



# Ứng dụng đặt hàng thức ăn: Happy Lunch

Version 3.1

Người viết: Nhóm 65

HCMUT, T5/2019



## LỊCH SỬ CHỈNH SỬA

NGÀY	Version	Thay đổi	Người thực hiện
03/04/2019	1.0	Thiết lập ban đầu	Nhóm 65
30/04/2019	2.0	Thêm vào một số chức năng	
04/05/2019	2.1	Hoàn chỉnh một số chức năng	
11/05/2019	3.0	Chỉnh sửa lỗi và thêm usecase	
22/05/2019	3.1	Hoàn thiện lại báo cáo	



## Mục lục

1. Giới thiệu.....	<b>Error! Bookmark not defined.</b>
2. Tầm nhìn hệ thống.....	<b>Error! Bookmark not defined.</b>
3. Các diagram.....	<b>Error! Bookmark not defined.</b>
4. Functional Requirement.....	9
5. Kết luận.....	13

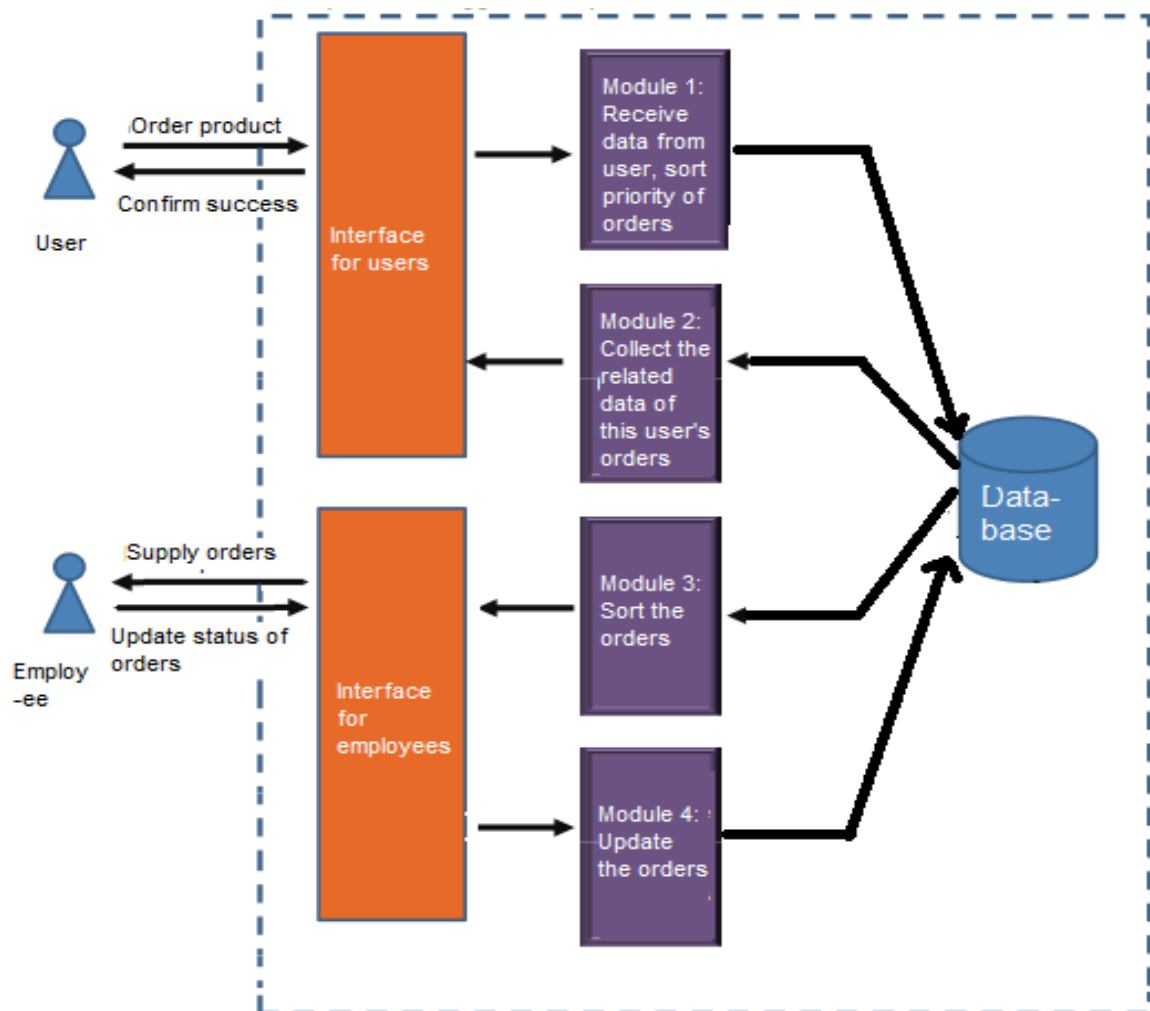
## 1. Giới thiệu

Tài liệu này miêu tả cơ bản cách Happy Lunch hoạt động, được ứng dụng vào hệ thống cửa hàng tiện lợi của Circle K để giải quyết vấn đề hàng đợi.

Ứng dụng giúp khách hàng (đặc biệt là sinh viên đại học Bách Khoa) giảm thiểu đáng kể thời gian xếp hàng khi họ muốn mua sản phẩm từ cửa hàng vào những giờ cao điểm (như giờ ăn trưa,...).

## 2. Tầm nhìn hệ thống

Hệ thống hoạt động cơ bản được mô tả trong Hình 1 dưới đây:



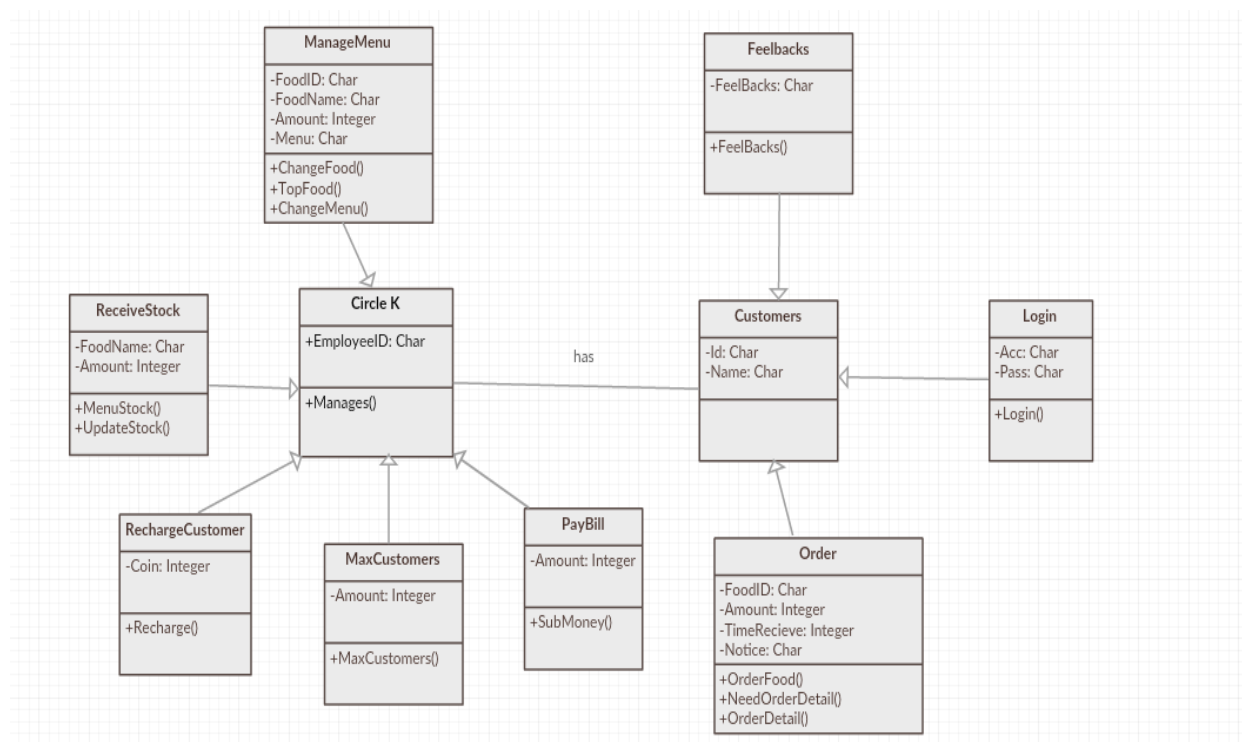
Hình 1 – Hệ thống Happy Lunch

Như được miêu tả ở Hình 1, hệ thống Happy Lunch bao gồm 4 module chính:

