Use Case Diagram for the Restaurant Management System

Learn how to define use cases and create the corresponding use case diagram for the restaurant management system.

We'll cover the following

Let's build the use case diagram for the restaurant management system and understand the relationship between its different components.

First, we'll define the different elements of our restaurant, followed by the complete use case diagram of the system.

System

Our system is a "restaurant."

Actors

Let's define the main actors of our restaurant management system.

Primary actors

- **Guest:** This is the restaurant's primary actor who can view the menu, place orders, and make payments.
- **Receptionist:** This actor is responsible for reserving tables and updating the table reservation status.
- **Waiter:** This actor takes the order from the guest and processes the payment.
- **Manager:** This actor acts as the admin of the system and can perform all tasks that a receptionist and a waiter can perform. Other than that, this actor updates the menu and sets the price of food items.

Use cases

In this section, we'll define the use cases for the restaurant management system. We've listed the use cases according to their respective interactions with a particular actor.

> **Note:** You will see some use cases occurring multiple times, because they are shared among different actors in the system.

Guest

- **View menu:** To view the food items available in the restaurant
- **Place order:** To place the food order in the restaurant
- **Add/update order item:** To add or remove a food item from the order
- **Cancel order:** To cancel an existing order
- **View order:** To view the item included in the order
- **Pay bill via cash:** To pay the food bill with cash
- **Pay bill via card:** To pay the food bill with the card
- **Pay bill via check:** To pay the food bill with the check

- **Reserve table:** To reserve a table for a guest
- **Update/cancel reservation:** To change the table or cancel the reservation

Manager

- **Add/modify menu section:** To add a new section of the food type in the menu
- **Add/modify menu item:** To add a new food item in the food section
- **Set menu item price:** To set the price of a food item
- **Generate report:** To generate an analytical report of orders, inventory, and more
- **Add/ Update Tables chart:** To keep an updated record of the table availability status
- **Reserve table:** To reserve a table for a guest
- **Update/cancel reservation:** To change the table or cancel the reservation
- **View menu:** To view the food item available in the restaurant
- **Place order:** To place a food order in the restaurant
- **Add/update order item:** To add or remove a food item from the order
- **Cancel order:** To cancel an existing order
- **View order:** To view the item included in the order
- **Process payment:** To generate a food bill from the system and receive payment from the guest

Receptionist

- **Add/ Update Tables chart:** To keep an updated record of the table availability status
- **Reserve table:** To reserve a table for a guest
- **Update/cancel reservation:** To change the table or cancel the reservation

Waiter

- **View menu:** To view the food item available in the restaurant
- **Place order:** To place the food order in the restaurant
- **Add/update order item:** To add or remove a food item from the order
- **Cancel order:** To cancel an existing order
- **View order:** To view the item that is included in the order
- **Process payment:** To generate a food bill from the system and receive payment from the guest.

Relationships

We describe the relationships between and among actors and their use cases in this section.

Generalization

- The manager is responsible for the receptionist and the waiter. It also has access to everything they both have. Therefore, the "Manager" has a generalization relationship with both "Receptionist" and "Waiter."

Associations

The below table shows the association relationship between actors and their use cases.

Guest	Receptionist	Waiter	Manager
Pay bill via card	Add/ Update Tables chart	View menu	Add/modify menu.
Pay bill via cash	Reserve table	View order	Add/modify menu i
Pay bill via check	Update/cancel reservation	Place order	Generate repo
View menu		Add/update order item	Set menu item p
View order		Cancel order	View menu
Place order		Process Payment	View order
Add/update order item			Place order
Cancel order			Add/update order
Reserve table			Cancel order
Update/cancel reservation			Add/ Update Table
			Reserve table
			Update/cancel rese
			Print booking
			Cancel bookir
			Process Payme

Include

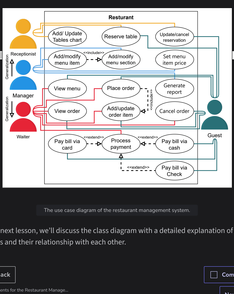
- Whenever the manager adds a new menu item, the menu section is modified. Therefore, the "Add/ modify menu item" use case has an include relationship with the "add/modify menu section" use case.

Extend

- If the payment is processed, it will be either by card, cash, or check. Therefore, the "Process payment" use case has an extend relationship with the "Pay bill via cash," "Pay bill via card," and the "Pay bill via check" use cases.

Use case diagram

Here's the use case diagram of the restaurant management system:



The use case diagram of the restaurant management system

In the next lesson, we'll discuss the class diagram with a detailed explanation of all classes and their relationship with each other.