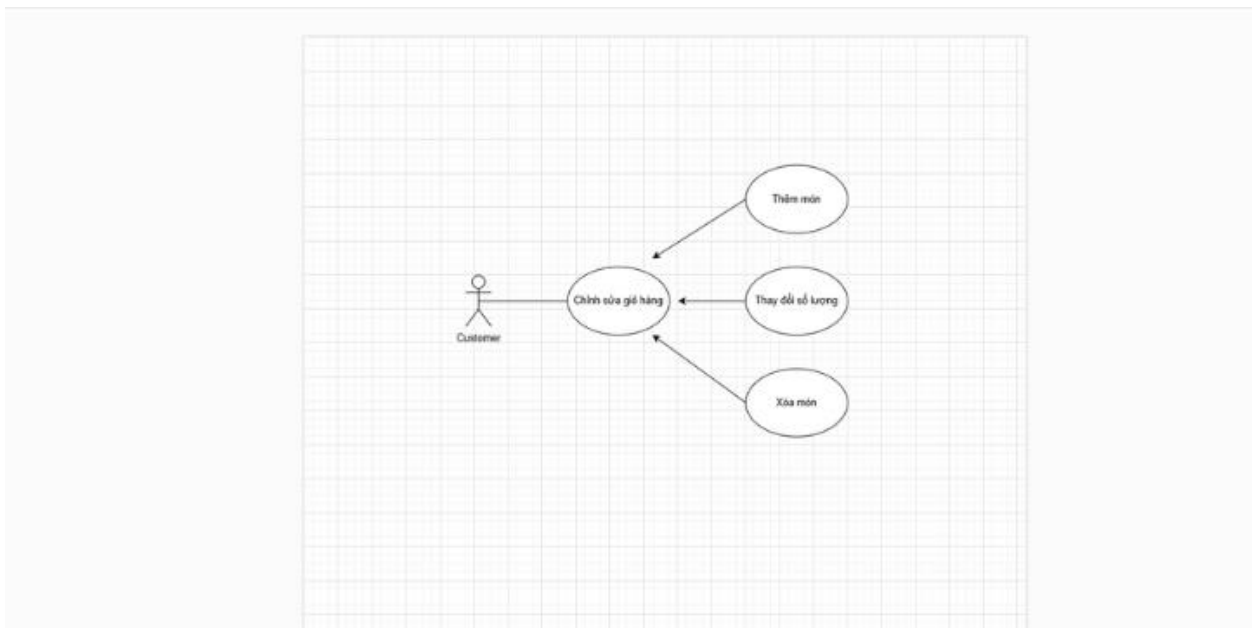


Tên use case	Khách Hàng
Bên liên quan (actor)	Customer
Mô tả	<ul style="list-style-type: none">- Đối với Đăng ký: Use case này mô tả người dùng muốn đăng ký tài khoản Customer trên hệ thống. Các bước thực hiện:<ul style="list-style-type: none">• Customer nhấn vào đăng ký tài khoản, trang web hệ thống trả về form đăng ký.• Form đăng ký yêu cầu nhập các thông tin: name, username, password, email, phone number.• Nếu đăng ký thành công, hệ thống sẽ báo đăng ký tài khoản thành công.• Nếu đăng ký thất bại, hệ thống sẽ báo đăng ký thất bại.- Đối với Chỉnh sửa thông tin: Cho phép customer xem thông tin của bản thân và thực hiện chỉnh sửa. Các bước thực hiện:

	<ul style="list-style-type: none"> • Customer nhấn vào my profile để mở thông tin người dùng • Thực hiện chỉnh sửa các thông tin như name, mail, số điện thoại, mật khẩu • Nếu thông tin được cập nhập hợp lệ hệ thống sẽ báo đã cập nhập thành công • Nếu không thì sẽ hiện thị lại thông tin khách hàng như ban đầu <p>- Đối với Quên mật khẩu và đặt lại mật khẩu: Khách hàng quên mật khẩu và yêu cầu đặt lại mật khẩu. Các bước thực hiện:</p> <ul style="list-style-type: none"> • Khách hàng nhấn vào liên kết "Quên mật khẩu". • Hệ thống yêu cầu khách hàng nhập email đã đăng ký. • Hệ thống gửi email chứa mật khẩu mới ngẫu nhiên. • Khách hàng dùng mật khẩu mới ngẫu nhiên để vào cập nhật mật khẩu mới theo ý muốn. • Hệ thống xác nhận và cập nhật mật khẩu mới. <p>- Đối với Đánh giá chất lượng: Khách hàng đánh giá và nhận xét về sản phẩm và dịch vụ sau khi sử dụng. Các bước thực hiện:</p> <ul style="list-style-type: none"> • Khách hàng vào trang chủ. • Khách hàng nhập đánh giá và nhận xét. • Khách hàng xác nhận và gửi đánh giá. <p>- Đối với Xem đơn hàng: Khách hàng theo dõi trạng thái của đơn hàng đã đặt. Các bước thực hiện:</p> <ul style="list-style-type: none"> • Khách hàng vào trang cá nhân. • Hệ thống hiển thị trạng thái đơn hàng (đang xử lý, đã hoàn thành). <p>- Đối với Thêm món vào giỏ hàng: Khách hàng thêm món ăn hoặc thức uống vào giỏ hàng. Các bước thực hiện:</p> <ul style="list-style-type: none"> • Khách hàng chọn món ăn hoặc thức uống muốn thêm vào giỏ hàng. • Khách hàng nhấn vào nút "Thêm vào giỏ hàng". • Hệ thống thêm món ăn hoặc thức uống vào giỏ hàng của khách hàng. <p>- Đối với Thanh toán đơn hàng: Khách hàng chọn Thanh toán cho đơn hàng. Các bước thực hiện:</p> <ul style="list-style-type: none"> • Khách hàng giỏ hàng. • Khách hàng vào trang thanh toán. • Khách hàng tiến hành thanh toán.
--	--

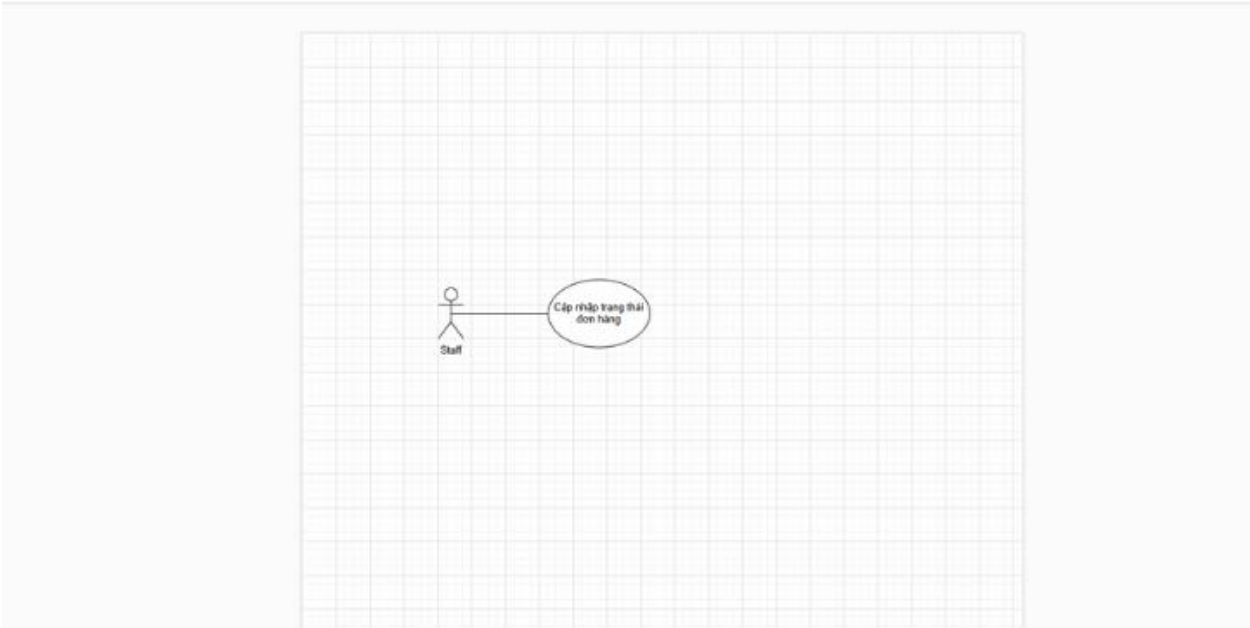
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập với quyền là customer ngoại trừ đối với Đăng ký thì customer phải chưa có tài khoản trước đó
Kết quả (Postconditions)	<ul style="list-style-type: none"> - Đối với Đăng ký: <ul style="list-style-type: none"> • Nếu đăng ký thành công, hệ thống báo đăng ký thành công. Customer có thể sử dụng tài khoản vừa đăng ký để đăng nhập. • Nếu đăng ký thất bại, hệ thống báo đăng ký thất bại. Yêu cầu Customer kiểm tra lại thông tin. Nguyên nhân thất bại có thể trùng với unique field trong cơ sở dữ liệu như: username, email. - Đối với Chỉnh sửa thông tin: <ul style="list-style-type: none"> • Nếu thông tin hợp lệ, hệ thống báo đã cập nhật thành công. • Nếu thông tin không hợp lệ, hệ thống yêu cầu xem lại gmail, mật khẩu có hợp lệ hay không - Đối với Quên mật khẩu và đặt lại mật khẩu: Mật khẩu mới được gửi đến khách hàng và tài khoản được cập nhật. - Đối với Đánh giá chất lượng: Đánh giá và nhận xét được lưu lại và hiển thị. - Đối với Xem đơn hàng: Trạng thái đơn hàng được hiển thị. - Đối với Thêm món vào giỏ hàng: Món ăn hoặc thức uống được thêm vào giỏ hàng của khách hàng. - Đối với Thanh toán đơn hàng: Đơn hàng được xác nhận và đặt hàng thành công hoặc thất bại.

2.2.3. Chỉnh sửa giỏ hàng:



Tên use case	Chỉnh sửa giỏ hàng
Bên liên quan (actor)	Customer
Mô tả	<ul style="list-style-type: none"> - Đối với Thêm món: Khách hàng thêm món ăn hoặc thức uống vào giỏ hàng. Các bước thực hiện: <ul style="list-style-type: none"> • Khách hàng chọn món ăn hoặc thức uống muốn thêm vào giỏ hàng. • Khách hàng nhấn vào nút "Thêm vào giỏ hàng". • Hệ thống thêm món ăn hoặc thức uống vào giỏ hàng của khách hàng. - Đối với Thay đổi số lượng: cho phép customer thay đổi số lượng món ăn của các món có trong giỏ hàng. Các bước thực hiện: <ul style="list-style-type: none"> • Actor nhấn vào các button “+” hoặc “-” với món ăn tương ứng • Nếu món ăn có số lượng = 0 thì món đó sẽ bị xóa khỏi giỏ hàng • Hệ thống lưu giỏ hàng hiện lên tương ứng với số lượng khách hàng đã chỉnh sửa - Đối với Xóa món ăn: Khách hàng xóa món ăn hoặc thức uống khỏi giỏ hàng. Các bước thực hiện: <ul style="list-style-type: none"> • Khách hàng vào giỏ hàng. • Khách hàng chọn món ăn hoặc thức uống muốn xóa. • Khách hàng nhấn vào nút "Xóa". • Hệ thống xóa món ăn hoặc thức uống khỏi giỏ hàng của khách hàng.
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập với quyền là customer
Kết quả (Postconditions)	<ul style="list-style-type: none"> - Đối với Thêm món: Món ăn hoặc thức uống được thêm vào giỏ hàng của khách hàng. - Đối với Thay đổi số lượng: Giỏ hàng có số lượng tương ứng. - Đối với Xóa món: Món ăn hoặc thức uống được xóa khỏi giỏ hàng của khách hàng.

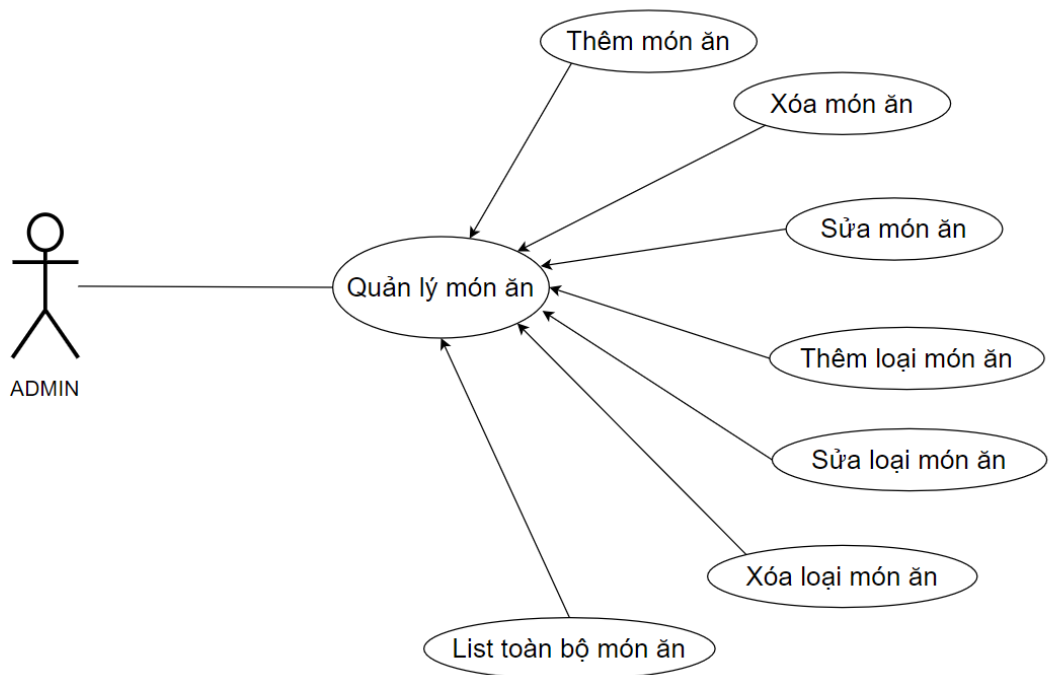
2.2.4. Cập nhật trạng thái đơn hàng:



Tên use case	Cập nhật trạng thái đơn hàng
Bên liên quan (actor)	Staff
Mô tả	<div>- Nhân viên cập nhật trạng thái đơn hàng sau khi đã xử lý (đã giao hàng hoàn thành). Các bước thực hiện:<ul style="list-style-type: none">• Nhân viên đăng nhập vào hệ thống.• Nhân viên vào trang danh sách đơn hàng.• Nhân viên chọn đơn hàng đã xử lý và cập nhật trạng thái.• Hệ thống cập nhật trạng thái đơn hàng.</div>
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập dưới quyền staff và phải có đơn hàng đang chờ xác nhận
Kết quả (Postconditions)	<div>- Trạng thái đơn hàng được cập nhật trong hệ thống.</div>

2.3. Admin

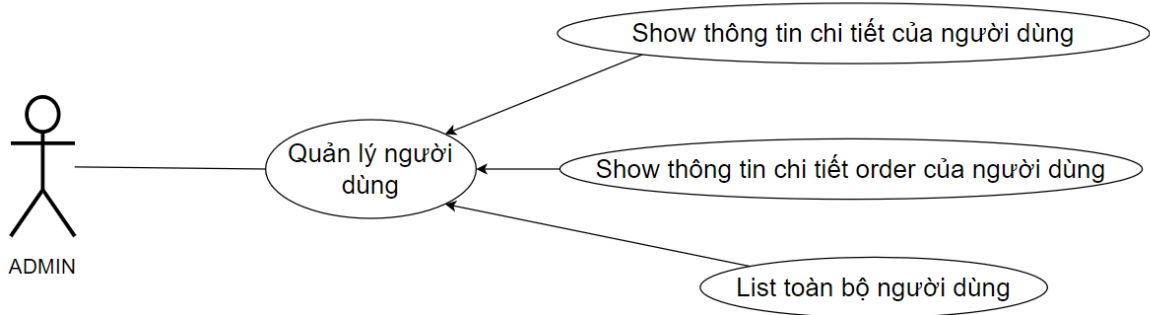
2.3.1. Quản lý món ăn



Tên use case	Quản lý món ăn
Bên liên quan (actor)	Admin
Mô tả	<div><div>- Đối với List toàn bộ món ăn: cho phép admin xem toàn bộ các món ăn đã phân theo trang. Các bước thực hiện<ul style="list-style-type: none">• Actor nhấn vào mục “Product” trên thanh taskbar• Hệ thống trả về giao diện món ăn</div><div>- Đối với Sửa món ăn: cho phép admin sửa thông tin của món ăn đã chỉ định. Các bước thực hiện<ul style="list-style-type: none">• Actor nhấn vào mục “Detail” trên thanh trong bảng món ăn• Hệ thống trả về giao diện chi tiết món ăn• Actor chỉnh sửa thông tin món ăn trong form• Actor nhấn nút “save” để lưu• Hệ thống xử lí thông tin và lưu vào trong database</div><div>- Đối với Thêm món ăn: cho phép admin thêm một món ăn. Các bước thực hiện<ul style="list-style-type: none">• Actor nhấn vào mục “Add product”• Hệ thống trả về giao diện thêm món ăn• Actor nhập thông tin món ăn theo form• Actor nhấn nút “Add” để lưu• Hệ thống xử lí và thêm món ăn</div></div>

	<ul style="list-style-type: none"> - Đối với Xóa món ăn: cho phép admin xóa món ăn đã chỉ định. Các bước thực hiện <ul style="list-style-type: none"> • Actor nhấn vào mục delete trong bảng món ăn • Hệ thống hỏi actor có muốn xóa hay không • Nếu xóa thì hệ thống xử lý và xóa món ăn - Đối với Thêm một loại món ăn: cho phép admin thêm một loại món ăn mới. Các bước thực hiện <ul style="list-style-type: none"> • Actor thực hiện chức năng thêm món ăn • Nếu loại món ăn là món ăn mới thì hệ thống sẽ tự động thêm loại món ăn đó. - Đối với Sửa một loại món ăn: cho phép admin sửa thông tin của món ăn bất kì. Các bước thực hiện <ul style="list-style-type: none"> • Actor nhấn vào mục detail trong bảng category • Actor chỉnh sửa thông tin theo form • Actor nhấn save để lưu • Hệ thống xử lý và sửa loại món ăn - Đối với Xóa loại món ăn: cho phép admin xóa một loại món ăn bất kì. Các bước thực hiện <ul style="list-style-type: none"> • Actor nhấn vào mục “Delete” trong bảng category • Hệ thống hỏi actor có muốn xóa hay không • Nếu có thì hệ thống xử lý và xóa loại món ăn
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập với quyền là admin
Kết quả (Postconditions)	<ul style="list-style-type: none"> - Đối với List toàn bộ món ăn: Hiển thị danh sách và thông tin của các món ăn trong bảng - Đối với Thêm món ăn: Nếu hợp lệ thì thêm món ăn thành công, nếu xảy ra lỗi thì in lỗi ra giao diện - Đối với Sửa món ăn: Nếu hợp lệ thì sửa món ăn thành công, nếu xảy ra lỗi thì in lỗi ra giao diện - Đối với Xóa món ăn: Nếu hợp lệ thì xóa món ăn thành công, nếu xảy ra lỗi thì in lỗi ra giao diện - Đối với Thêm loại món ăn: Nếu hợp lệ thì thêm loại món ăn thành công, nếu xảy ra lỗi thì in lỗi ra giao diện - Đối với Sửa loại món ăn: Nếu hợp lệ thì sửa loại món ăn thành công, nếu xảy ra lỗi thì in lỗi ra giao diện - Đối với Xóa loại món ăn: Nếu hợp lệ thì xóa loại món ăn thành công, nếu xảy ra lỗi thì in lỗi ra giao diện

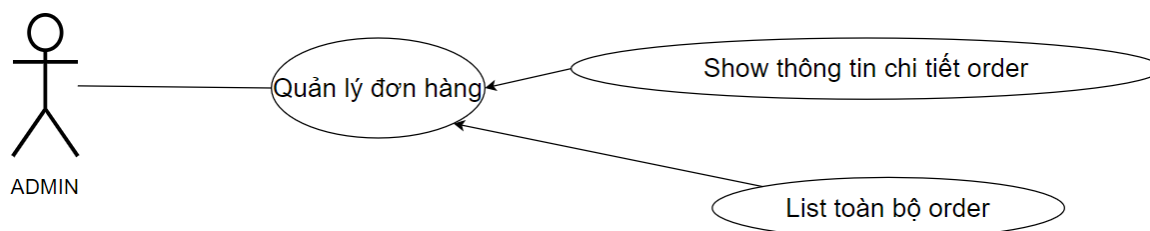
2.3.2. Quản lý người dùng



Tên use case	Quản lý người dùng
Bên liên quan (actor)	Admin
Mô tả	<ul style="list-style-type: none">- Đối với List toàn bộ người dùng: cho phép admin xem toàn bộ người dùng đã phân theo trang. Các bước thực hiện<ul style="list-style-type: none">• Actor nhấn vào mục “Customer” trên thanh taskbar• Hệ thống trả về giao diện người dùng- Đối với Show thông tin chi tiết của người dùng: cho phép admin xem thông tin của người dùng đã chỉ định. Các bước thực hiện<ul style="list-style-type: none">• Actor nhấn vào mục “Detail” trên thanh trong bảng người dùng• Hệ thống trả về giao diện chi tiết thông tin người dùng- Đối với Show thông tin chi tiết order món ăn: cho phép admin xem thông tin chi tiết một order của người dùng. Các bước thực hiện<ul style="list-style-type: none">• Actor nhấn vào mục “Detail” trong giao diện chi tiết người dùng• Hệ thống trả về giao diện chi tiết order
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập với quyền là admin
Kết quả (Postconditions)	<ul style="list-style-type: none">- Đối với List toàn bộ món ăn: Hiển thị danh sách và thông tin của người dùng trong bảng- Đối với Show thông tin chi tiết của người dùng: Hiển thị thông tin chi tiết của người dùng

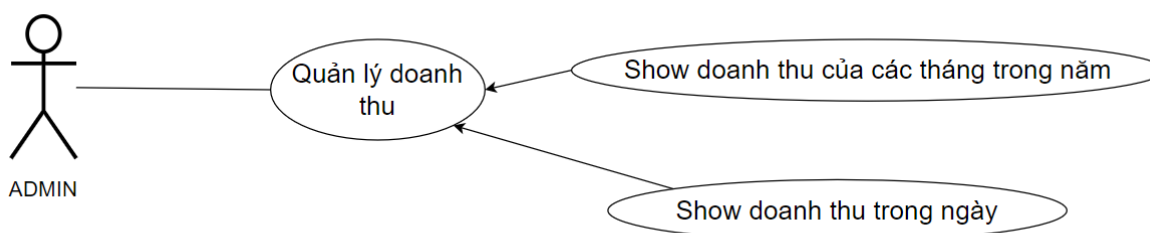
	- Đối với Show thông tin chi tiết order món ăn: Hiển thị thông tin chi tiết order của người dùng
--	---

2.3.3. Quản lý đơn hàng



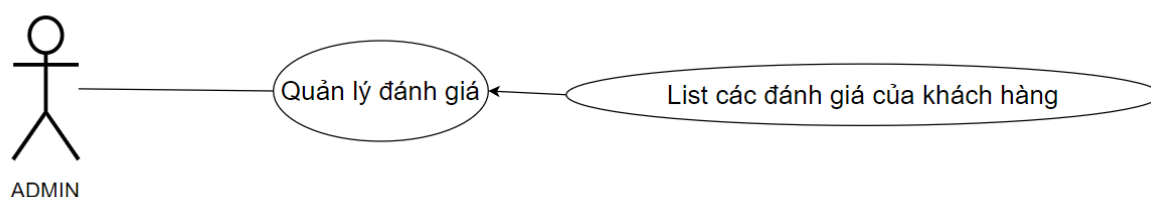
Tên use case	Quản lý đơn hàng
Bên liên quan (actor)	Admin
Mô tả	<ul style="list-style-type: none"> - Đối với List toàn bộ order: cho phép admin xem toàn bộ order đã phân theo trang. Các bước thực hiện <ul style="list-style-type: none"> • Actor nhấn vào mục “Order” trên thanh taskbar • Hệ thống trả về giao diện đơn hàng - Đối với Show thông tin chi tiết của order: cho phép admin xem thông tin của người dùng đã chỉ định. Các bước thực hiện <ul style="list-style-type: none"> • Actor nhấn vào mục “Detail” trên thanh trong bảng order • Hệ thống trả về giao diện chi tiết chi tiết đơn hàng
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập với quyền là admin
Kết quả (Postconditions)	<ul style="list-style-type: none"> - Đối với List toàn bộ order: Hiển thị danh sách và thông tin của đơn hàng trong bảng - Đối với Show thông tin chi tiết của order: Hiển thị thông tin chi tiết của đơn hàng

2.3.5. Quản lý doanh thu



Tên use case	Quản lý doanh thu
Bên liên quan (actor)	Admin
Mô tả	<ul style="list-style-type: none"> - Đối với Show doanh thu của các tháng trong năm: cho phép admin xem doanh thu của các tháng trong năm. (Tự động) - Đối với Show doanh thu trong ngày: cho phép admin xem doanh thu trong ngày. (Tự động)
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập với quyền là admin
Kết quả (Postconditions)	<ul style="list-style-type: none"> - Đối với Show doanh thu của các tháng trong năm: Tính toán và hiển thị doanh thu của các tháng trong năm. - Đối với Show doanh thu trong ngày: Tính toán và hiển thị doanh thu của trong ngày.

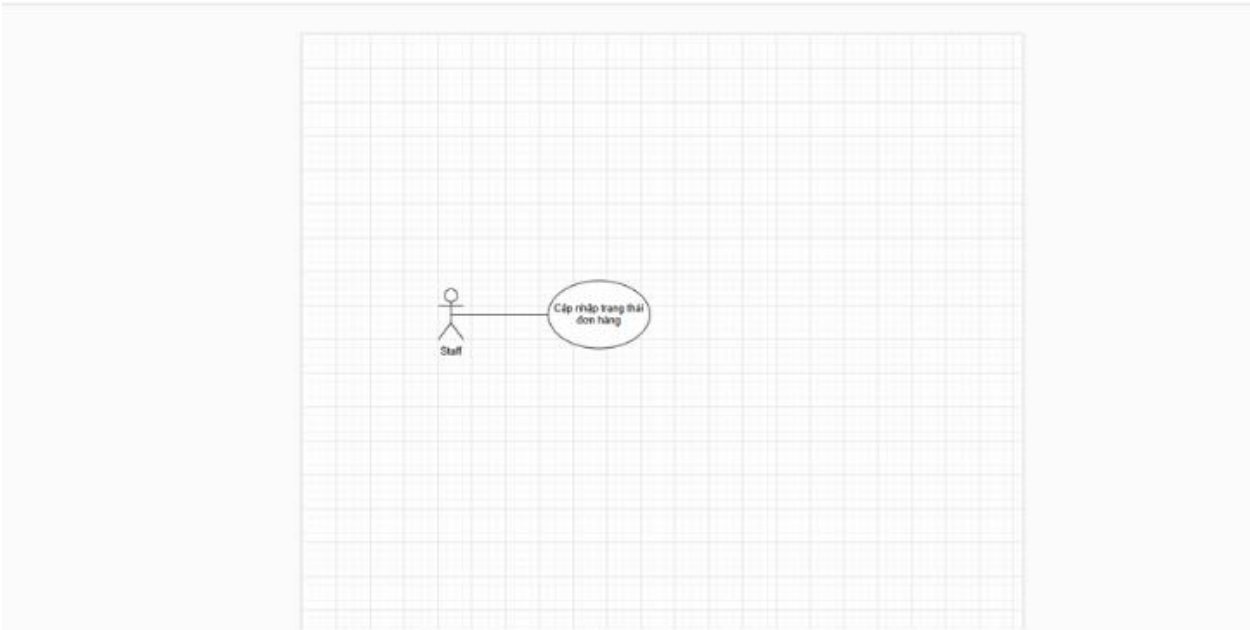
2.3.6. Quản lý đánh giá



Tên use case	Quản lý doanh thu
Bên liên quan (actor)	Admin
Mô tả	<ul style="list-style-type: none"> - Đối với List các đánh giá của khách hàng: cho phép admin xem các đánh giá của khách hàng. Các bước thực hiện <ul style="list-style-type: none"> • Actor nhấn vào mục “Review” trên thanh taskbar • Hệ thống trả về giao diện đánh giá
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập với quyền là admin
Kết quả (Postconditions)	<ul style="list-style-type: none"> - Đối với List các đánh giá của khách hàng: Hiển thị các review của khách hàng.

2.4. Nhân viên

2.4.1. Cập nhật trạng thái đơn hàng



Tên use case	Cập nhật trạng thái đơn hàng
Bên liên quan (actor)	Staff
Mô tả	<div>- Nhân viên cập nhật trạng thái đơn hàng sau khi đã xử lý (đã giao hàng hoàn thành). Các bước thực hiện:<ul style="list-style-type: none">• Nhân viên đăng nhập vào hệ thống.• Nhân viên vào trang danh sách đơn hàng.• Nhân viên chọn đơn hàng đã xử lý và cập nhật trạng thái.• Hệ thống cập nhật trạng thái đơn hàng.</div>
Tiền điều kiện (Preconditions)	Actor cần phải đăng nhập dưới quyền staff và phải có đơn hàng đang chờ xác nhận
Kết quả (Postconditions)	<div>- Trạng thái đơn hàng được cập nhật trong hệ thống.</div>

CHƯƠNG 3. THIẾT KẾ DỮ LIỆU

3.1. Tổng quan

3.1.1. Mục tiêu

Mục tiêu của phần này là thiết kế một cơ sở dữ liệu cho nền tảng đặt đồ ăn trực tuyến nhằm đảm bảo tính toàn vẹn, an toàn và hiệu quả trong việc lưu trữ, quản lý và truy xuất dữ liệu. Thiết kế cơ sở dữ liệu sẽ hỗ trợ việc tổ chức thông tin khách hàng, quản trị viên, món ăn, đơn hàng và các bình luận đánh giá, tạo nền tảng cho việc phát triển các tính năng ứng dụng một cách mạch lạc và tối ưu hóa hiệu suất hệ thống. Cụ thể, mục tiêu thiết kế dữ liệu bao gồm:

- *Tối ưu hóa truy vấn dữ liệu:* Đảm bảo các truy vấn dữ liệu liên quan đến khách hàng, món ăn, đơn hàng và bình luận được thực thi một cách nhanh chóng và hiệu quả.
- *Đảm bảo tính toàn vẹn dữ liệu:* Thiết lập các ràng buộc và quy tắc nhằm đảm bảo dữ liệu được lưu trữ một cách nhất quán và chính xác.
- *Hỗ trợ mở rộng và bảo trì hệ thống:* Thiết kế cấu trúc dữ liệu linh hoạt, dễ dàng mở rộng và bảo trì, phù hợp với các yêu cầu kinh doanh hiện tại và tương lai.
- *Bảo mật dữ liệu:* Xây dựng các biện pháp bảo mật để bảo vệ dữ liệu nhạy cảm của khách hàng và nhà hàng, ngăn chặn các truy cập trái phép.

3.1.2. Phạm vi và mô tả hệ thống

Dự án này sẽ bao gồm các phần sau:

1. *Quản lý khách hàng:* Tạo tài khoản, đăng nhập, cập nhật thông tin cá nhân và quản lý các đơn hàng của khách hàng.
2. *Quản lý quản trị viên:* Đăng nhập, quản lý thông tin nhà hàng và duyệt các đơn hàng.
3. *Quản lý món ăn:* Thêm, xóa, cập nhật thông tin món ăn và phân loại món ăn theo danh mục.
4. *Quản lý đơn hàng:* Quản lý thông tin đơn hàng, xác nhận trạng thái đơn hàng và tính tổng giá trị đơn hàng.
5. *Quản lý chi tiết đơn hàng:* Lưu trữ chi tiết các món ăn trong từng đơn hàng.
6. *Quản lý bình luận:* Khách hàng có thể để lại bình luận và đánh giá về các món ăn.

Hệ thống sẽ được xây dựng dựa trên một cơ sở dữ liệu quan hệ, sử dụng SQL Server để quản lý và lưu trữ dữ liệu. Dữ liệu được tổ chức thành các bảng sau: Khách hàng (Customers), Quản trị viên (Admins), Món ăn (Foods), Danh mục món ăn (Categories), Đơn hàng (Orders), Chi tiết đơn hàng (Order_Details) và Bình luận (Comments). Mỗi bảng sẽ chứa các thông tin cần thiết và được thiết kế để đảm bảo tính toàn vẹn dữ liệu và hiệu suất truy vấn cao.

Chi tiết thiết kế cơ sở dữ liệu :

1. *Bảng Khách hàng (Customers)*: Lưu trữ thông tin về khách hàng bao gồm tên người dùng, số điện thoại, tên, email và mật khẩu.
2. *Bảng Quản trị viên (Admins)*: Lưu trữ thông tin về quản trị viên bao gồm tên người dùng, số điện thoại, tên và mật khẩu.
3. *Bảng Món ăn (Foods)*: Lưu trữ thông tin về các món ăn bao gồm tên món ăn, đường dẫn hình ảnh, giá và mã danh mục.
4. *Bảng Danh mục món ăn (Categories)*: Lưu trữ thông tin về các danh mục món ăn.
5. *Bảng Đơn hàng (Orders)*: Lưu trữ thông tin về các đơn hàng bao gồm ngày đặt hàng, mã khách hàng, trạng thái đơn hàng và tổng giá trị đơn hàng.
6. *Bảng Chi tiết đơn hàng (Order_Details)*: Lưu trữ chi tiết các món ăn trong từng đơn hàng bao gồm mã món ăn, mã đơn hàng, số lượng và giá của từng món.
7. *Bảng Bình luận (Comments)*: Lưu trữ bình luận và đánh giá của khách hàng về các món ăn.

