Task 1. Requirement elicitation

Task 1.1. Identify context, stakeholders, expected features, project scope

Context: In recent years, the COVID-19 pandemic has been spreading at an alarming rate in many countries around the world, causing massive economic consequences. As a result, those working in the restaurant industry have to face a lot of new challenges, one of which is to limit direct interaction between customers and clerks. In order to meet this demand, our team will develop a web-based system that facilitates the food ordering process. This can be a good replacement for a traditional menu, and it also offers a variety of features such as table reservations and payment processing. This is an excellent tool for restaurants looking to minimize face-to-face interaction between customers and clerks.

Stakeholders: restaurant owner, customer, chef, clerk, food safety specialist, online payment administrator.

Expected features: The system can be used by customers, clerks, managers and kitchen staffs of a restaurant. For each of these group, the system provides a different function.

- For customers, the system provides them with a web-based menu that they can make their food order on by scanning the QR code on each table.

- For clerks, they can use the system to process the orders of the customers and send them to kitchen staffs.

- For managers, the system provides them with an interface on which they can adjust the menu at any time, usually at the beginning of the day based on the availability of ingredients.

- For kitchen staffs, they receive the orders via the system.

Project scope: The project is a general web-based POS model that can be used by any restaurant, each restaurant has its own customized system configuration.

Task 1.2. Functional and non-functional requirements. Use-case diagram

Functional requirements:

- The menu can be modified by the manager at any time, which includes adding/removing dishes and changing the quantity available of each dish.

- The system provides customers with a website (accessed by scanning QR code, each table will have its own QR code which corresponds to the table's number) where customers can view the menu of the restaurant.

- The website allows customers to select their dishes with quantity and create orders.

- The system allows customers to pay for their order via online payment systems like credit cards, bank transfering or digital wallets when the order is created.

- The system allows clerks to review the orders of the customers and confirm them, then the order will be sent to the kitchen.

- The system provides kitchen staff with an interface to receive the orders of the customers.

Non-functional requirements:

- The system can supply 300 orders per day and 50 accesses at the same time.

- The system can run on multiple platforms: web, mobile, tablet,… and can support different browsers: Chrome, Edge, Firefox,...

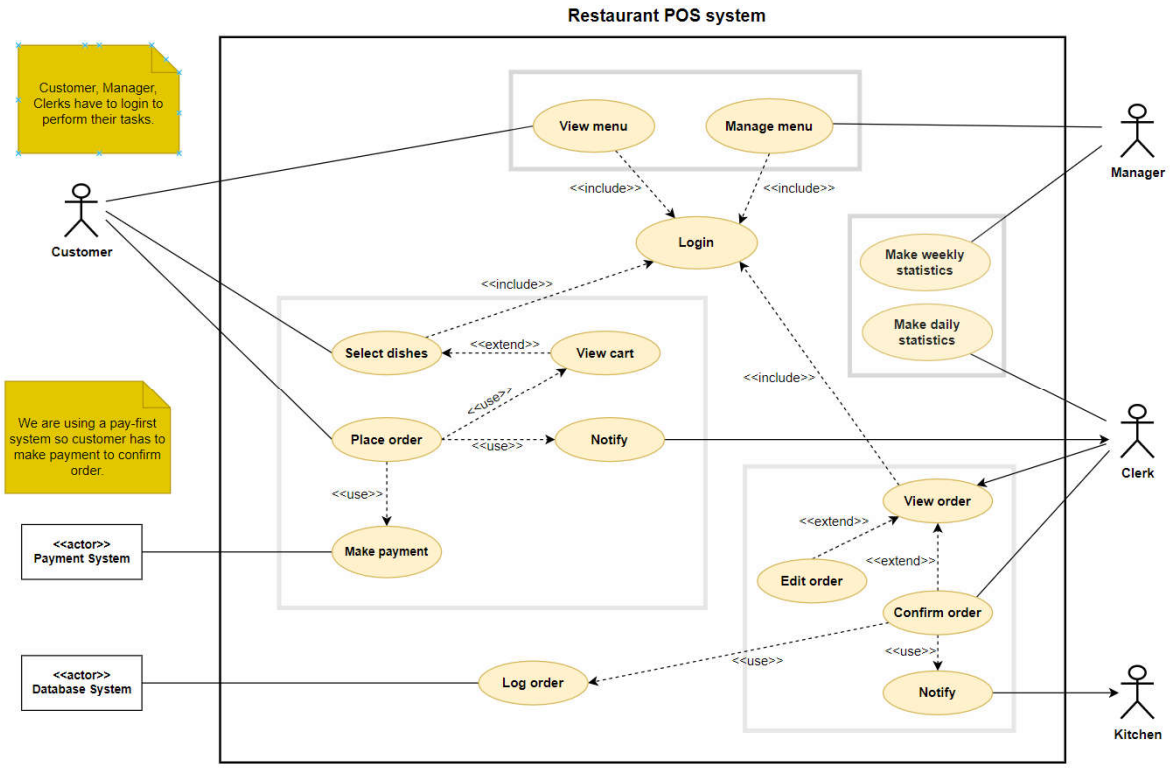
- The system should be operated around the clock and 7 days a week (24/7 mode).

- Response time should not exceed 3 seconds.

- Orders should be processed in less than 5 minutes.

- Customers can use the website immediately without any training.

Use-case diagram:



Actor list:

Name	Description
Customer	Customers of the restaurant, can view menu and make order, payment,…
Manager	Can manage the menu (update, add, delete) and make weekly statistics
Clerk	Can view, edit, confirm customers' orders and make daily statistics
Kitchen	Can receive orders from customers via clerks and make food accordingly
Payment system	Third-party online payment system (we use pay-first model)
Database system	Log orders of customers for clerks and managers to make statistics

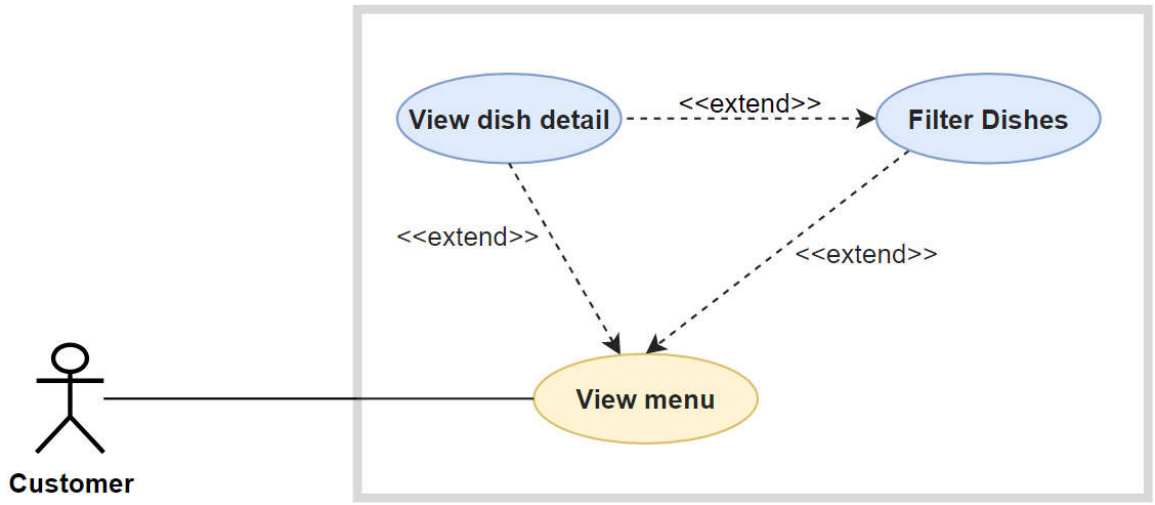
Use-case list:

Name	Actors	Description
Login	Customer, Manager, Clerk	Login to the system to access the website's functions based on account's role
View menu	Customer	View full food list of the restaurant or view by category, view dish details.
Manage menu	Manager	View and modify menu (update, add, delete)
Select dishes	Customer	Choose dishes from menu to add to cart
View cart	Customer	View all dishes added to cart
Place order	Customer	Place order to send to clerk
Make payment	Customer, Payment System	Pay for order using online payment systems (we use pay-first business model)
Notify	Customer, Clerk	Send order to clerk or kitchen
View order	Clerk	Review customer's order
Edit order	Clerk	Edit customer's order
Confirm order	Clerk	Confirm customer's order to send to kitchen
Log order	Database System	Save order to order history in database
Make weekly statistics	Manager	Make statistics using data from database at the end of the week
Make daily statistics	Clerk	Make statistics using data from database at the end of the day

Task 1.3. Specific feature use-case diagram and description

Feature: Viewing menu

Use-case diagram:



Note: (Yellow: primary use-case, Blue: secondary use-case)

Use-case scenarios:

- Use-case: **View menu**

Name	**View menu**
Actor	Customer
Description	The customer wants to see the restaurant menu.
Trigger event	Customer clicks on “Menu” section on website’s navigation bar.
Pre-conditions	Customer is logged in and is at the homepage.
Post-conditions	Customer can see restaurant’s menu with full list of dishes.
Normal flow	1. Customer clicks on the “Menu” section on website’s navigation bar. 2. System redirects customer to Menu page 3. Customer can see the blank Menu 4. System loads food list from database 5. Customer waits less than 2 seconds 6. System displays the list of food 7. Customer can see the list of food
Exceptions	At step 5, customer waits for too long due to limited/interrupted network connection
Alternative flow	At step 2, system reloads the menu page *Continue step 3 in the normal flow*
Extension points	View dish detail, Filter dishes

- Use-case: **Filter dishes**

Name	**Filter dishes**
Actor	Customer
Description	Customer wants to see the dish list of a specific category.
Trigger event	Customer clicks on a specific button from a list of buttons on the menu, each has its label corresponding to a specific category.
Pre-conditions	Customer is at the menu page with full dish list or dish list of a specific category.
Post-conditions	Customer can see full list of dishes within the category that he clicks on.
Normal flow	1. Customer clicks on a specific button from the category list. 2. System loads the food list of that category. 3. Customer waits less than 1 second. 4. System displays the food list of that category. 5. Customer can see all dishes of that category.
Exceptions	Exception 1: At step 5, customer cannot see any dish within that category because there is no food of that catefory added. Exception 2: At step 3, customer waits for too long due to limited/interrupted network connection.
Alternative flow	Alternative 1: At step 1, customer select a different category. *Continue step 2 in the normal flow* Alternative 2: At step 1, customer reloads the page *Continue step 2 in the normal flow*
Extension point	View dish detail

- Use-case: **View dish detail**

Name	**View dish detail**
Actor	Customer
Description	Customer wants to see the more information about a dish.
Trigger event	Customer clicks on a specific dish on the menu.
Pre-conditions	Customer is at the menu page with full dish list or dish list of a specific category.
Post-conditions	Customer can see the detailed description of the dish he clicks on.
Normal flow	1. Customer clicks on a specific dish on the menu. 2. System opens a pop-up 3. Customer see the pop-up opened. 4. System loads dish's details from database. 5. Customer waits less than 0.5 second. 6. System displays the details of the dish. 7. Customer can see the detail of the dish.
Exceptions	At step 5, customer waits for too long due to limited/interrupted network connection.
Alternative flow	At step 1, customer closes the pop-up and clicks on the dish again. *Continue step 2 in the normal flow*