SUB: Software Engineering
Branch: COMPS B

Name :	**Chetan Deepak Patil**
UID :	**2023301012**
Expt no :	**02**
Aim :	**Use Case Diagram for Online Booking System**

Table 1: List of Actors

Actor	Description
Customer	The Customer is the individual who uses the food delivery system to order meals from various restaurants. They are the primary end-user and their experience with the system is crucial for its success.
Restaurant	The Restaurant is a business entity that participates in the food delivery system by offering food items to customers. They are essential for providing the menu options and preparing the ordered food.
Admin	The Admin is responsible for overseeing and managing the food delivery system's operations. They ensure that the system runs smoothly, handle user issues, and manage system-wide settings.

Table 2: List of Use-Cases

#	Use Case	Description
UC1	View Services	Customers can search and view a list of services offered by various service providers, along with details and descriptions.
UC2	Book Appointment /Reservations	Customers can book appointments or make reservations for services of their choice, based on availability.
UC3	Cancel Booking	Customers can cancel an existing booking or appointment, based on cancellation policies.
UC4	Make Payment	Customers can securely make payments for their booked appointments or reservations using various payment methods.
UC5	Send Notifications	The system sends notifications to customers regarding their bookings, cancellations, or any updates related to their appointments.
UC6	View Booking	Service Providers can view all the bookings made by customers for their services.
UC7	Manage Availability	Service Providers can manage their availability schedules on the platform to accept or decline bookings.
UC8	Manage Users	Admin can manage customer and service provider accounts, including add removing users from the system.
UC9	Manage Service Providers	Admin can manage service providers, including their details, services offered, and availability.
UC11	Handle Complaints	Admin can handle and resolve complaints raised by customers or service providers regarding services or bookings.

Use Case Scenarios:

Table 3: UC1 – View Services

Use Case:	UC1. View Services	
Goal:	Allow customers to find and view a list of services offered by various service providers.	
Actors:	Customer	
Pre-condition:	The customer is logged into the system and has access to the restaurant listing.	
Post-condition:	The customer can view the list of available services and select one to proceed with booking.	
Mainline Scenario:	**Actor Actions**	**System Actions**
	1. The customer logs into the system.	1. The system authenticates the customer and provides access.
	2. The customer navigates to the services section.	2. The system displays a list of available services.
	3. The customer browses the list of available services.	3. The system displays the list of services matching the criteria.
	4. The customer selects a service to view more details.	4. The system retrieves and shows detailed information for the selected service.
Alternate Flows:	If no services are available, the system displays a "No services found" message.	

Table 4: UC2 – Book Appointment | Reservations

Use Case:	UC2. Book Appointment	Reservations		
Goal:	Enable customers to book appointments or make reservations for selected services.		
Actors:	Customer		
Pre-condition:	The customer has viewed the available services and selected one.		
Post-condition:	The customer successfully books an appointment or reservation.		
Mainline Scenario:	**Actor Actions**	**System Actions**	
	1. The customer selects a service to book.	1. The system displays booking options for the selected service.	
	2. The customer chooses a date and time for the appointment. 3. The customer provides necessary details (e.g., contact information). 4. The customer confirms the booking.	2. The system checks the availability of the chosen date and time. 3.The system collects the customer's details. 4. The system confirms the booking and sends a confirmation notification to the customer.	
Alternate Flows:	If the selected time slot is unavailable, the system prompts the customer to select a different time.		

Table 5: UC3 -Make Payment

Use Case:	**Use Case:** UC3. Make Payment	
Goal:	Facilitate secure payment for the booked service using various payment methods.	
Actors:	Customer, Payment Gateway	
Pre-condition:	The customer has completed the booking process and is ready to make a payment.	
Post-condition:	Payment is processed successfully, and the transaction is recorded.	
Mainline Scenario:	**Actor Actions** 1. The customer proceeds to the checkout page. 2. The customer selects a payment method. 3. The customer enters payment details. 4. The customer confirms the payment.	**System Actions** 1. The system presents the payment options. 2. The system processes the payment securely through the payment gateway. 3. The system updates the booking status upon successful payment. 4. The system sends a payment confirmation receipt to the customer.
Alternate Flows:	If payment fails, the system prompts the customer to re-enter payment details or choose an alternative payment method.	

Table 6: UC4 – Manage Availability

Use Case:	UC4. Manage Availability	
Goal:	Allow service providers to manage and update their availability for bookings.	
Actors:	Service Provider	
Pre-condition:	The service provider is logged into the system.	
Post-condition:	The service provider's availability is updated.	
Mainline Scenario:	**Actor Actions**	**System Actions**
	1. The service provider logs into the system. 2. The service provider accesses the availability management section. 3. The service provider updates available time slots. 4. The service provider saves the changes.	1. The system authenticates the service provider. 2. The system displays the availability management interface. 3. The system updates the availability based on the changes made. 4. The system confirms that availability has been updated successfully.
Alternate Flows:	If an error occurs while updating availability, the system displays an error message and prompts the service provider to try again.	

Table 7: UC5 – Cancel Booking

Use Case:	UC5. Cancel Booking	
Goal:	Allow customers to cancel their existing bookings.	
Actors:	Customer	
Pre-condition:	The customer has a confirmed booking.	
Post-condition:	The booking is canceled, and the customer receives a confirmation of cancellation.	
Mainline Scenario:	**Actor Actions**	**System Actions**
	1. The customer accesses their booking history. 2. The customer selects a booking to cancel. 3. The customer confirms the cancellation.	1. The system displays the customer's booking history. 2. The system processes the cancellation request. 3. The system updates the booking status to "Canceled" and sends a cancellation confirmation to the customer.
Alternate Flows:	If the cancellation period has expired, the system notifies the customer that cancellation is no longer possible.	

Table 8: UC6 – View Bookings

Use Case:	UC6. View Bookings	
Goal:	Allow service providers to view and manage customer bookings.	
Actors:	Service Provider	
Pre-condition:	Service provider is registered and logged into the system.	
Post-condition:	Booking details are successfully displayed to the service provider.	
Mainline Scenario:	**Actor Actions**	**System Actions**
	1. Service Provider logs into their account.	1. The system authenticates the service provider and provides access to booking data.
	2. Service Provider selects "View Bookings."	2. The system displays a list of all current and future bookings.
	3. Service Provider reviews booking details.	3. The system provides detailed information for each booking, including customer information.
Alternate Flows:	The system displays a message: "No bookings are available at this time."	

Table 9: UC7 – Generate Reports

Use Case:	UC7: Generate Reports	
Goal:	Allow the admin to generate reports for analysis.	
Actors:	Admin	
Pre-condition:	Admin is logged into the system.	
Post-condition:	The report is successfully generated and available for download or view.	
Mainline Scenario:	**Actor Actions**	**System Actions**
	1. Admin logs into the system.	1. The system authenticates the admin and provides access to administrative functions
	2. Admin selects "Generate Reports."	2. The system displays report generation options (e.g., time period, data type).
	3. Admin configures the report settings.	3. The system processes the request and generates the report.
	4. Admin views or downloads the generated report.	4. The system provides the generated report in the selected format (e.g., PDF, Excel).
Alternate Flows:	The system displays an error message: "Report generation failed. Please try again later."	

Table 10: UC8 – Handle Complaints

Use Case:	UC8. Handle Complaints	
Goal:	Address complaints raised by users or service providers.	
Actors:	Admin	
Pre-condition:	Admin is logged into the system. Complaints exist in the system.	
Post-condition:	Complaint status is updated, and the concerned party is notified of the action taken.	
Mainline Scenario:	**Actor Actions** 1. Admin accesses the complaints section. 2. Admin reviews and selects a complaint to handle. 3. Admin takes action (e.g., resolves, escalates).	**System Actions** 1. The system displays all complaints raised by customers or service providers. 2. The system retrieves the details of the selected complaint. 3. The system updates the complaint status and notifies the user or service provider accordingly.
Alternate Flows:	The system displays a message: "No complaints are currently available.".	

Table 11: UC9 – Send Notifications

| Use Case	UC1. Send Notifications
Goal	To notify customers about their bookings, cancellations, or any updates related to their appointments.
Actors	System, Customer
Pre-condition	The customer must have an active booking or be subscribed to notifications for updates.
Post-condition	The customer receives notifications regarding their booking status, cancellations, or other relevant updates.
Mainline Scenario	1. The system identifies events that require customer notification (e.g., booking confirmation, cancellation).
Alternate Flows	If the notification fails (e.g., due to incorrect contact details), the system retries sending the notification and alerts the admin of the issue.

Table 12: UC10 - Manage Users

Use Case:	UC10. Manage Users	
Goal:	Allow the admin to manage customer and service provider accounts, including adding, updating, or removing users from the system.	
Actors:	Admin	
Pre-condition:	The admin is logged into the system with sufficient privileges to manage user accounts	
Post-condition:	The user's account information is added, updated, or removed from the system.	
Mainline Scenario:	**Actor Actions**	**System Actions**
	1. The admin logs into the system.	1. The admin logs into the system.
	2. The admin navigates to the user management section.	2. The admin navigates to the user management section.
	3. The admin selects an option to add, update, or remove a user.	3. The admin selects an option to add, update, or remove a user.
	4. The admin confirms the changes.	4. The admin confirms the changes.
Alternate Flows:	If validation fails (e.g., missing required fields), the system displays an error message and prompts the admin to correct the details before proceeding.	

Table 12: UC11 - Manage Service Providers

Use Case:	UC10. Manage Service Providers	
Goal:	Allow the admin to manage service providers, including their details, services offered, and availability.	
Actors:	Admin	
Pre-condition:	The admin is logged into the system with sufficient privileges to manage service provider details.	
Post-condition:	The service provider's details are added, updated, or removed from the system.	
Mainline Scenario:	**Actor Actions**	**System Actions**
	1. The admin logs into the system.	1. The admin logs into the system.
	2. The admin navigates to the service management section.	2. The admin navigates to the user management section.
	3. The admin selects an option to add, update, or remove a service prov	3. The admin selects an option to add, update, or remove a user.
	4. The admin confirms the changes.	4. The admin confirms the changes.
Alternate Flows:	If the admin is unable to resolve the complaint, they can escalate it, and the system logs the escalation and notifies a higher authority or specialized department for further handling.	

Use Case Diagram:

Use Case Diagram for Online Booking System
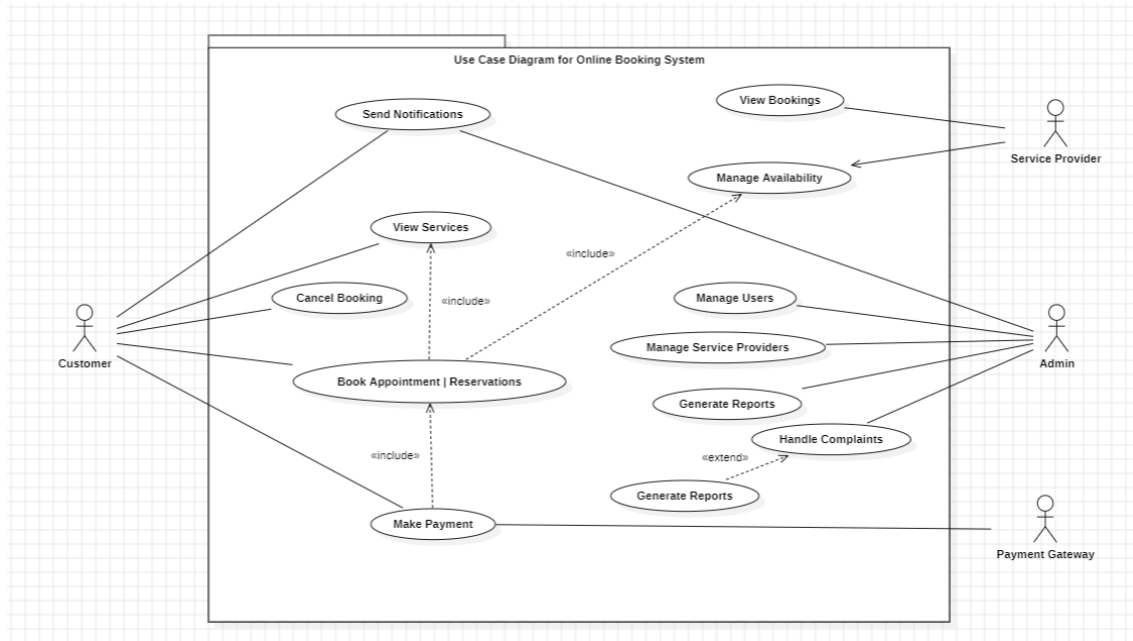
Conclusion:

From this experiment, we have gained a thorough understanding of how to design and implement a use case diagram by analysing and defining various use case scenarios. This process has helped us effectively map out the interactions between different actors and the system, clarifying the functional requirements and user actions for the online booking system.