ACTIVITY - 2

NAME: VIKASINI R

REG.NO: 23CS062

CLASS: BE-CSE-C

SUBJECT: OBJECT ORIENTED SOFTWARE ENGINEERING

TITLE: ONLINE CATERING HUB.

1. Give students a problem statement and ask them to:

- **Identify functional and non-functional requirements**
- **Create Use Case diagrams**
- **Design at least two more UML diagrams (e.g., Class Diagram, Activity Diagram)**

1. Functional and Non-Functional Requirements

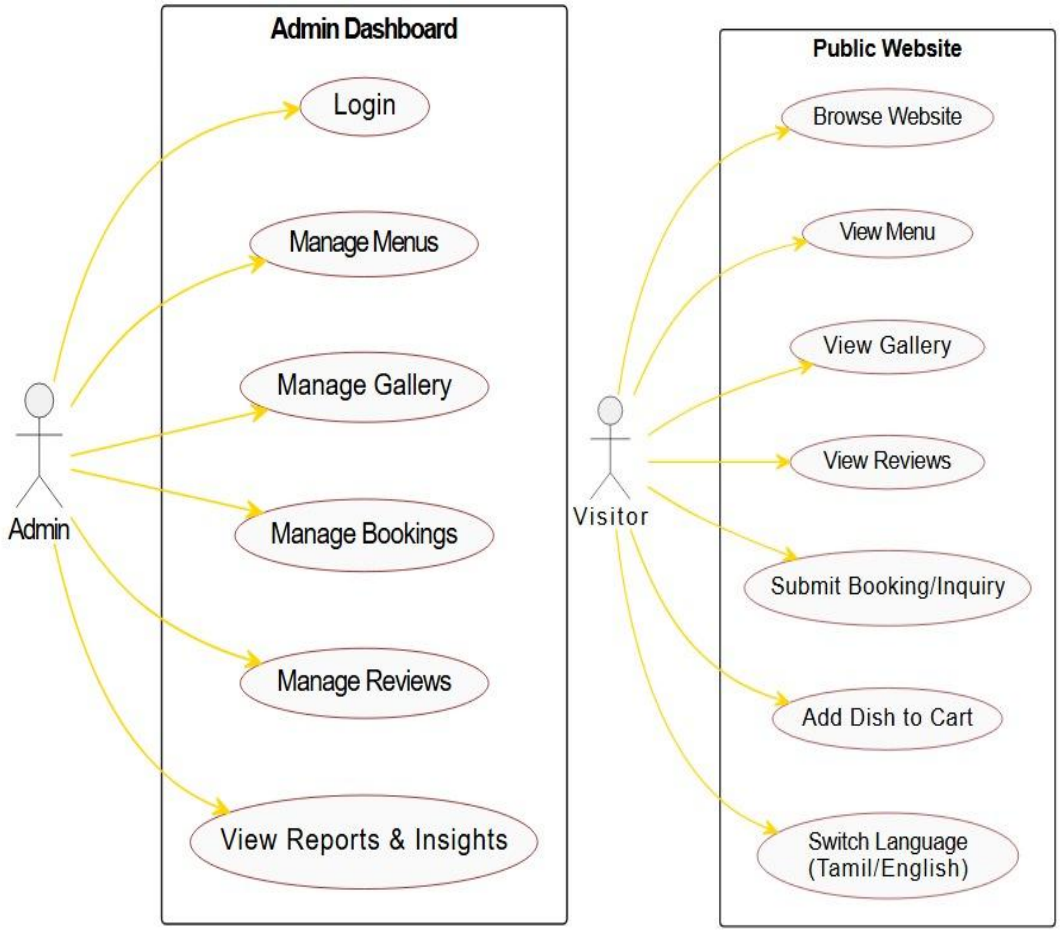
Functional Requirements (What the system must do):