3.2. Mô tả dữ liệu

3.2.1. Loại dữ liệu

1. Dữ liệu định lượng:

Số lượng món ăn: Thông tin về số lượng các món ăn trong mỗi đơn hàng. Dữ liệu này giúp nhà hàng biết được số lượng từng món ăn mà khách hàng đã đặt.

Giá món ăn và tổng giá trị đơn hàng: Bao gồm giá của từng món ăn và tổng giá trị của đơn hàng. Đây là thông tin quan trọng để tính toán doanh thu và quản lý tài chính.

Đánh giá món ăn: Điểm đánh giá của khách hàng đối với các món ăn, thường ở dạng thang điểm từ 1 đến 5. Dữ liệu này hỗ trợ việc đánh giá chất lượng món ăn và dịch vụ.

2. Dữ liệu định tính:

Tên và mô tả món ăn: Bao gồm tên món ăn và các mô tả liên quan, giúp khách hàng có thể hiểu rõ hơn về món ăn trước khi đặt hàng.

Bình luận của khách hàng: Nội dung bình luận về các món ăn từ khách hàng, cung cấp cái nhìn chi tiết về trải nghiệm của họ.

Thông tin cá nhân khách hàng: Bao gồm tên, số điện thoại, email và địa chỉ của khách hàng, được sử dụng để quản lý tài khoản và liên lạc.

3. Dữ liệu thời gian thực:

Thời gian đặt hàng: Thời gian cụ thể khi khách hàng đặt hàng. Dữ liệu này giúp quản lý đơn hàng theo thứ tự thời gian.

Trạng thái đơn hàng: Trạng thái cập nhật của đơn hàng theo thời gian thực như "Đang chuẩn bị", "Đã hoàn thành". Điều này giúp khách hàng và nhà hàng theo dõi tiến trình đơn hàng.

4. Dữ liệu lịch sử:

Lịch sử đặt hàng: Thông tin về các đơn hàng đã hoàn thành của khách hàng, giúp nhà hàng có thể phân tích thói quen mua sắm của khách hàng và cung cấp dịch vụ cá nhân hóa.

Lịch sử bình luận và đánh giá: Lưu trữ các bình luận và đánh giá trước đây của khách hàng, giúp nhà hàng theo dõi phản hồi và cải thiện chất lượng dịch vụ.

3.2.2. Nguồn dữ liệu

Dữ liệu trong dự án này được thu thập từ các nguồn sau:

Hệ thống quản lý khách hàng: Lưu trữ thông tin cá nhân và lịch sử đặt hàng của khách hàng. Đây là nguồn dữ liệu chính để quản lý các giao dịch và thông tin liên quan đến khách hàng.

Hệ thống quản lý đơn hàng: Chứa thông tin chi tiết về các đơn hàng, bao gồm danh sách món ăn, số lượng, giá cả, và trạng thái đơn hàng.

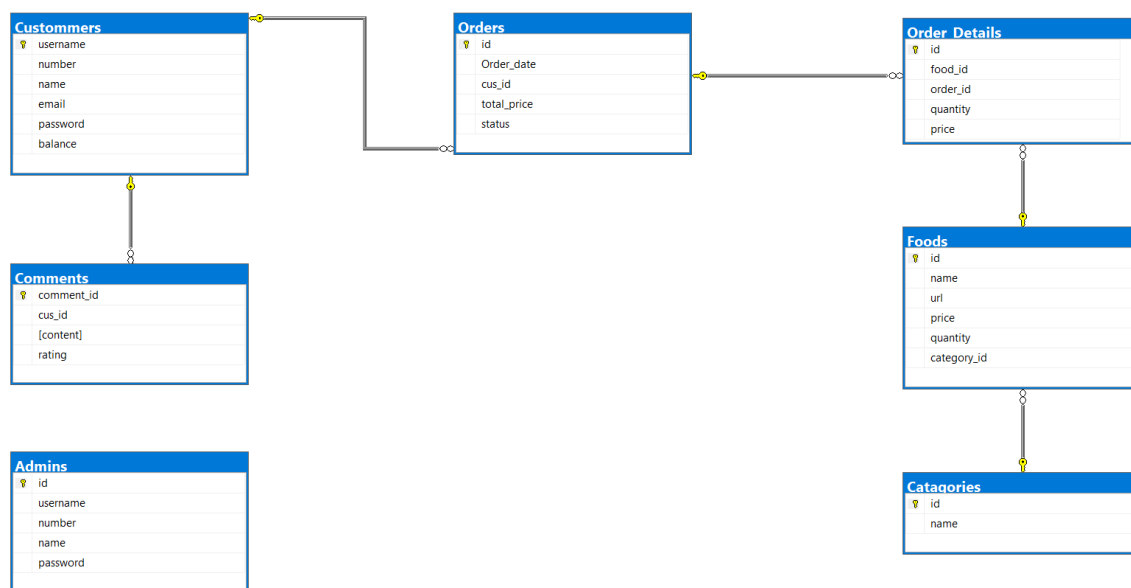
Hệ thống quản lý món ăn: Lưu trữ thông tin về các món ăn, bao gồm tên, giá cả, và danh mục. Dữ liệu này giúp nhà hàng quản lý thực đơn và cập nhật thông tin món ăn.

Log hệ thống: Ghi lại các hoạt động của người dùng trên nền tảng, bao gồm thời gian đăng nhập, thao tác đặt hàng, bình luận và đánh giá. Dữ liệu này giúp phân tích hành vi người dùng và tối ưu hóa trải nghiệm khách hàng.

3.3. Cấu trúc dữ liệu

3.3.1. Mô hình dữ liệu

Cấu trúc dữ liệu được thiết kế theo mô hình quan hệ, với các bảng dữ liệu chính được liên kết chặt chẽ thông qua các khóa chính và khóa ngoại. Điều này đảm bảo dữ liệu được tổ chức một cách logic, dễ dàng truy xuất và quản lý.



Hình 3.1

3.3.2. Bảng dữ liệu

Dưới đây là mô tả chi tiết về các bảng dữ liệu chính trong cơ sở dữ liệu của dự án, bao gồm tên bảng, các cột (thuộc tính) và loại dữ liệu của từng cột:

Bảng Customers (Khách hàng)

Tên cột	Loại dữ liệu	Mô tả
username	nvarchar(255)	Khóa chính, tên người dùng
number	nvarchar(10)	Số điện thoại của khách hàng
name	nvarchar(255)	Tên đầy đủ của khách hàng
email	nvarchar(255)	Địa chỉ email của khách hàng
password	nvarchar(255)	Mật khẩu của khách hàng

Bảng Admins (Quản trị viên)

Tên cột	Loại dữ liệu	Mô tả
username	nvarchar(255)	Khóa chính, tên người dùng
number	nvarchar(10)	Số điện thoại của quản trị viên
name	nvarchar(255)	Tên đầy đủ của quản trị viên
password	nvarchar(255)	Mật khẩu của quản trị viên

Bảng Foods (Món ăn)

Tên cột	Loại dữ liệu	Mô tả
id	integer	Khóa chính, mã món ăn
name	nvarchar(255)	Tên món ăn
url	nvarchar(255)	Đường dẫn hình ảnh của món ăn
price	money	Giá của món ăn
category_id	integer	Khóa ngoại, mã danh mục món ăn

Bảng Orders (Đơn hàng)

Tên cột	Loại dữ liệu	Mô tả
id	integer	Khóa chính, mã đơn hàng
Order_date	datetime	Ngày đặt hàng
cus_id	nvarchar(255)	Khóa ngoại, mã khách hàng
status	nvarchar(100)	Trạng thái của đơn hàng
total_price	money	Tổng giá trị của đơn hàng

Bảng Order_Details (Chi tiết đơn hàng)

Tên cột	Loại dữ liệu	Mô tả
id	integer	Khóa chính, mã chi tiết đơn hàng
food_id	integer	Khóa ngoại, mã món ăn
order_id	integer	Khóa ngoại, mã đơn hàng
quantity	integer	Số lượng món ăn trong đơn hàng
price	money	Giá của chi tiết đơn hàng

Bảng Comments (Bình luận)

Tên cột	Loại dữ liệu	Mô tả
comment_id	integer	Khóa chính, mã bình luận
cus_id	nvarchar(255)	Khóa ngoại, mã khách hàng
content	nvarchar(max)	Nội dung bình luận
rating	tinyint	Đánh giá của khách hàng (1-5)

3.3.3. Chi tiết kỹ thuật

• **Ràng buộc dữ liệu:**

Khóa chính (Primary Key): Mỗi bảng đều có một trường làm khóa chính, đảm bảo tính duy nhất của mỗi bản ghi. Ví dụ: username trong bảng Customers, id trong các bảng Foods, Orders, Order_Details, Comments.

Khóa ngoại (Foreign Key): Sử dụng để xác định mối quan hệ giữa các bảng. Ví dụ: category_id trong bảng Foods là khóa ngoại tham chiếu đến id trong bảng Categories, cus_id trong bảng Orders và Comments là khóa ngoại tham chiếu đến username trong bảng Customers.

Ràng buộc duy nhất (Unique Constraint): Đảm bảo giá trị của cột là duy nhất. Ví dụ: username trong bảng Customers, Admins, Employees là duy nhất để không có hai người dùng cùng tên đăng ký.

Ràng buộc không rỗng (Not Null Constraint): Đảm bảo các trường dữ liệu quan trọng không được để trống. Ví dụ: name, email, password trong bảng Customers, Admins, Employees không được phép null.

• **Chuẩn hóa dữ liệu:**

1NF (First Normal Form): Đảm bảo mỗi ô trong bảng chứa giá trị duy nhất và không chia sẻ.

2NF (Second Normal Form): Loại bỏ sự phụ thuộc của các trường non-key vào khóa chính.

3NF (Third Normal Form): Loại bỏ sự phụ thuộc của các trường non-key vào các trường khác non-key.

3.4. Kết luận

Trong phần thiết kế dữ liệu cho dự án nền tảng đặt đồ ăn cho nhà hàng, nhóm 10 đã trình bày mô hình dữ liệu quan hệ, các bảng dữ liệu chính, ràng buộc dữ liệu, chỉ mục và quy tắc chuẩn hóa. Việc áp dụng mô hình quan hệ và các chuẩn mực quản lý dữ liệu đã giúp cho dự án có cơ sở dữ liệu rõ ràng, an toàn và hiệu quả. Tuy nhiên, việc duy trì và cải thiện cơ sở dữ liệu trong tương lai vẫn cần sự chú ý và nỗ lực liên tục để đáp ứng được các yêu cầu ngày càng cao của người dùng và doanh nghiệp.

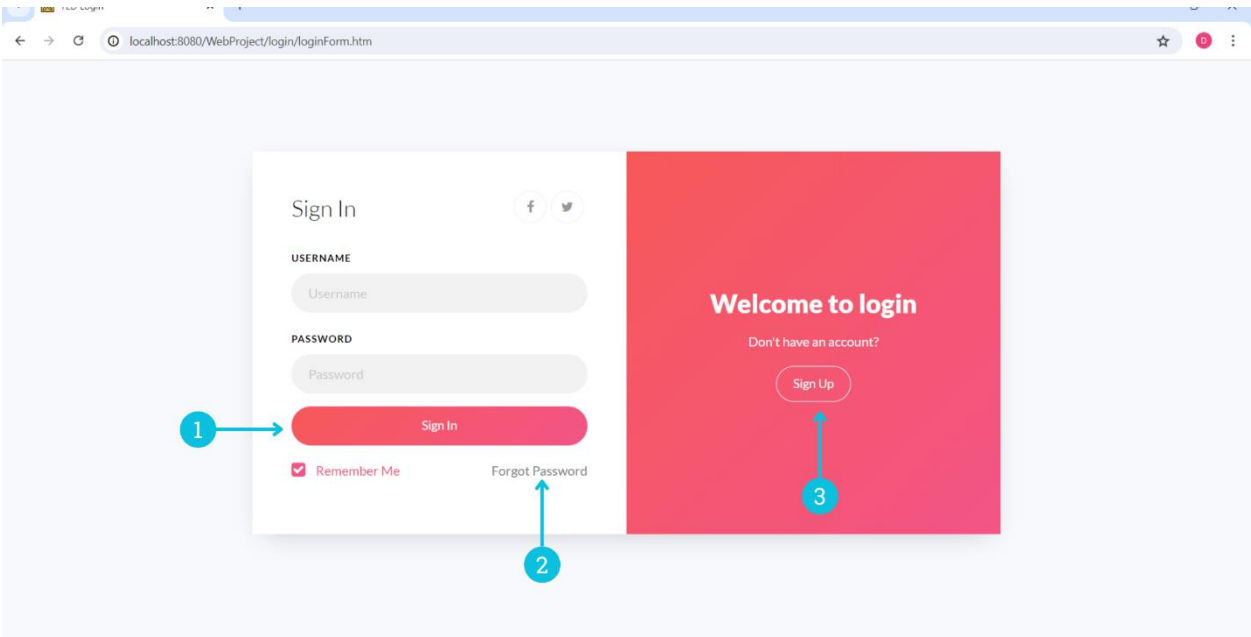
CHƯƠNG 4. THIẾT KẾ GIAO DIỆN

4.1. Customer

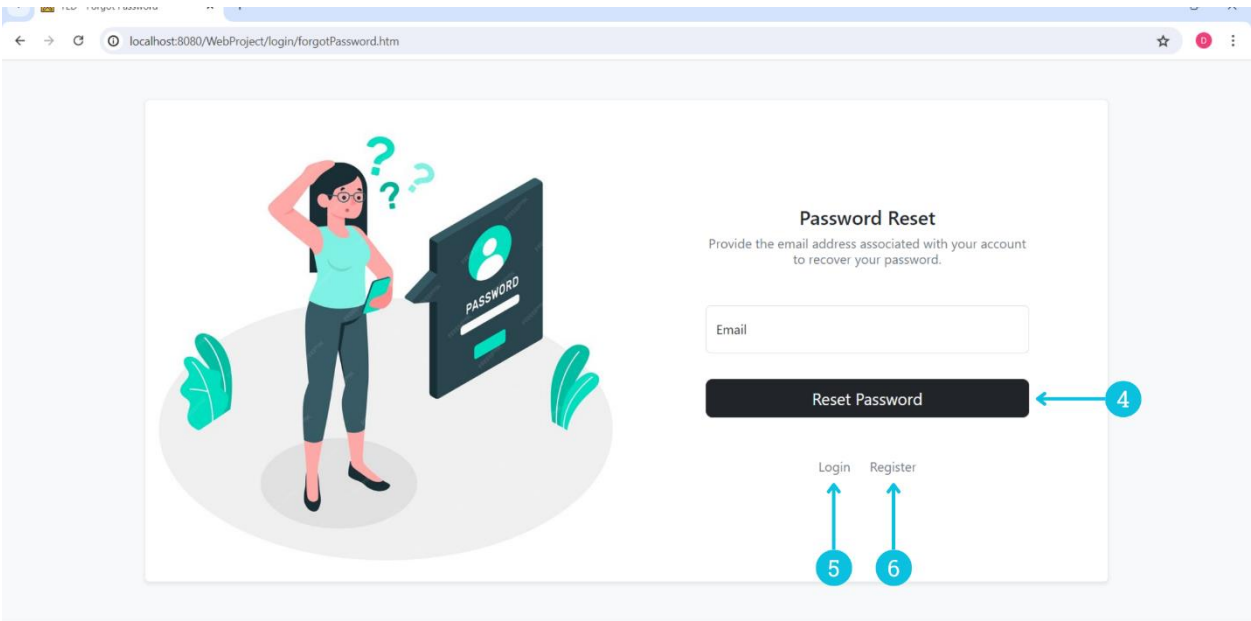
4.1.1. Thiết kế giao diện

4.1.1.1. Giao diện Login

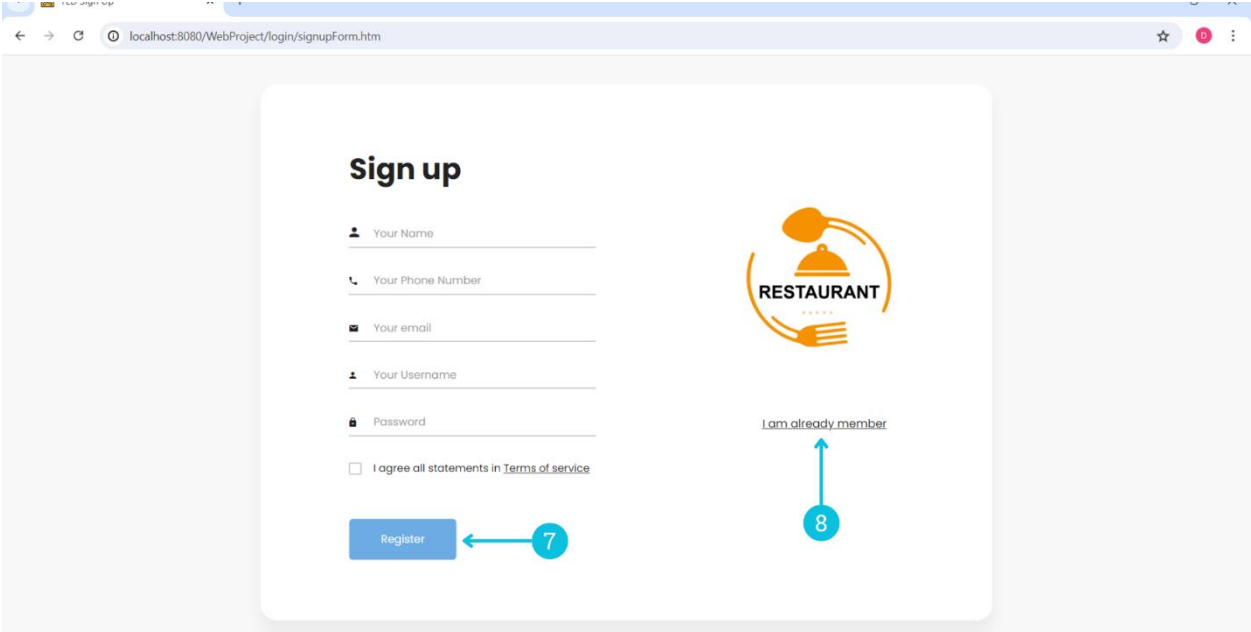
4.1.1.1.1. Form login



4.1.1.1.2. Form forgot password



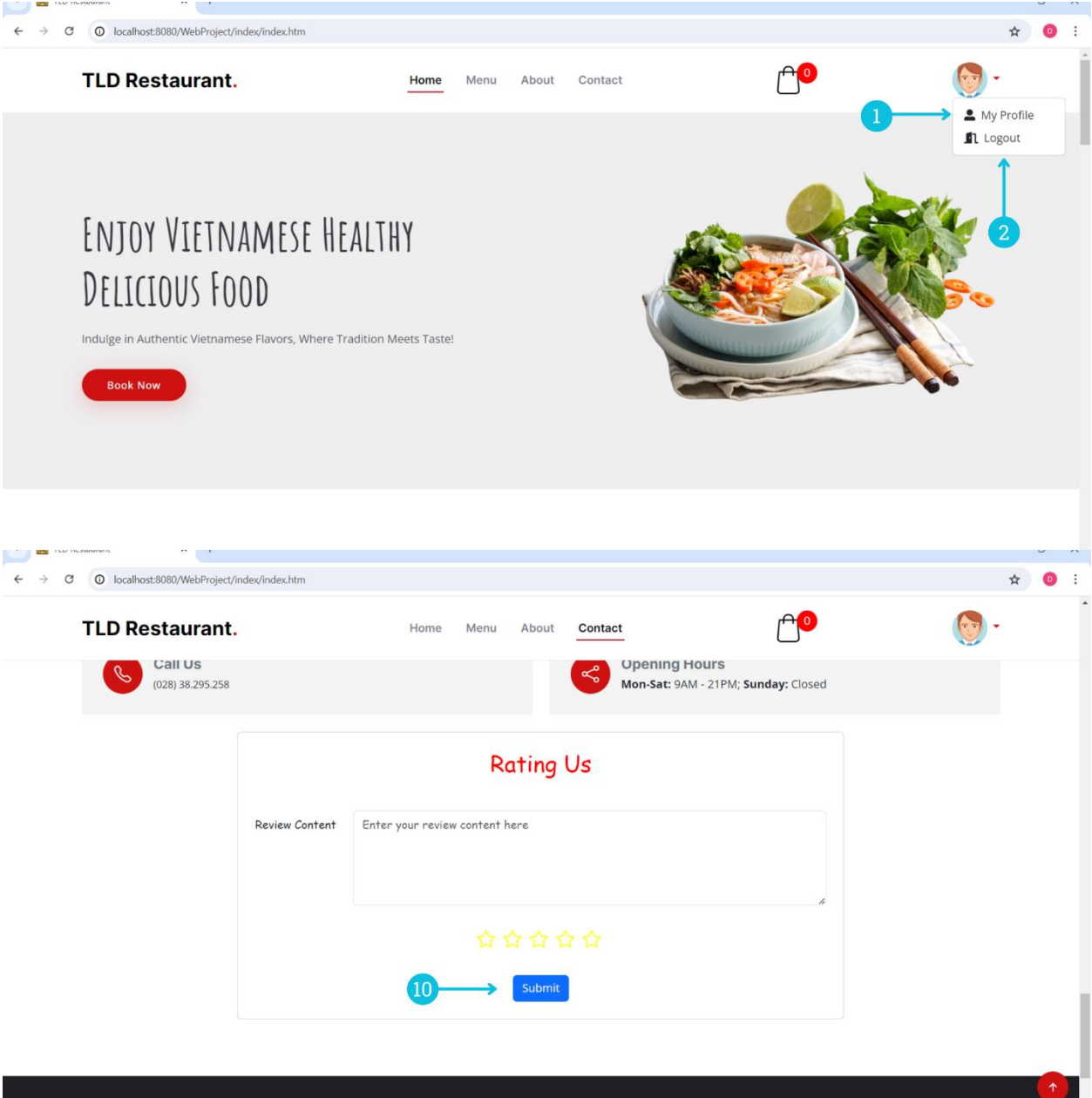
4.1.1.1.3. Form signup



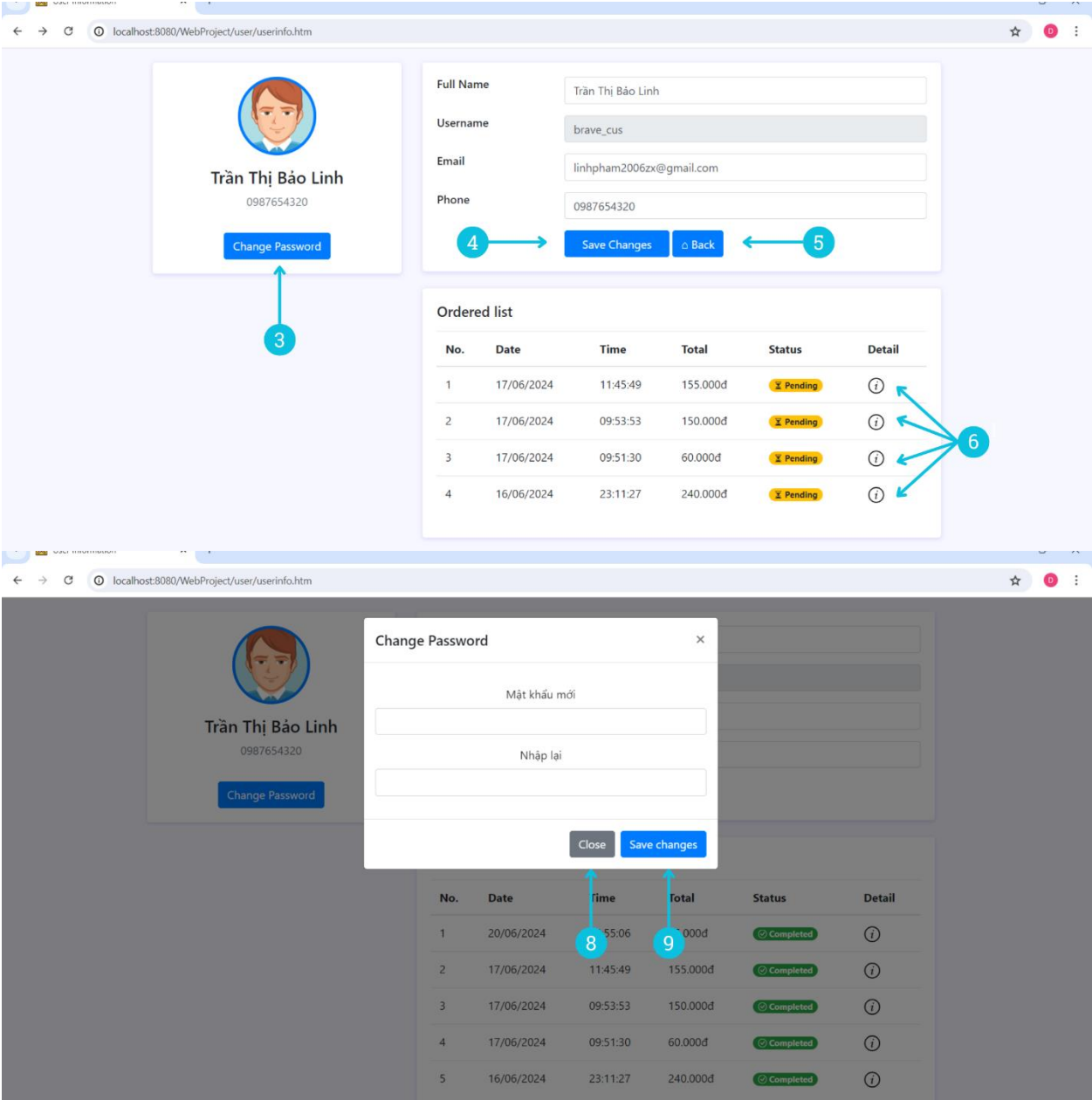
STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Đăng nhập vào trang web	Click vào button “Sign in”	signIn()
2	Chuyển sang giao diện quên mật khẩu	Click vào button “Forgot Password”	showForgotPassword()
3	Chuyển sang giao diện đăng ký	Click vào button “Sign up”	showSignUp()
4	Thực hiện gửi mail với mật khẩu mới	Click vào button “Reset Password”	forgotPassword()
5	Chuyển sang giao diện login	Click vào button “Login”	showLogin()
6	Chuyển sang giao diện đăng ký	Click vào button “Register”	showSignUp()
7	Đăng ký tài khoản khách hàng	Click vào button “Register”	signUp()
8	Chuyển sang giao diện login	Click vào button “Login”	showLogin()

4.1.1.2. Giao diện người dùng

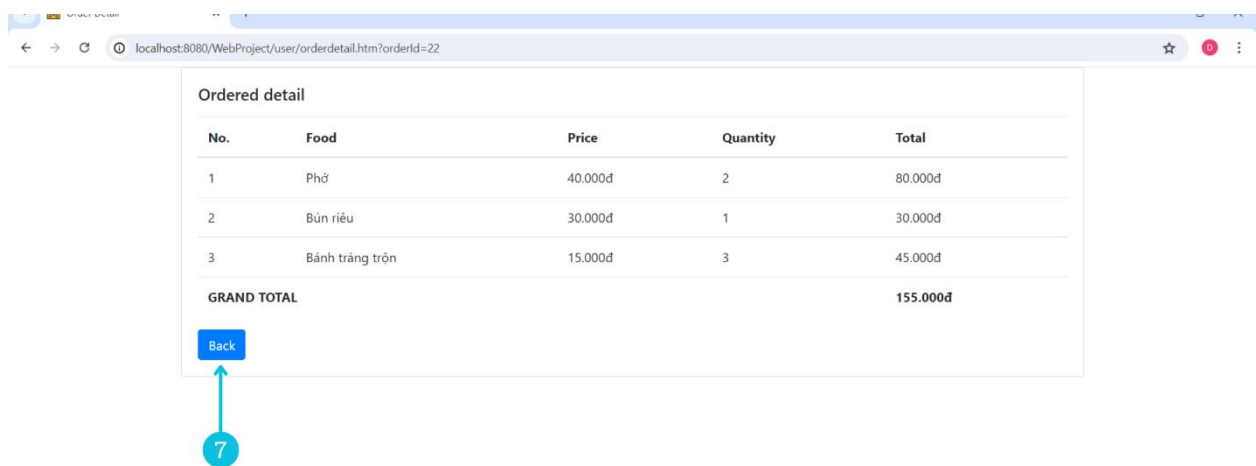
4.1.1.2.1. Giao diện trang chủ



4.1.1.2.2. Giao diện thông tin khách hàng



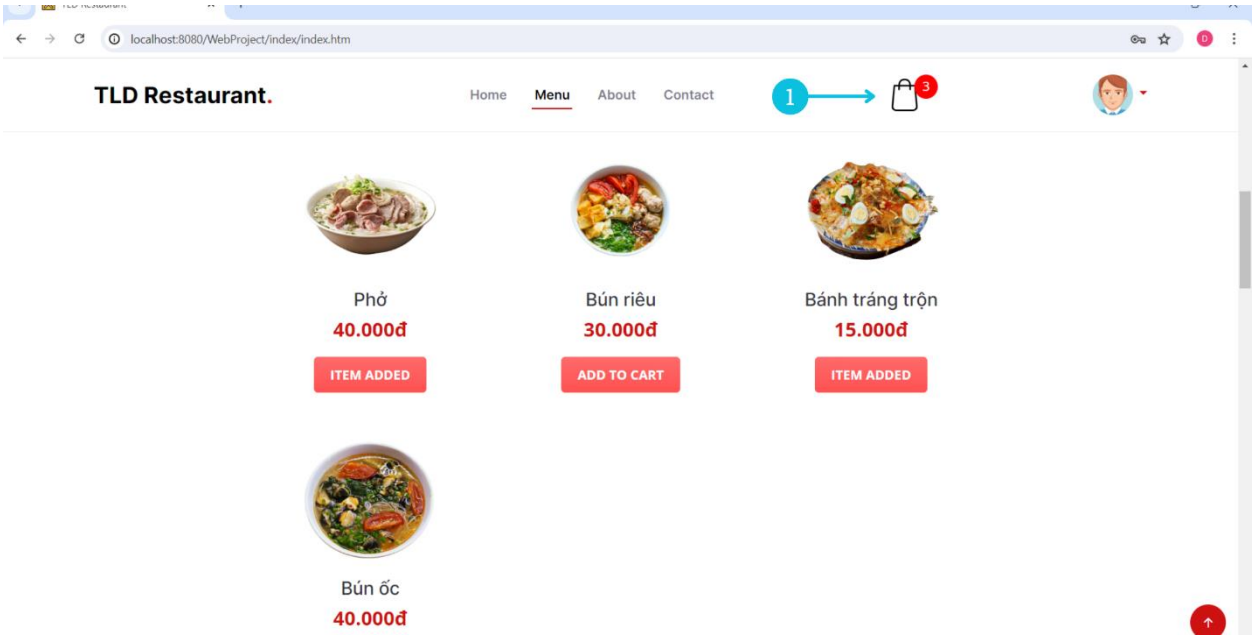
4.1.1.2.3. Giao diện chi tiết đơn hàng:



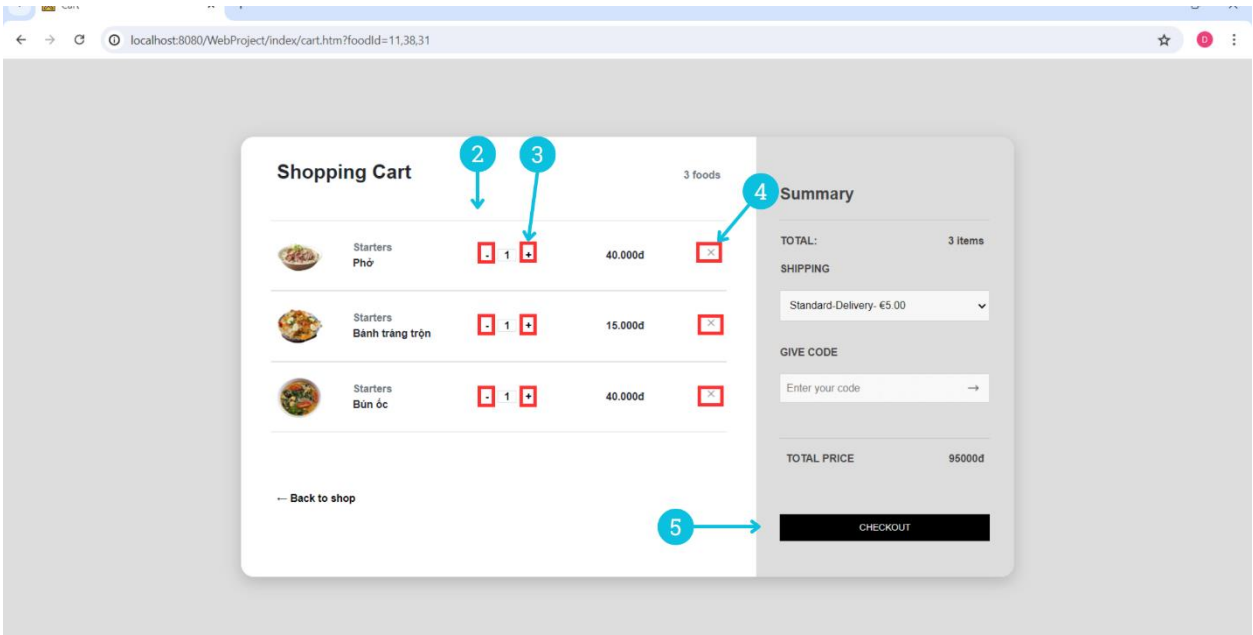
STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Chuyển sang giao diện thông tin người dùng	Click vào button “My profile”	showUserInfor()
2	Đăng xuất khỏi trang web	Click vào button “Logout”	logOut()
3	Mở form thay đổi mật khẩu	Click vào button “Change Password”	showFormChangePass()
4	Lưu thay đổi của khách hàng	Click vào button “Save Changes”	saveChanges()
5	Chuyển sang giao diện trang chủ	Click vào button ”Back”	showIndex()
6	Chuyển sang giao diện chi tiết đơn hàng	Click vào button như trên hình	showOrderDetail()
7	Chuyển sang giao diện thông tin người dùng	Click vào button “My profile”	showUserInfor()
8	Thoát khỏi form thay đổi mật khẩu	Click vào button “Back”	closeForm ChangePass()
9	Thay đổi mật khẩu	Click vào button “Save changes”	changePassword()
10	Đánh giá nhà hàng	Click vào button “Submit”	rateWeb()

4.1.1.3. Giao diện đặt hàng

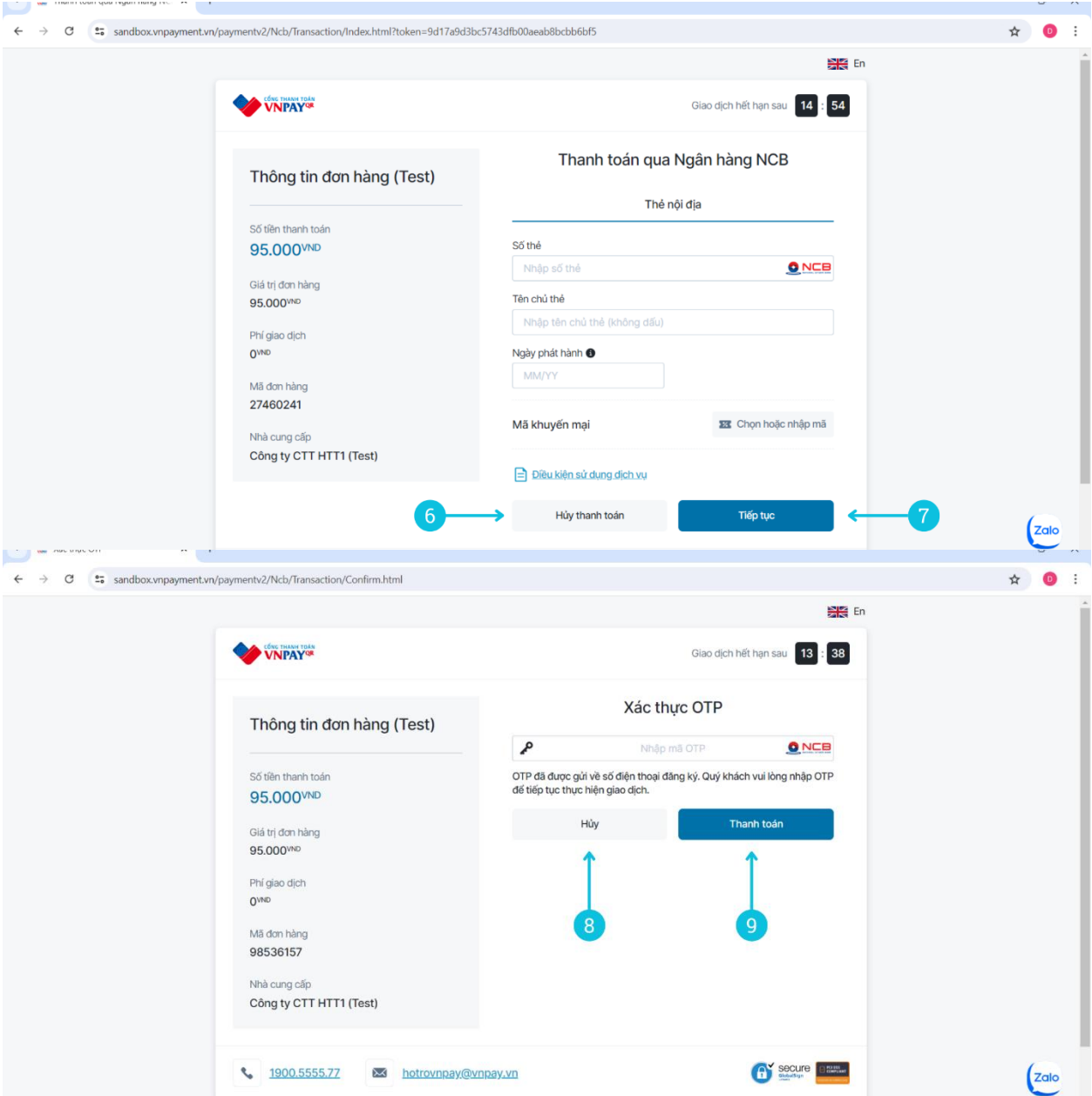
4.1.1.3.1. Giao diện món ăn



4.1.1.3.2. Giao diện giỏ hàng



4.1.1.3.3. Giao diện thanh toán vnpay



STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Show giao diện giỏ hàng	Click vào button giỏ hàng	showCart()
2	Giảm số lượng của món	Click vào button “-”	Xử lý trên giao diện
3	Tăng số lượng món	Click vào button “+”	Xử lý trên giao diện
4	Xóa món	Click vào button “X”	Xử lý trên giao diện
5	Chuyển sang giao diện thanh toán của vnpay	Click vào button “Check out”	createPayment()

6	Thoát giao diện thanh toán vnpay	Click vào button “Hủy thanh toán”	Chức năng hỗ trợ của vnpay
7	Chuyển sang giao diện nhập mã OTP	Click vào button “Tiếp tục”	Chức năng hỗ trợ của vnpay
8	Thoát giao diện thanh toán vnpay	Click vào button “Hủy thanh toán”	Chức năng hỗ trợ của vnpay
9	Thanh toán đơn hàng	Click vào button “Thanh toán”	Chức năng hỗ trợ của vnpay

4.1.2. Thiết kế chi tiết các hàm

4.1.2.1. Các hàm show giao diện

- showSignUp()
- showForgotPassword()
- showLogin()
- showFormChangePass()

Các hàm show giao diện ở trên được xử lý trên giao diện

- showIndex()
- showUserInfor()
- showOrderDetail()
- showCart()

Pseudo Code chung cho các hàm load dữ liệu từ database:

- Step 1: Load toàn bộ dữ liệu của table từ database lên controller
- Step 2: Truyền dữ liệu đã chọn ra ngoài view.

4.1.2.2. Các hàm xử lý chức năng

- signIn()

Pseudo Code:

- Step 1: Chuyển username và password người dùng nhập vào cho controller
- Step 2: Thực hiện kiểm tra trùng username và password trên 2 table Customers và Admins
- Step 3: Nếu không trùng username hoặc password báo lỗi về cho người dùng. Nếu trùng thì trả về controller tương ứng với vai trò của username

- forgotPassword()

Pseudo Code:

- Step 1: Gửi mail người dùng nhập vào về cho controller
- Step 2: Kiểm tra mail người dùng có hợp lệ không
- Step 3: Nếu hợp lệ tiến hành tạo mail với mã hóa mật khẩu mới về cho mail người dùng và cập nhập mật khẩu thành mật khẩu đã được mã hóa trên database
- Step 4: Nếu mail không hợp lệ thì thông báo về cho người dùng.

- signUp()

Pseudo Code:

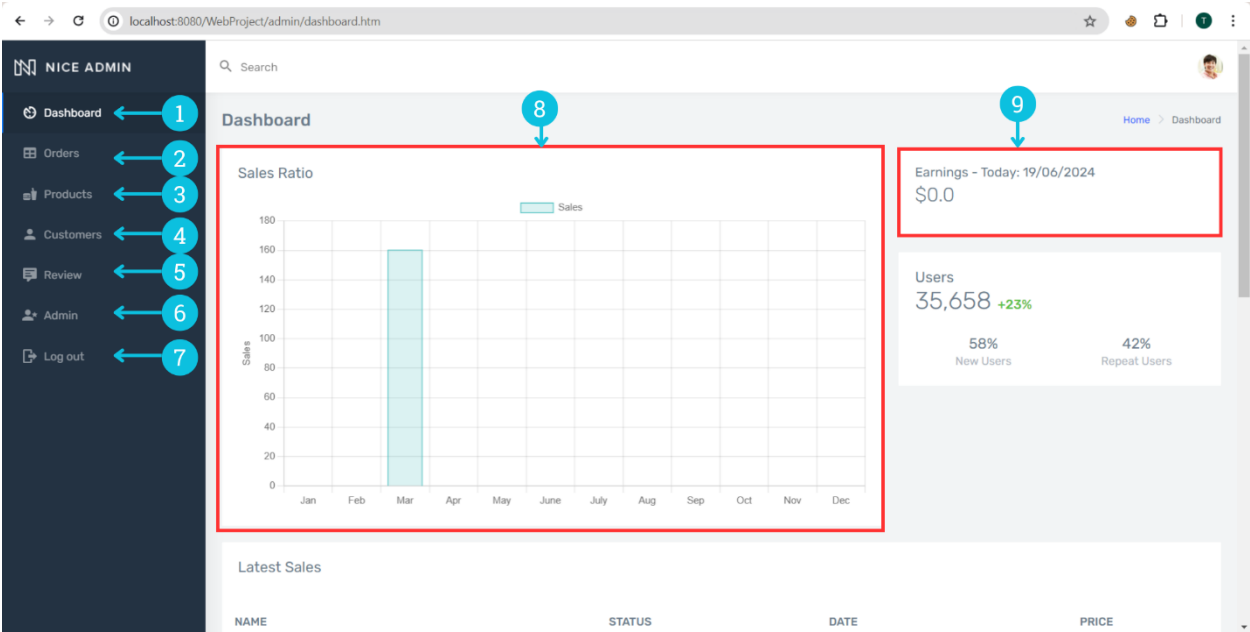
- Step 1: Gửi thông tin của customer cho controller.
- Step 2: Kiểm tra username và mail có hợp lệ hay không

- Step 3: Nếu không hợp lệ trả về thông báo cho người dùng
- Step 4: Nếu hợp lệ trả về trang chủ của web cho người dùng và cập nhập lên customer table
- logout()
Pseudo Code:
 - Step 1: Xóa tất cả các attribute ứng với 1 session của người dùng
 - Step 2: Return về trang chủ
- saveChanges()
Pseudo Code:
 - Step 1: Gửi các thông tin người dùng đã chỉnh sửa lên trên controller
 - Step 2: Kiểm tra email người dùng đã thay đổi có tồn tại trước chưa
 - Step 3: Nếu đã tồn tại thì trả về thông báo cho người dùng
 - Step 4: Nếu chưa thì tiến hành update thông tin của customer dựa vào username
- changePassword()
Pseudo Code:
 - Step 1: Gửi mật khẩu mới lên cho controller
 - Step 2: Tiến hành cập nhập mật khẩu mới dựa vào username của customer
- rateWeb()
Pseudo Code:
 - Step 1: Gửi lời nhận xét và số sao đánh giá về cho controller
 - Step 2: Cập nhập bảng Comments với lời nhận xét và số sao ứng với username của khách hàng
- createPayment()
Pseudo Code:
 - Step 1: Gửi thông tin đơn hàng về cho controller
 - Step 2: Lưu các thông tin vào trong session
 - Step 3: Tạo đường link gồm các thông tin cần thiết về đơn hàng để vnpay tạo ra giao dịch
 - Step 4: Tiến hành chạy đường link trên trong thanh url của người dùng để tiến hành thanh toán

4.2. Admin

4.2.1. Thiết kế giao diện

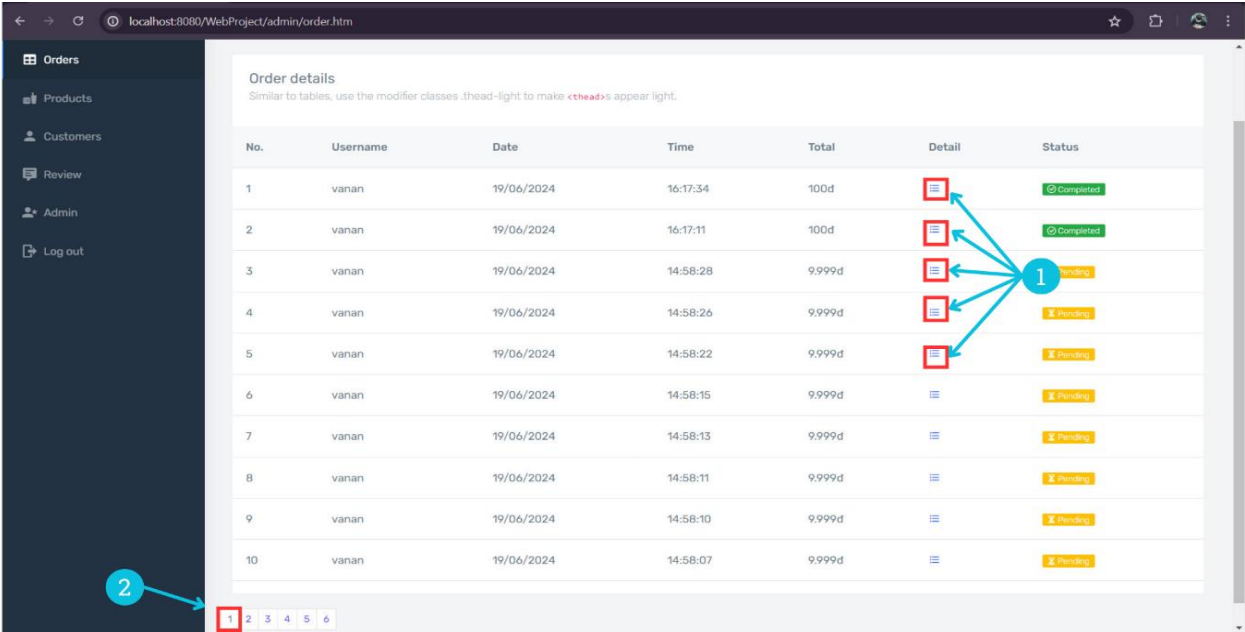
4.2.1.1. Giao diện Dashboard (Màn hình chính)



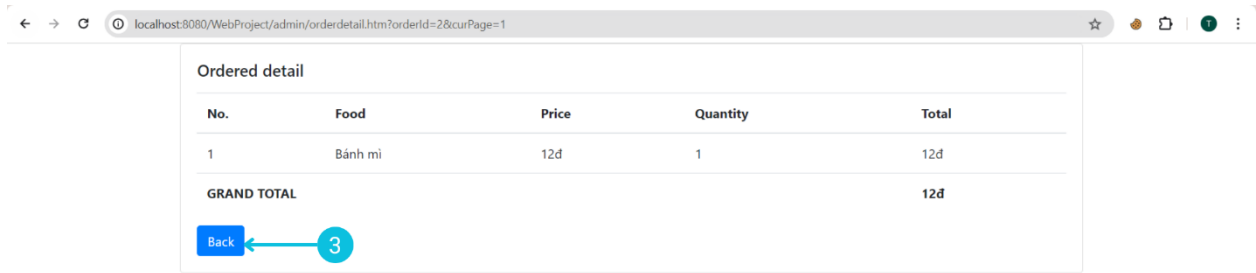
STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Chuyển sang giao diện Dashboard	Click vào button “Dashboard”	showDashboard()
2	Chuyển sang giao diện Orders	Click vào button “Orders”	showOrder()
3	Chuyển sang giao diện Products	Click vào button “Products”	showProduct()
4	Chuyển sang giao diện Customers	Click vào button “Customers”	showCustomer()
5	Chuyển sang giao diện Review	Click vào button “Review”	showReview()
6	Chuyển sang giao diện Admin	Click vào button “Admin”	showAdmin()
7	Đăng xuất tài khoản login	Click vào button “Log out”	logout()
8	Load dữ liệu doanh thu của từng tháng trong năm	Khi gọi hàm	calSales()
9	Load doanh thu hiện tại trong ngày	Khi gọi hàm	calSaleInDate()

4.2.1.2. Giao diện Orders

4.2.1.2.1. Bảng Orders



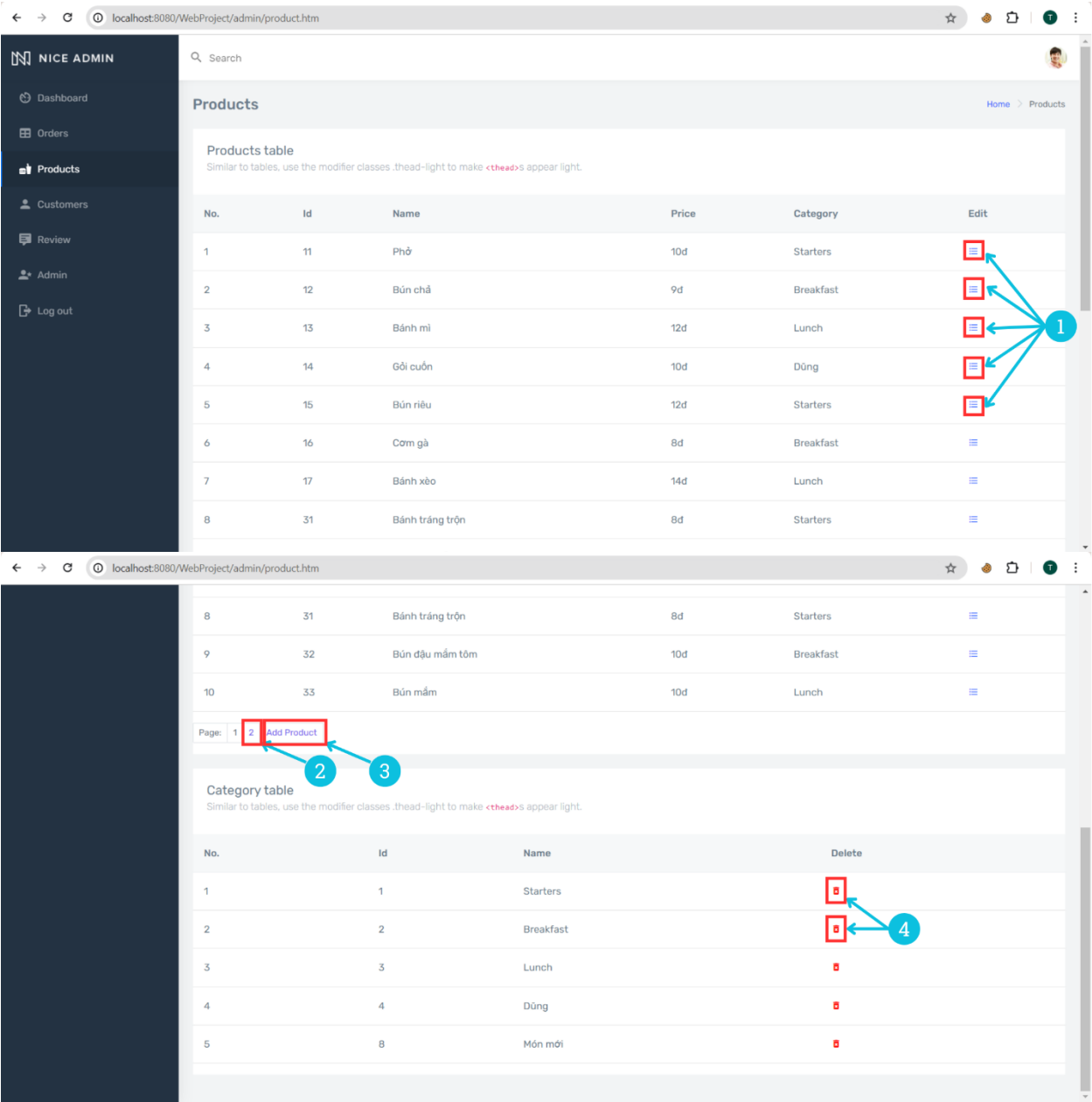
4.2.1.2.1. Giao diện order detail



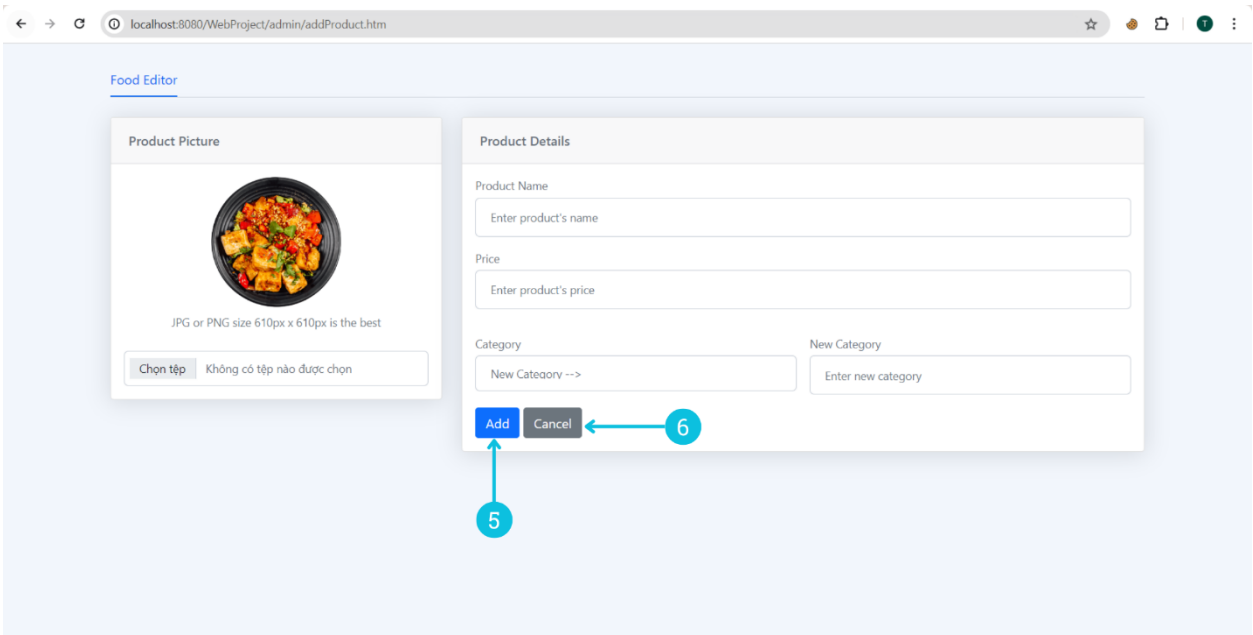
STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Xem chi tiết các món ăn trong order đã được đặt	Click vào button “≡”	showOrderDetail()
2	Load các dữ liệu order thuộc từng trang	Click vào button thể hiện số trang	loadOrderByPage()
3	Quay lại trang Order	Click vào button “Back”	backOrder()

4.2.1.3. Giao diện Product

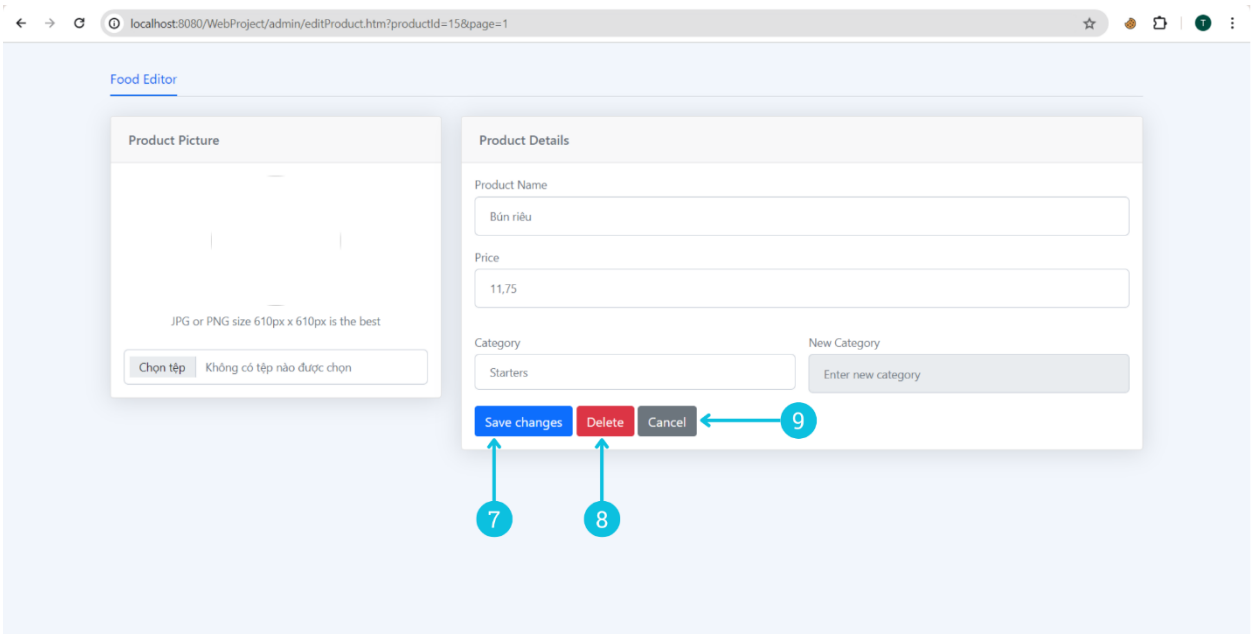
4.2.1.3.1. Bảng Product



4.2.1.3.2. Giao diện thêm product



4.2.1.3.3. Giao diện sửa product

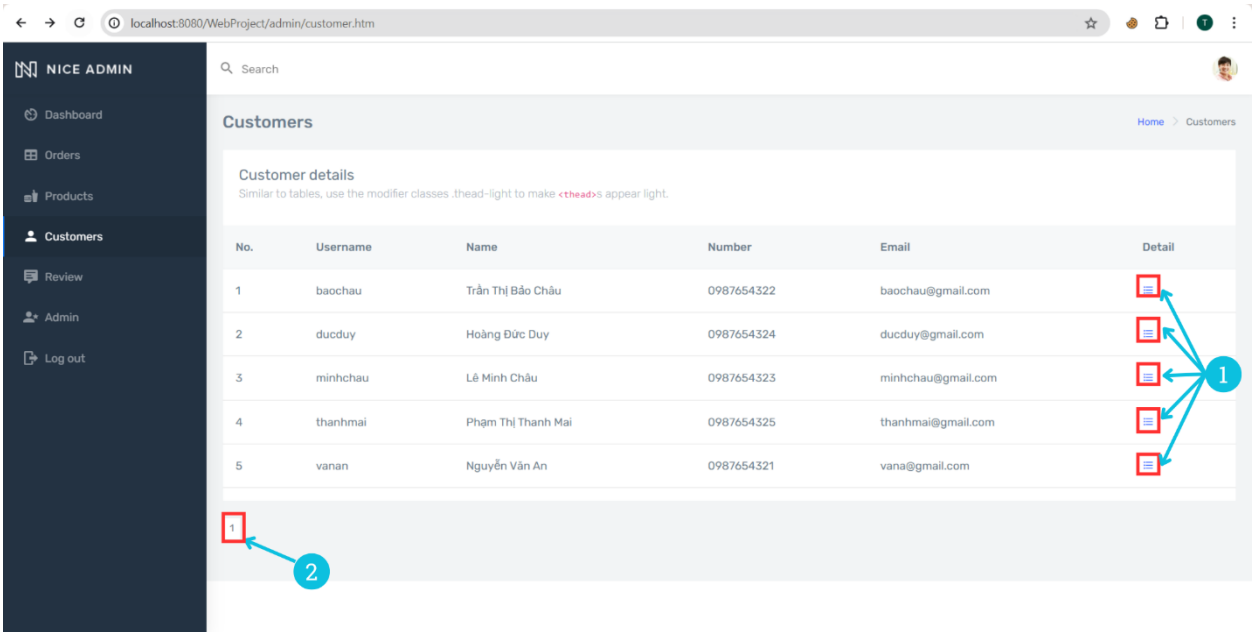


STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Show giao diện chỉnh sửa chi tiết các product	Click vào button “≡”	showProductDetail()
2	Load các dữ liệu product thuộc từng trang	Click vào button thể hiện số trang	loadProductByPage()
3	Show giao diện thêm một product	Click vào button “Add Product”	showAddProduct()

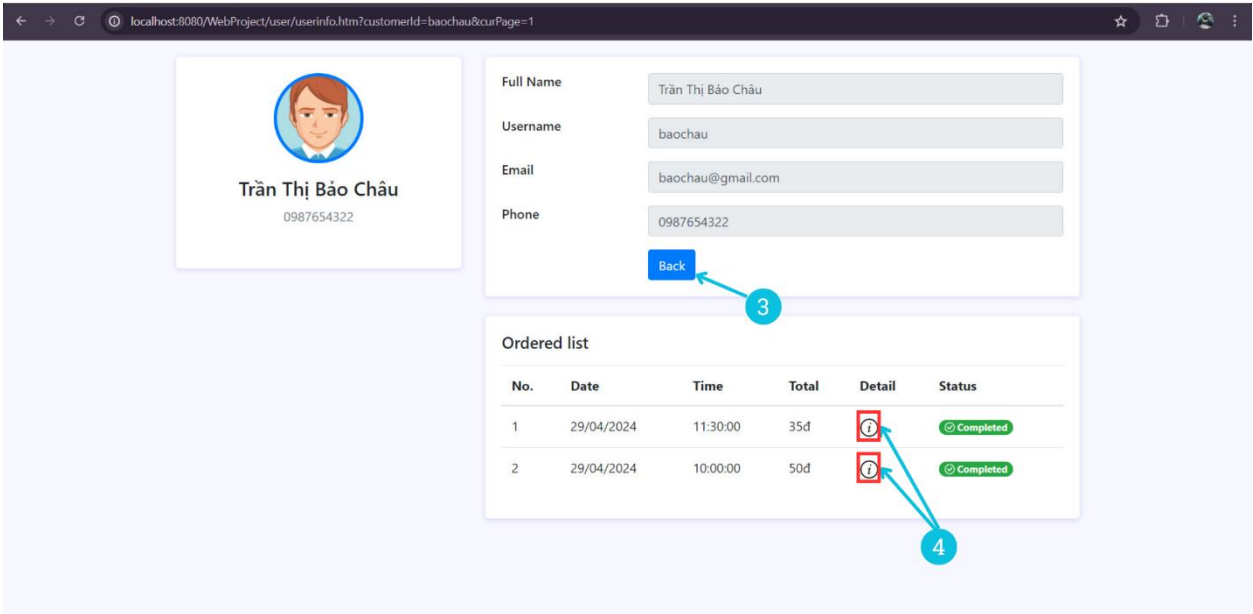
4	Xóa một category	Click vào button trong cột Delete của Categories	deleteCategory()
5	Thêm product	Click vào button “Add”	addProduct()
6	Thoát giao diện add product	Click vào button “Cancel”	backProduct()
7	Sửa thông tin product	Click vào button “Save changes”	modifyProduct()
8	Xóa product	Click vào button “Delete”	deleteProduct()
9	Thoát giao diện sửa product	Click vào button “Cancel”	backProduct()

4.2.1.4. Giao diện Customers

4.2.1.4.1. Bảng Customers

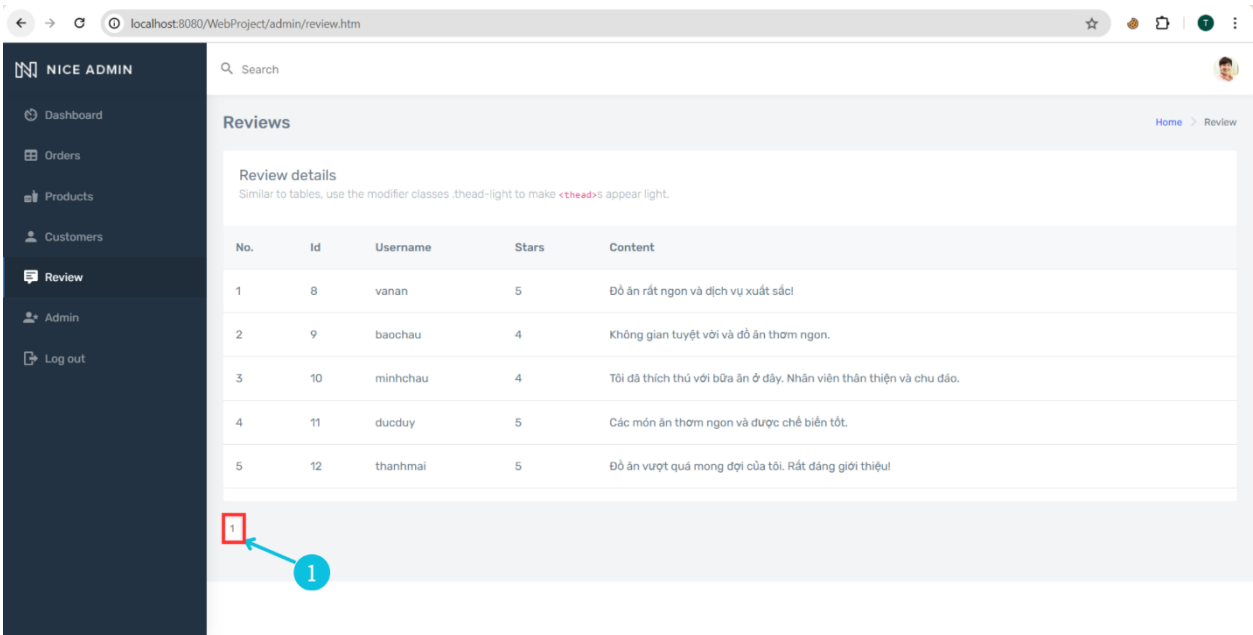


4.2.1.4.1. Giao diện thông tin của customer



STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Show giao diện thông tin của customer	Click vào button “≡”	showCustomerDetail()
2	Load các dữ liệu customer thuộc từng trang	Click vào button thể hiện số trang	loadCustomerByPage()
3	Thoát giao diện thông tin của customer	Click vào button “Back”	backCustomer()
4	Xem chi tiết order của khách hàng	Click vào button trong cột Details của khách hàng	showOrderDetail()

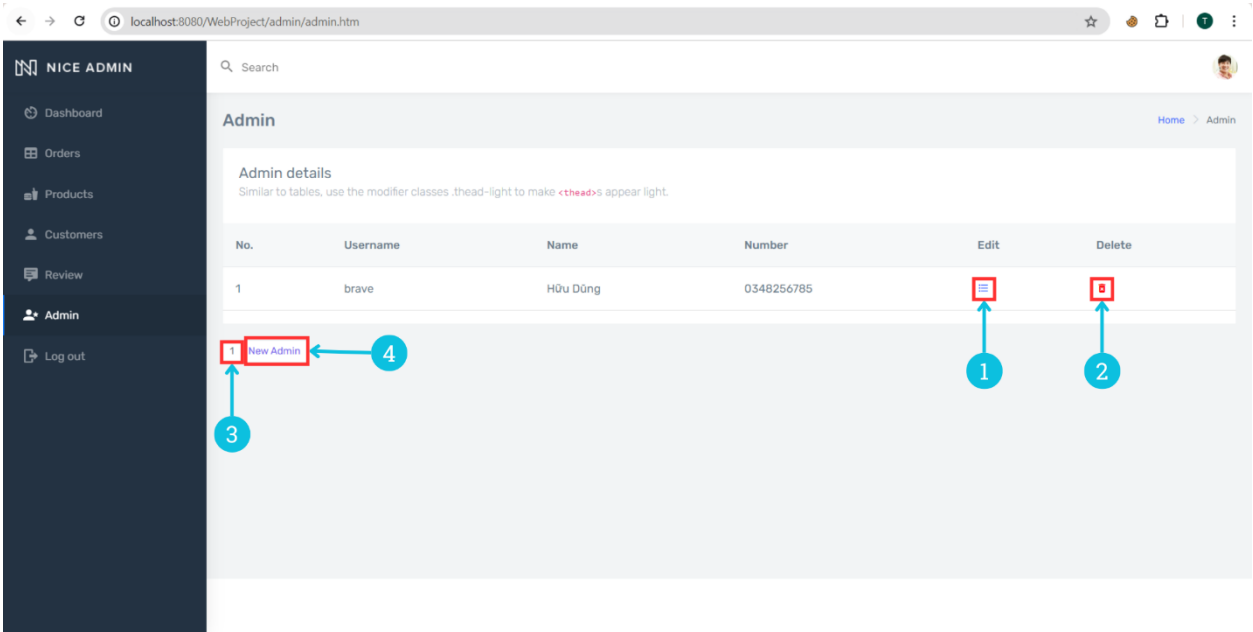
4.2.1.5. Giao diện Review



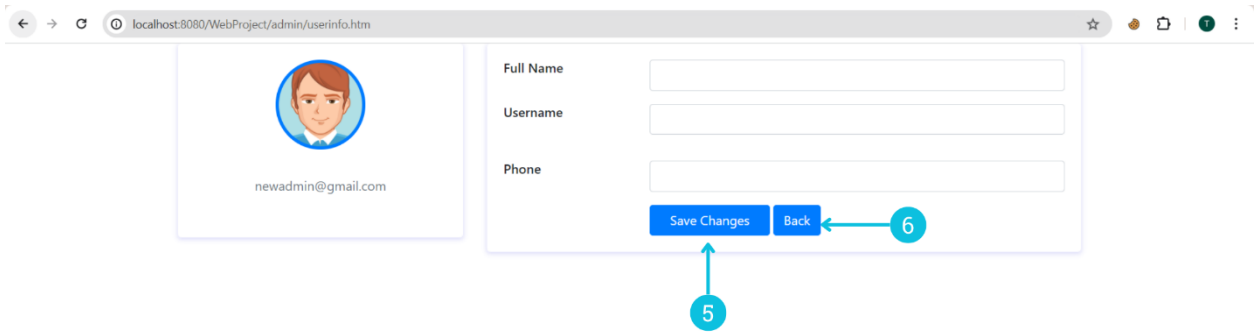
STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Load các dữ liệu review thuộc từng trang	Click vào button thể hiện số trang	loadReviewByPage()

4.2.1.6. Giao diện Admin

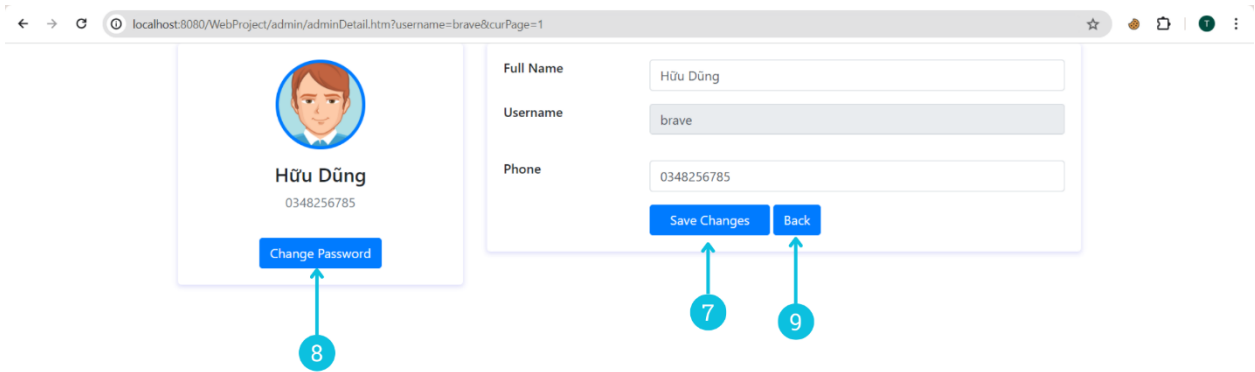
4.2.1.6.1. Bảng admin



4.2.1.6.2. Giao diện thêm admin



4.2.1.6.3. Giao diện sửa admin



STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Show giao diện chỉnh sửa chi tiết admin	Click vào button “≡”	showAdminDetail()
2	Load các dữ liệu admin thuộc từng trang	Click vào button thể hiện số trang	loadAdminByPage()
3	Show giao diện thêm một admin	Click vào button “New Admin”	showAddAdmin()

4	Xóa một admin	Click vào button trong cột Delete của Admin	deleteAdmin()
5	Thêm admin	Click vào button “Save changes”	addAdmin()
6	Thoát giao diện add admin	Click vào button “Back”	backAdmin()
7	Sửa thông tin admin	Click vào button “Save changes”	modifyAdmin()
8	Đổi password của admin	Click vào button “Change Password”	changePassword()
9	Thoát giao diện sửa admin	Click vào button “Back”	backAdmin()

4.2.2. Thiết kế chi tiết các hàm

4.2.2.1. Các hàm load dữ liệu

- loadOrderByPage()
- loadProductByPage()
- loadCustomerByPage()
- loadReviewByPage()
- loadAdminByPage()

Pseudo Code chung cho các hàm load theo page:

- Step 1: Load toàn bộ dữ liệu của table từ database lên controller
- Step 2: Chi dữ liệu đã load thành các đoạn tương ứng với các page
- Step 3: Chọn đoạn dữ liệu tương ứng với page đã chọn
- Step 4: Truyền dữ liệu đã chọn ra ngoài view.

4.2.2.2. Các hàm xử lý chức năng

- calSales()

Pseudo Code:

- Step 1: Load các toàn bộ order trong database
- Step 2: Dùng thư viện lấy thông tin năm hiện tại
- Step 3: Tạo mảng sales gồm 12 phần tử tương ứng với 12 tháng
- Step 4: Chạy vòng lặp của bảng order, kiểm tra order có thuộc năm hiện tại không, nếu có thì cộng dồn vào phần tử trong mảng sales với index = tháng order – 1.
- Step 5: Trả về mảng sales

- calSaleInDate()

Pseudo Code:

- Step 1: Load các toàn bộ order trong database
- Step 2: Dùng thư viện lấy thông tin ngày, tháng, năm hiện tại
- Step 3: Tạo biến sale = 0
- Step 4: Chạy vòng lặp của bảng order, kiểm tra order có thuộc ngày, tháng, năm hiện tại không, nếu có thì cộng dồn vào biến sale.
- Step 5: Trả về biến sale

- addProduct()

Pseudo Code:

- Step 1: Kiểm tra category của product có tồn tại hay không, nếu không có thì mình thêm mới category.
- Step 2: Lấy id category
- Step 3: Tạo object product mới và fill các thuộc tính của object
- Step 4: Tạo câu lệnh query INSERT mới
- Step 5: Thực thi câu lệnh query
- Step 6: Commit và kiểm tra nếu có lỗi thì return “Lỗi”

- modifyProduct(product_id)

Pseudo Code:

- Step 1: Kiểm tra product_id có tồn tại trong bảng hay không, nếu không có thì return “Lỗi”
- Step 2: Lấy object product theo product_id trong bảng
- Step 3: Fill các giá trị mới vào trong object
- Step 4: Tạo câu lệnh query UPDATE mới
- Step 5: Thực thi câu lệnh query
- Step 6: Commit và kiểm tra nếu có lỗi thì return “Lỗi”

- deleteProduct(product_id)

Pseudo Code:

- Step 1: Kiểm tra product_id có tồn tại trong bảng hay không, nếu không có thì return “Lỗi”
- Step 2: Lấy object product theo product_id trong bảng
- Step 3: Tạo câu lệnh query DELETE mới
- Step 4: Thực thi câu lệnh query
- Step 5: Commit và kiểm tra nếu có lỗi thì return “Lỗi”

- deleteCategory(category_id)

Pseudo Code:

- Step 1: Kiểm tra category_id có tồn tại trong bảng hay không, nếu không có thì return “Lỗi”
- Step 2: Kiểm tra có product nào thuộc category_id không, nếu có thì return “Lỗi”
- Step 3: Lấy object category theo category_id trong bảng
- Step 4: Tạo câu lệnh query DELETE mới
- Step 5: Thực thi câu lệnh query
- Step 6: Commit và kiểm tra nếu có lỗi thì return “Lỗi”

- addAdmin()

Pseudo Code:

- Step 1: Kiểm tra username có bị trùng hay không, nếu có thì return “Lỗi”
- Step 2: Tạo object admin mới và fill các thuộc tính của object
- Step 3: Tạo câu lệnh query INSERT mới
- Step 4: Thực thi câu lệnh query
- Step 5: Commit và kiểm tra nếu có lỗi thì return “Lỗi”

- modifyAdmin(username)

Pseudo Code:

- Step 1: Kiểm tra username có tồn tại trong bảng hay không, nếu không có thì return “Lỗi”
- Step 2: Lấy object admin theo username trong bảng
- Step 3: Fill các giá trị mới vào trong object
- Step 4: Tạo câu lệnh query UPDATE mới
- Step 5: Thực thi câu lệnh query
- Step 6: Commit và kiểm tra nếu có lỗi thì return “Lỗi”
- deleteAdmin(username)
Pseudo Code:
 - Step 1: Kiểm tra username có tồn tại trong bảng hay không, nếu không có thì return “Lỗi”
 - Step 2: Lấy object admin theo username trong bảng
 - Step 3: Tạo câu lệnh query DELETE mới
 - Step 4: Thực thi câu lệnh query
 - Step 5: Commit và kiểm tra nếu có lỗi thì return “Lỗi”
- changePassword(username)
Pseudo Code:
 - Step 1: Kiểm tra username có tồn tại trong bảng hay không, nếu không có thì return “Lỗi”
 - Step 2: Lấy object admin theo username trong bảng
 - Step 3: Fill giá trị trường password vào trong object
 - Step 4: Tạo câu lệnh query UPDATE mới
 - Step 5: Thực thi câu lệnh query
 - Step 6: Commit và kiểm tra nếu có lỗi thì return “Lỗi”

4.2.2.3. Các hàm show giao diện

- showProductDetail(), showOrderDetail(), showCustomerDetail(), showAdminDetail()
Pseudo Code chung:
 - Step 1: Lấy id của object
 - Step 2: Thực hiện lệnh SELECT object FROM table_name WHERE id = id
 - Step 3: Nếu “Lỗi” thì in lỗi ra view, nếu không return object ra view
- showOrder(), showProduct(), showCustomer(), showReview(), showAdmin()
Pseudo Code chung:
 - Step 1: Thực hiện hàm loadByPage(page=1)
 - Step 2: Lấy được object từ hàm
 - Step 3: Return object ra view
- backOrder(), backProduct(), backCustomer(), backAdmin()
Pseudo Code chung:
 - Step 1: Lấy thông tin curPage từ view hiện tại
 - Step 2: Thực hiện hàm loadByPage(page= curPage)
 - Step 3: Lấy được object từ hàm
 - Step 4: Return object ra view
- showAddProduct, showAddAdmin()

Pseudo Code chung:

- Step 1: Return add view của object

- showDashboard()

Pseudo Code chung:

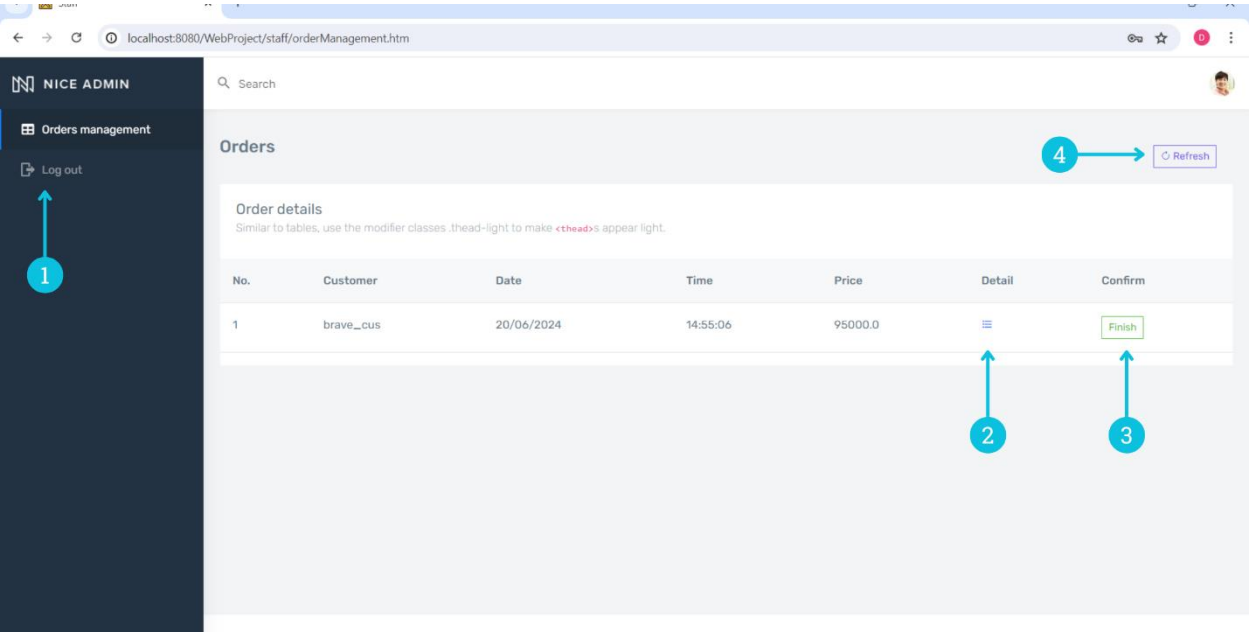
- Step 1: Thực hiện hàm calSales() và lấy được mảng sales
- Step 2: Thực hiện hàm calSaleInDate() và lấy được biến sale
- Step 3: Return sales và sale ra view

4.3. Staff

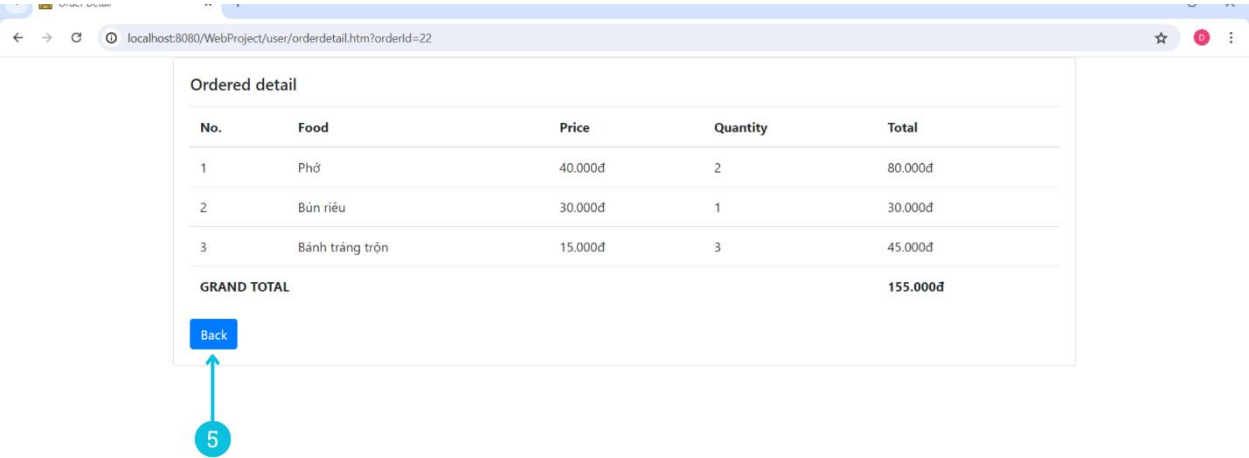
4.3.1. Thiết kế giao diện

4.3.1.1. Giao diện Quản lý order

4.3.1.1.1. Giao diện danh sách order:



4.3.1.1.2. Form chi tiết đơn hàng



STT	Chức năng	Điều kiện thực thi	Tên hàm
1	Đăng xuất khỏi trang web	Click vào button “Logout”	logOut()
2	Chuyển sang giao diện chi tiết đơn hàng	Click vào button “≡”	ShowOrderDetail()
3	Xác nhận đơn hàng	Click vào button “Finish”	confirmOrder()
4	Update bảng order	Click vào button “Refresh”	updateOrders()
5	Chuyển sang giao diện quản lý order	Click vào button “back”	showOrders()

4.3.2. Thiết kế chi tiết các hàm

4.3.2.1. Các hàm load dữ liệu

- ShowOrderDetail()
- showOrders()

Pseudo Code chung cho các hàm load theo page:

- Step 1: Load toàn bộ dữ liệu của table từ database lên controller
- Step 2: Truyền dữ liệu đã chọn ra ngoài view.

4.3.2.2. Các hàm xử lý chức năng

- logOut()

Pseudo Code:

- Step 1: Xóa tất cả các attribute ứng với 1 session của người dùng
- Step 2: Return về trang chủ

- confirmOrder()

Pseudo Code:

- Step 1: Lấy orderId gửi lên controller
- Step 2: Set status của orderId tương ứng thành “Completed”
- Step 3: Update status theo orderId đó trong orders table

- updateOrders()

Pseudo Code:

- Step 1: Load toàn bộ dữ liệu của table từ database lên controller.
- Step 2: Thiết lập table html dựa trên các dữ liệu từ table orders
- Step 3: Gắn table html trên vào trong HTTP response sau đó trả về cho người dùng

CHƯƠNG 5. CÀI ĐẶT VÀ KẾT QUẢ THỰC NGHIỆM

5.1. Cài đặt

5.1.1. Môi trường phát triển

Để phát triển và triển khai hệ thống web nhà hàng trực tuyến, Nhóm đã sử dụng các công cụ và hệ điều hành sau:

- + Hệ điều hành Window 10.
- + IDE: Eclipse.
- + Ngôn ngữ lập trình: Java, JavaScript, HTML, CSS.
- + Cơ sở dữ liệu: Microsoft SQL Server.
- + Máy chủ web: Apache TomCat.

5.1.2. Mô hình phát triển

Nhóm đã đề xuất sử dụng mô hình Spring MVC để thiết kế và lập trình web vì những lý do sau đây:

1. *Cấu trúc rõ ràng và dễ bảo trì*: Mô hình Spring MVC giúp tách biệt rõ ràng giữa ba thành phần chính: Model (dữ liệu và logic nghiệp vụ), View (giao diện người dùng), và Controller (điều khiển và xử lý yêu cầu). Điều này giúp code dễ đọc, dễ bảo trì và phát triển.
2. *Trình tự xử lý rõ ràng*: Controller xử lý các yêu cầu từ người dùng và tương tác với Model. Model quản lý dữ liệu và thực hiện các thao tác với cơ sở dữ liệu. View hiển thị thông tin cho người dùng dựa trên dữ liệu từ Model. Sự tách biệt này giúp trình tự xử lý rõ ràng và dễ hiểu.
3. *Khả năng mở rộng tốt*: Spring MVC cung cấp nhiều tính năng hỗ trợ mở rộng và tích hợp với các công nghệ khác như Spring Security, Spring Data, và các dịch vụ RESTful. Điều này giúp ứng dụng có thể phát triển và mở rộng dễ dàng theo nhu cầu.
4. *Tích hợp tốt với Spring Framework*: Spring MVC là một phần của Spring Framework, do đó nó dễ dàng tích hợp và sử dụng các tính năng khác của Spring như Dependency Injection (DI), Aspect-Oriented Programming (AOP), và các dịch vụ giao dịch (transaction services).
5. *Cộng đồng hỗ trợ mạnh mẽ*: Với một cộng đồng lớn và tài liệu phong phú, người dùng Spring MVC có thể dễ dàng tìm kiếm sự hỗ trợ và tài liệu học tập từ cộng đồng, diễn đàn, và các nguồn tài nguyên trực tuyến.
6. *Hỗ trợ các tính năng mới*: Spring MVC liên tục được cập nhật với các tính năng và cải tiến mới, giúp ứng dụng luôn bắt kịp xu hướng và công nghệ mới.
7. *Hiệu suất cao*: Spring MVC được thiết kế để tối ưu hóa hiệu suất và khả năng chịu tải của ứng dụng, đặc biệt khi kết hợp với các kỹ thuật như caching, lazy loading, và asynchronous processing.

8. *Tích hợp dễ dàng với các view technologies*: Spring MVC hỗ trợ nhiều công nghệ view khác nhau như JSP, Thymeleaf, FreeMarker, và các framework JavaScript phía client như Angular và React, giúp phát triển giao diện người dùng linh hoạt và hiện đại.

5.2. Triển khai và thực nghiệm

Mở SQL Server để web có thể cập nhật dữ liệu.

Mở Eclipse và import dự án sau đó sử dụng server TomCat để chạy phần backend.

5.2.1. Báo cáo Kiểm thử

ID	Mô tả trường hợp kiểm thử	Thủ tục kiểm thử	Kết quả mong đợi	Kết quả
1	Kiểm tra tên đăng nhập và mật khẩu trống	Tên đăng nhập và mật khẩu trống	Thông báo 'Tên đăng nhập và mật khẩu là bắt buộc'	Đạt
2	Kiểm tra đăng nhập thành công	Nhập tên đăng nhập và mật khẩu hợp lệ	Người dùng đăng nhập thành công	Đạt
3	Kiểm tra mật khẩu không chính xác	Nhập tên đăng nhập hợp lệ và mật khẩu không chính xác	Thông báo 'Tên đăng nhập hoặc mật khẩu không chính xác'	Đạt
4	Kiểm tra đăng xuất	Người dùng nhấn nút đăng xuất	Người dùng được đăng xuất và trở về trang chủ	Đạt
5	Kiểm tra xem chi tiết đơn hàng	Đi tới trang chi tiết đơn hàng	Chi tiết đơn hàng được hiển thị	Đạt
6	Kiểm tra đăng ký với tên đăng nhập đã tồn tại	Thử đăng ký với tên đăng nhập đã tồn tại	Thông báo 'Tên đăng nhập đã tồn tại'	Đạt
7	Kiểm tra đăng ký với tên đăng nhập mới	Nhập tên đăng nhập mới và các thông tin cần thiết	Người dùng đăng ký thành công	Đạt
8	Kiểm tra quên mật khẩu	Nhấn vào 'Quên mật khẩu' và nhập email đã đăng ký	Mật khẩu mới được gửi đến email của người dùng	Đạt
9	Kiểm tra giỏ hàng khi chưa đăng nhập	Nhấn vào 'Giỏ hàng' khi chưa đăng nhập	Thông báo yêu cầu người dùng đăng nhập trước khi xem giỏ hàng	Đạt
10	Kiểm tra thêm món vào giỏ hàng	Chọn món và nhấn 'Thêm vào giỏ hàng'	Món được thêm vào giỏ hàng	Đạt
11	Kiểm tra xóa món khỏi giỏ hàng	Đi tới giỏ hàng, chọn món và nhấn 'X'	Món được xóa khỏi giỏ hàng	Đạt

12	Kiểm tra sửa số lượng món trong giỏ hàng	Đi tới giỏ hàng, chọn món và tăng, giảm số lượng	Số lượng món được cập nhật trong giỏ hàng	Đạt
13	Kiểm tra thanh toán	Đi tới trang thanh toán và hoàn tất thanh toán	Đơn hàng được đặt thành công	Đạt
14	Kiểm tra theo dõi trạng thái đơn hàng	Đi tới trang cá nhân và xem trạng thái đơn hàng	Trạng thái đơn hàng được hiển thị	Đạt
15	Kiểm tra đánh giá và nhận xét	Đi tới trang chủ, nhập đánh giá và nhận xét	Đánh giá và nhận xét được gửi thành công	Đạt
16	Kiểm tra admin thêm món mới	Admin vào quản lý món, thêm món mới	Món mới được thêm thành công	Đạt
17	Kiểm tra admin xóa món	Admin vào quản lý món, xóa món	Món được xóa thành công	Đạt
18	Kiểm tra admin sửa món	Admin vào quản lý món, sửa thông tin món	Thông tin món được cập nhật thành công	Đạt
19	Kiểm tra admin xem danh sách người dùng	Admin vào quản lý người dùng	Danh sách người dùng được hiển thị	Đạt
20	Kiểm tra admin xem danh sách đơn hàng	Admin vào quản lý đơn hàng	Danh sách đơn hàng được hiển thị	Đạt
21	Kiểm tra admin xem báo cáo doanh thu	Admin vào dashboard	Báo cáo doanh thu theo tháng được hiển thị	Đạt
22	Kiểm tra admin quản lý đánh giá và nhận xét	Admin vào quản lý đánh giá và nhận xét	Danh sách đánh giá và nhận xét được hiển thị	Đạt
23	Kiểm tra nhân viên xem danh sách đơn hàng cần xử lý	Nhân viên đăng nhập và vào danh sách đơn hàng	Danh sách đơn hàng cần xử lý được hiển thị	Đạt
24	Kiểm tra nhân viên cập nhật trạng thái đơn hàng	Nhân viên cập nhật trạng thái đơn hàng sau khi xử lý	Trạng thái đơn hàng được cập nhật(Pending -> Completed)	Đạt
25	Kiểm tra đăng ký với các trường trống	Các trường thông tin cần nhập bị thiếu	Thông báo trường thông tin cần nhập bị thiếu và yêu cầu người dùng nhập thêm	Đạt
26	Kiểm tra xem menu	Khách hàng ấn vào mục menu	Danh sách các món ăn đã có của nhà hàng	Đạt

27	Kiểm tra profile khách hàng	Khách hàng ấn vào mục profile của mình và thay đổi thông tin	Xuất hiện thông tin của khách hàng và thay đổi được	Đạt
28	Kiểm tra admin xóa danh mục	Admin vào products, ấn xóa danh mục	Chỉ xóa được các danh mục chưa có sản phẩm nào	Đạt
29	Kiểm tra admin thêm admin	Admin vào mục Admin và ấn thêm admin	Thêm được admin mới	Đạt
30	Kiểm tra admin xóa admin	Admin vào mục Admin và ấn xóa admin	Thao tác lỗi	Đạt
31	Kiểm tra admin sửa admin	Admin vào mục Admin và ấn sửa admin	Sửa được các thông tin của một admin	Đạt
32	Kiểm tra admin cập nhật profile	Admin vào mục Profile và sửa thông tin sau đó ấn update profile	Sửa được thông tin của admin	Đạt

5.2.2. Đánh giá chức năng

STT	Chức năng	Mức độ hoàn thành	Ghi chú
1	Đăng nhập, đăng ký, đăng xuất	100%	Hoàn thành tốt
2	Quản lý món ăn	95%	Chưa hoàn thành phần nhập số lượng tồn
3	Giỏ hàng	100%	Hoàn thành tốt
4	Thanh toán	95%	Cài đặt thành công API, chưa sử dụng thực tế.
5	Quản lý, chỉnh sửa thông tin	100%	Hoàn thành tốt
6	Báo cáo, thống kê	70%	Đang phát triển thêm 1 vài báo cáo
7	Đánh giá chất lượng	100%	Hoàn thành tốt
8	Tư vấn trực tuyến cho khách hàng	50%	Đang được nghiên cứu và phát triển thêm

9	Gợi ý các món ăn phù hợp	50%	Đang được nghiên cứu và phát triển thêm
---	--------------------------	-----	---

CHƯƠNG 6. TỔNG KẾT

6.1. Mục tiêu và phạm vi của đề tài

Trong báo cáo này, nhóm đã nghiên cứu và phát một ứng dụng web để bán đồ ăn. Sử dụng các kỹ thuật và công nghệ như Spring MVC và HTML, CSS.

Mục đích của dự án là thiết kế và triển khai một hệ thống cho phép người dùng đăng ký, đăng nhập, xem và mua đồ ăn trực tuyến. Hơn nữa dự án còn giúp cho các chủ nhà hàng có thể quản lý tốt hơn.

6.2 Tóm tắt kết quả đạt được

Nhóm đã thành công trong việc thiết kế và phát triển một giao diện web thân thiện, dễ sử dụng đối với người dùng. Giúp cho việc quản lý của nhà hàng được đơn giản hóa, giúp tối ưu hiệu quả kinh doanh. Và hơn hết là, giúp cho việc đặt đồ ăn trực tuyến của khách hàng dễ dàng hơn, có nhiều tính năng và tiện ích hơn.

Dự án của Nhóm mang lại những kết quả đáng kể trong việc cải thiện trải nghiệm người dùng và tăng cường khả năng cạnh tranh của nhà hàng trong lĩnh vực đặt đồ ăn trực tuyến.

Việc thiết kế web được sử dụng theo mô hình Spring MVC giúp cho việc cài đặt, bảo trì và nâng cấp trong tương lai dễ dàng hơn. Bằng cách sử dụng Spring MVC, Nhóm có thể tách biệt logic điều khiển và dữ liệu một cách hiệu quả, từ đó tối ưu hóa quy trình phát triển và bảo trì hệ thống.

6.3. Đánh giá kết quả

Tuy dự án đã đạt được một số thành công nhất định, nhưng Nhóm đã nhận thấy những thách thức trong quá trình phát triển. Ví dụ như việc bảo mật của dự án về mặt dữ liệu người dùng, hỗ trợ khách hàng thông qua chat trực tuyến.

6.4. Học hỏi và đề xuất tương lai

Từ dự án này Nhóm đã học hỏi được nhiều về quản lý dự án phần mềm, phát triển giao diện người dùng hiệu quả và tích hợp các công nghệ mới nhất. Đề xuất trong tương lai là Nhóm sẽ cải thiện bảo mật cho dữ liệu người dùng và sẽ nâng cấp, bổ sung thêm nhiều tính năng mới để tăng tương tác với khách hàng.

6.5. Kết luận

Dự án không chỉ đạt được những mục tiêu đã đề ra mà còn đem lại giá trị thực tiễn cho cả nhà hàng và khách hàng. Nhóm rất hạnh phúc về những thành tựu đã đạt được và cam kết tiếp tục cải thiện và phát triển sản phẩm để đáp ứng nhu cầu ngày càng cao của thị trường.

TÀI LIỆU THAM KHẢO

- [1] <https://getbootstrap.com>
- [2] <https://stackoverflow.com>
- [3] chat.openai.com
- [4] Giáo trình Cơ sở dữ liệu PTIT, Ts. Phạm Thế Quế.
- [5] Slide Lập trình web PTIT, Nguyễn Trung Hiếu.
- [6] Giáo trình nhập môn công nghệ phần mềm PTIT, Trần Đình Quế, Nguyễn Mạnh Hùng.

PHÂN CÔNG CÔNG VIỆC

<div>Thành viên</div> <div>Công việc</div>	Nguyễn Hữu Dũng	Phạm Thanh Linh	Hà Hữu Thắng	Nguyễn Hoàng Thiện
Khảo sát hiện trạng, xác định yêu cầu	X		X	
Phân tích use case		X		X
Mô hình hóa, thiết kế cơ sở dữ liệu	X	X	X	X
Xây dựng chức năng Đăng nhập		X		X
Xây dựng chức năng Đăng xuất		X		X
Xây dựng chức năng Đăng ký		X		X
Xây dựng chức năng Quên mật khẩu		X		X
Xây dựng chức năng Thêm món vào giỏ hàng		X		X
Xây dựng chức năng Chỉnh sửa giỏ hàng		X		X
Xây dựng chức năng Thanh toán đơn hàng		X		X
Xây dựng chức năng Quản lý đơn hàng		X		X
Xây dựng chức năng Chỉnh sửa thông tin khách hàng		X		X
Xây dựng chức năng Xem thông tin khách hàng		X		X
Xây dựng chức năng Quản lý món ăn		X		X
Xây dựng chức năng Quản lý người dùng		X		X
Xây dựng chức năng Quản lý đơn hàng		X		X

Xây dựng chức năng Quản lý doanh thu		X		X
Xây dựng chức năng Quản lý đánh giá và nhận xét		X		X
Xây dựng chức năng Cập nhật trạng thái đơn hàng		X		X
Thiết kế giao diện chức năng Đăng nhập	X		X	
Thiết kế giao diện chức năng Đăng xuất	X		X	
Thiết kế giao diện chức năng Đăng ký	X		X	
Thiết kế giao diện chức năng Quên mật khẩu	X		X	
Thiết kế giao diện chức năng Chính sửa giỏ hàng	X		X	
Thiết kế giao diện chức năng Thanh toán đơn hàng	X		X	
Thiết kế giao diện chức năng Quản lý đơn hàng	X		X	
Thiết kế giao diện chức năng Chính sửa thông tin khách hàng	X		X	
Thiết kế giao diện Xem thông tin khách hàng	X		X	
Thiết kế giao diện Quản lý món ăn	X		X	
Thiết kế giao diện Quản lý người dùng	X		X	
Thiết kế giao diện Quản lý đơn hàng	X		X	
Thiết kế giao diện Quản lý doanh thu	X		X	
Thiết kế giao diện Quản lý đánh giá và nhận xét	X		X	
Viết báo cáo	X	X	X	X
Tổng hợp báo cáo	X	X	X	X