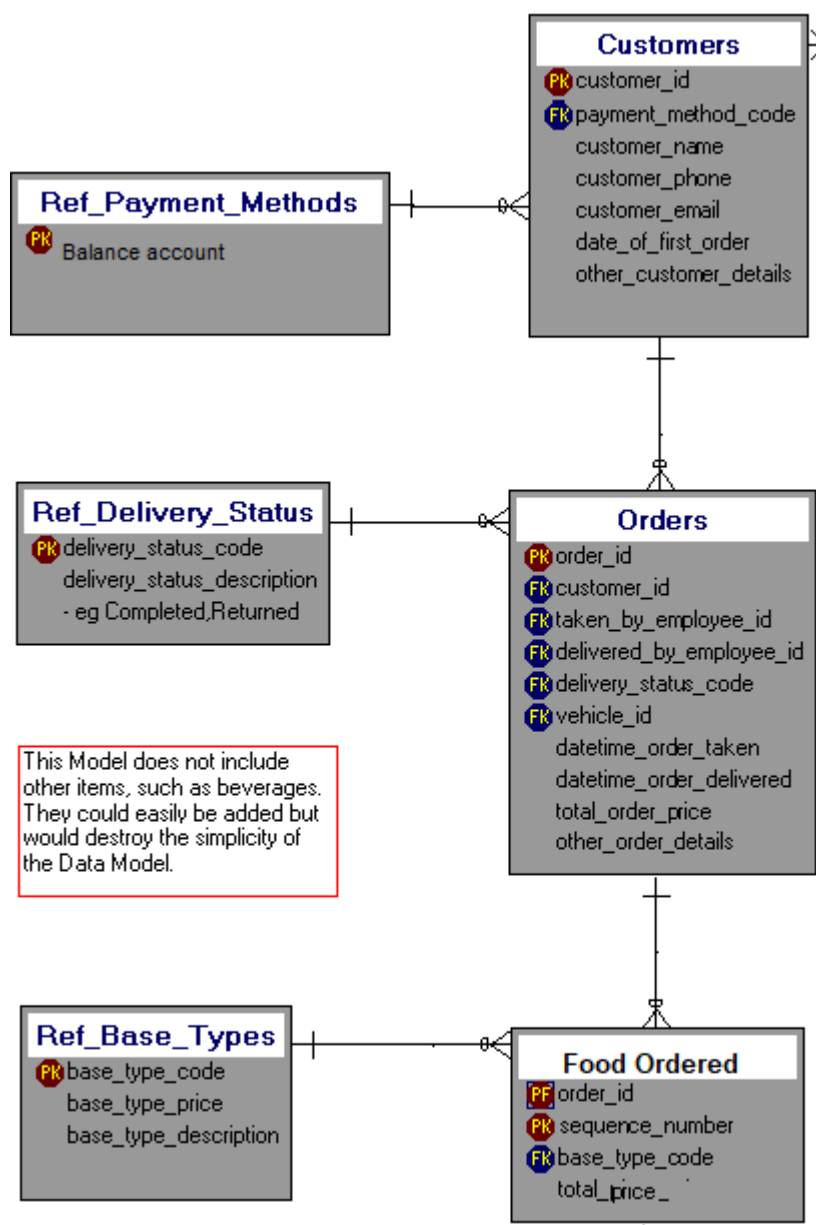
- *Module 1*: Nhận dữ liệu từ user, sắp xếp dữ liệu hợp lý rồi đưa vào database chính.
- *Module 2*: Thu thập dữ liệu có liên quan đến khách hàng từ database.
- *Module 3*: Thu thập dữ liệu từ database hiển thị lên màn hình của Employee. Lúc này dữ liệu đã được sắp xếp dễ dàng cho nhân viên thao tác công việc của mình
- *Module 4*: Cho phép nhân viên hiệu chỉnh thông tin, các trường của món ăn và cập nhật thẳng lên database chính

### 3. Các Diagram

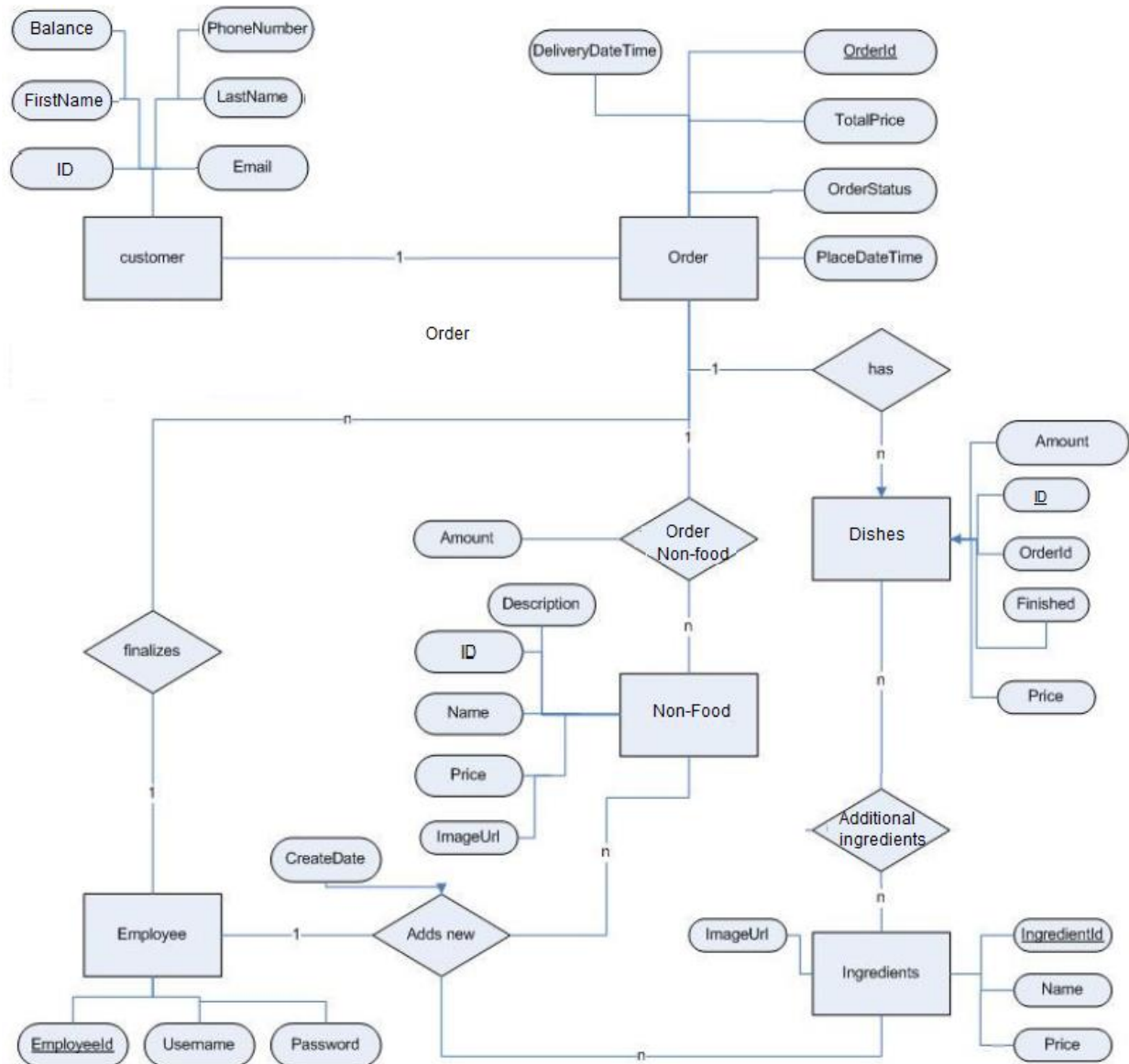
- Class diagram



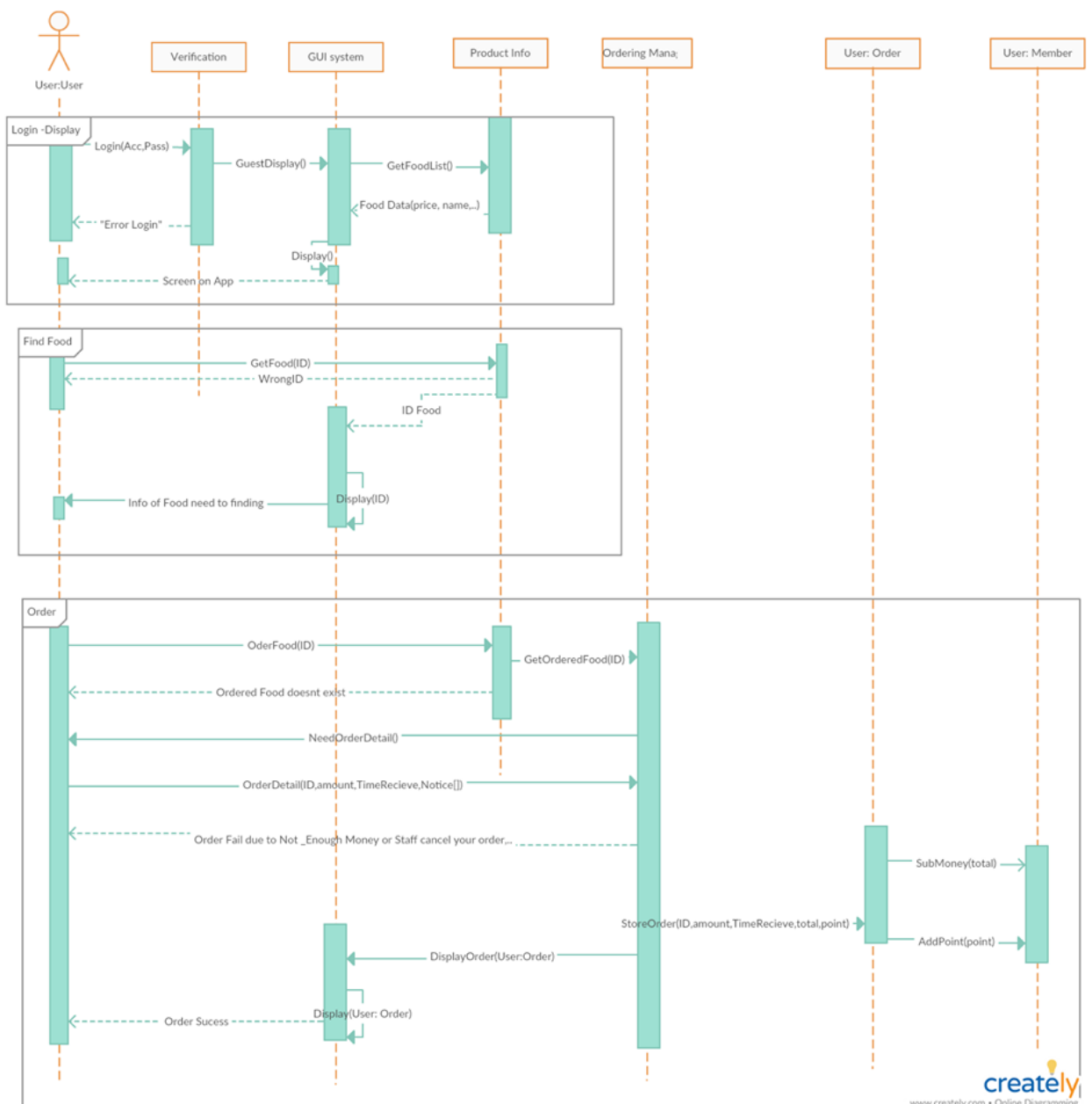
- Data diagram



- Entity Relationship Diagram



- Sequence Diagram

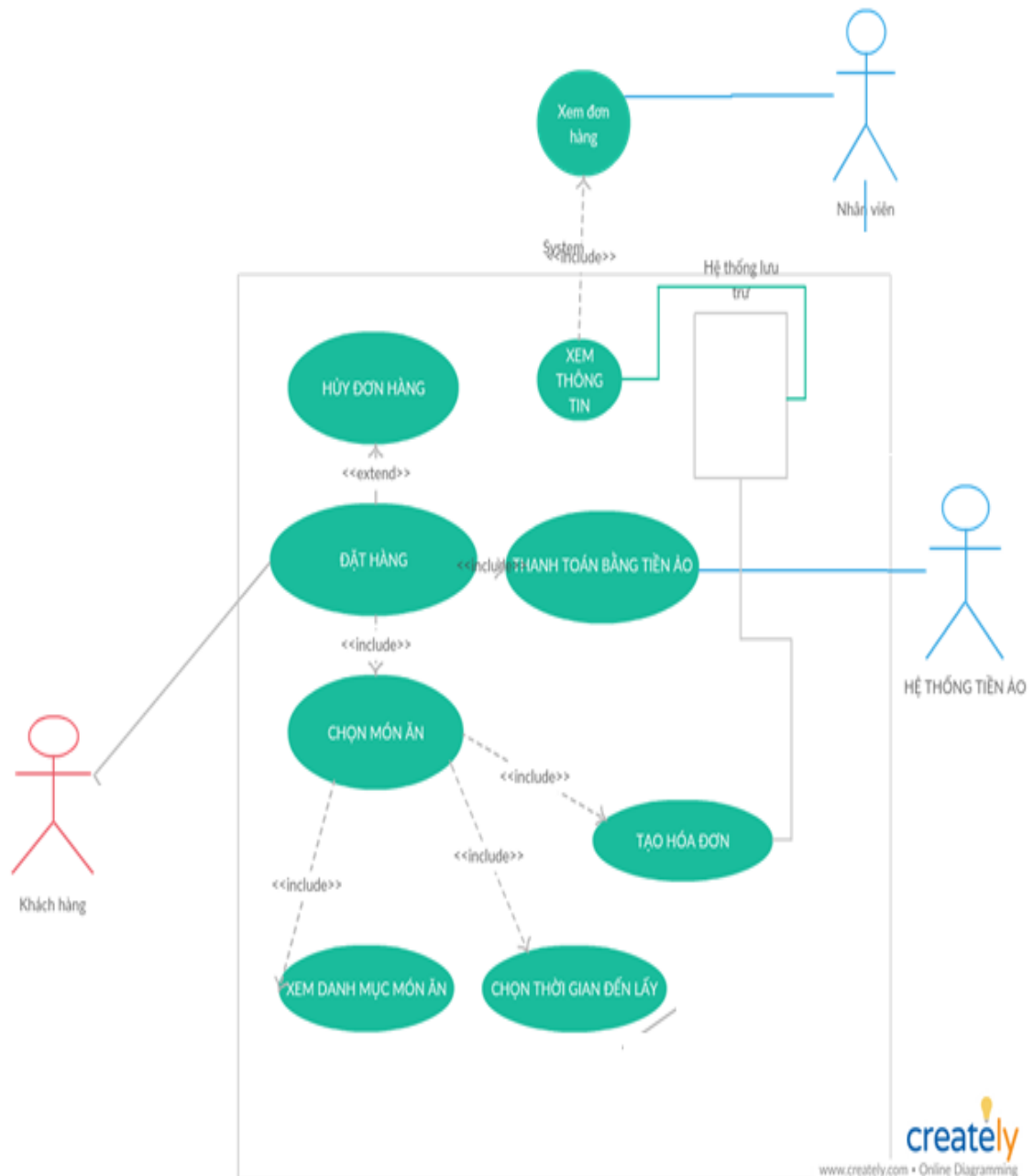




#### 4. Functional requirements

- Đặt hàng và những dịch vụ liên quan

- Use-case



- Use-case description

Actor	Khách hàng đặt hàng
Description	Khách hàng sẽ chọn món hàng mình muốn, chọn số lượng, và tạo ra hóa đơn nếu cần
Preconditions	Kiểm tra số tiền còn đủ? Số lượng hàng còn?
Normal Flow	1. Danh mục các món ăn sẽ hiện ra với giao diện dạng menu 2. Chọn món hàng mình mong muốn, đưa vào hóa đơn(giỏ hàng), Hệ thống sẽ lưu lại gói hàng và gửi cho nhân viên, để nhân viên chuẩn bị món hàng cho mình
Alternative Flows	- None

## • Thanh toán gói hàng qua tiền ảo

*\*Description*

Actor	Khách hàng khi thanh toán
Description	Khách hàng thanh toán gói hàng của mình
Preconditions	Gói hàng phải không trống
Normal Flow	1. Khách hàng nhấn button thanh toán trong gói hàng 2. Hệ thống sẽ lưu lại gói hàng và gửi cho nhân viên, đồng thời cập nhật lại lượng tiền ảo của khách hàng 3. Hệ thống sẽ gửi lại cho khách hàng một mã code để kiểm tra khi nhận hàng
Alternative Flows	2a. Nếu lượng tiền ảo của khách hàng không đủ, hệ thống sẽ thông báo thông qua Dialog

## • Hệ thống quản lý menu món

### a. Hiệu chỉnh loại thức ăn (thêm mới, cập nhật, xóa) loại thức ăn

*\*Description*

Actor	Nhân viên, hệ thống quản lý menu món
Description	Nhân viên hiệu chỉnh các loại món ăn (bao gồm các thao

	tác thêm mới, cập nhật, xóa ) được tùy chọn. Giá trị hiệu chỉnh sẽ được chỉnh sửa vào database
Data	Các thông số của nhân viên đã nhập vào
Stimulus	Nhân viên nhập thông số vào
Respond	Hệ thống xác nhận đã hiệu chỉnh hoặc trả về lỗi nếu thất bại
Comments	Nhân viên phải cung cấp đầy đủ tất cả dữ liệu trước khi gửi về hệ thống

### b. Hiệu chỉnh các trường một món ăn cụ thể

#### *\*Description*

Actor	Nhân viên, Hệ thống quản lý menu món
Description	Nhân viên có thể hiệu chỉnh các trường đặc tính của một món ăn cụ thể (bao gồm các thao tác thêm mới, cập nhật, xóa ) được tùy chọn. Giá trị hiệu chỉnh sẽ được chỉnh sửa vào database
Data	Các thông số của nhân viên đã nhập vào
Stimulus	Nhân viên nhập thông số vào
Respond	Hệ thống xác nhận đã hiệu chỉnh hoặc trả về lỗi nếu thất bại
Comments	Nhân viên phải cung cấp đầy đủ tất cả dữ liệu trước khi gửi về hệ thống

### c. Cập nhật tiền ảo cho khách hàng

#### *\*Description*

Actor	Nhân viên, Hệ thống quản lý menu món
Description	Nhân viên tăng lượng tiền ảo của khách hàng khi họ nộp tiền vào tài khoản ảo trực tiếp cho cửa hàng. Giá trị hiệu chỉnh sẽ được chỉnh sửa vào database
Data	Số tiền đã nộp
Stimulus	Nhân viên nhập thông số vào
Respond	Hệ thống xác nhận đã hiệu chỉnh hoặc trả về lỗi nếu thất bại
Comments	Nhân viên chỉ được tăng số tiền đã có trong database của khách hàng

- **Second Shop**

**a. Bán lại thức ăn đã đặt**

*\*Description*

Actor	Người dùng
Description	Người dùng khi đặt món ăn mà không muốn lấy nữa thì có thể đưa món ăn vào Second Shop với tổn thất 10% giá trị món ăn (nếu có người mua lại). Nếu không có người mua lại sẽ mất hoàn toàn 100% tiền.
Data	Món ăn cần bán lại
Stimulus	None
Respond	Hệ thống xác nhận đã đưa vào second shop
Comments	None

**b. Mua lại thức ăn trong second shop**

*\*Description*

Actor	Người dùng
Description	Người dùng muốn mua lại món ăn trong Second Shop với giá thấp hơn 10%
Data	Món ăn cần mua lại
Stimulus	None
Respond	Hệ thống xác nhận đã đưa vào giỏ hàng
Comments	None



## **5. Kết luận**

Trong tài liệu này, chúng tôi đã cung cấp một giải pháp giải quyết vấn đề hàng đợi bằng ứng dụng Happy Lunch. Ứng dụng có thể giúp người dùng vận hành một cách hiệu quả hơn cũng như tiết kiệm quỹ thời gian quý báu

Trong tương lai gần, chúng tôi mong rằng có thể hợp tác với hệ thống Circle K để biến đề xuất này thành hiện thực.