- Users can view homepage, menu, gallery, reviews, and contact/booking pages.
- Users can switch language between Tamil and English instantly.
- Users can browse dishes by category (Starters, Main Course, Desserts, Beverages).
- Users can add dishes (names only) to a cart and proceed to booking.
- Users can submit booking/inquiry forms with event details.
- Admin can log in to access the dashboard.
- Admin can add, edit, or delete menu items (with images + Tamil/English names).
- Admin can upload or delete gallery images with captions.
- Admin can view, approve, or delete customer reviews.
- Admin can view booking requests and mark them as approved/pending.
- Admin can see reports (bookings per month, most booked event types, most selected dishes).
- Admin can mark available/unavailable dates on a calendar.

Non-Functional Requirements (How the system should be):

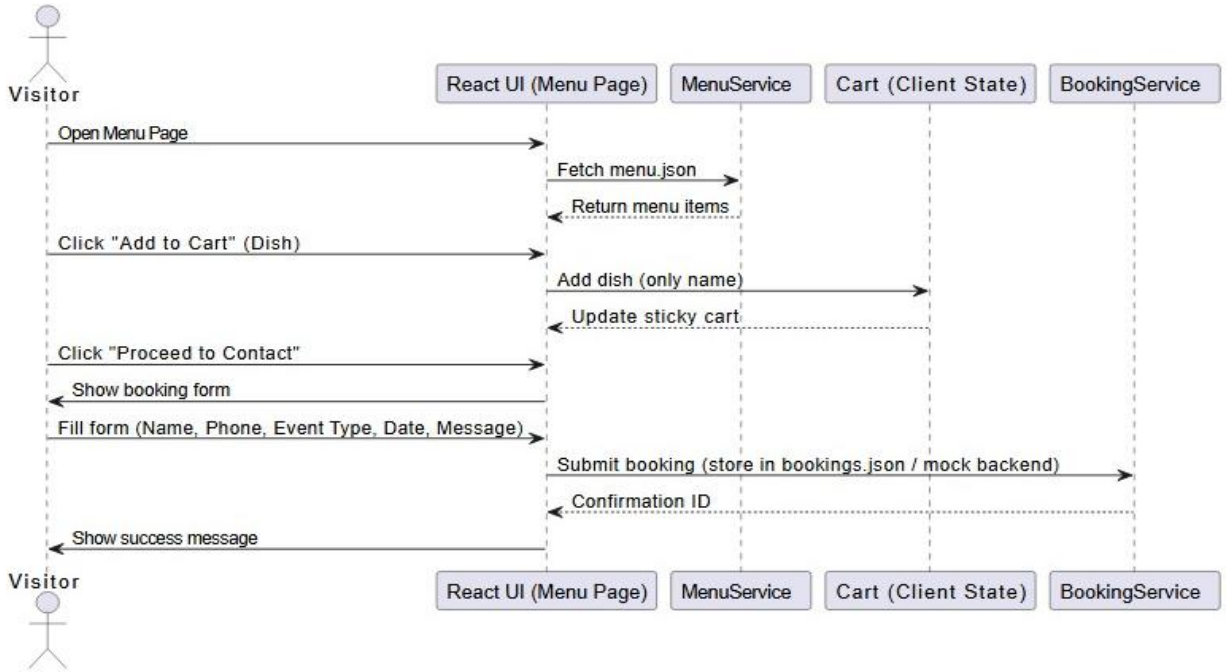
- The website should have a festive, modern, and professional design.
- The interface should be easy to use and mobile-friendly.
- Data (menu, reviews, bookings) should load quickly from JSON/backend.
- The system should be reliable and handle many users at the same time.
- Only admins should have access to manage menu, bookings, reviews, and reports.
- Smooth animations, hover effects, and responsive design must be included.
- Support for both Tamil and English must be instant and accurate.

2. Create Use Case diagrams:



3. **Design at least two more UML diagrams (e.g., Class Diagram, Activity Diagram)**

SEQUENCE DIAGRAM:



CLASS DIAGRAM: