# CS 255 System Design Document Template

## UML Diagrams

### UML Use Case Diagram



### UML Activity Diagrams

**The two use cases I have chosen are:**

1. Purchase Package
   1. Actor: Customer
2. Disable Package
   1. Actor: Owner

**Their respective activity diagram is as follows:**

1. Use Case: Purchase Package
   1. Actor: Customer
   2. Activity Diagram: 
2. Use Case: Disable Package
   1. Actor: Owner
   2. Activity Diagram: 

### UML Sequence Diagram

* Use Case: Disable Package
* Actor: Owner
* Sequence Diagram:
* 

### UML Class Diagram



## Technical Requirements

**Hardware Requirements:**

1. Servers:
   1. Web Server: A strong server for hosting the website (like a dedicated server or AWS EC2).
   2. Database Server: Another server for managing the database securely.
2. Client Devices:
   1. User Devices: Modern computers or mobile devices (smartphones, tablets) that can run web browsers well.
3. Network Equipment:
   1. Routers and Switches: Reliable network devices for fast internet access to the servers.

**Software Requirements:**

1. Operating Systems:
   1. Servers: Linux (Ubuntu) for stability and security.
   2. Client Devices: Any modern OS like Windows, macOS, or Linux.
2. Web Server Software:
   1. Server: Node.js with Express.js for handling web operations.
3. Database Management System (DBMS):
   1. Database: MongoDB or MySQL for storing and managing data.
4. Web Browsers:
   1. Client Browsers: Google Chrome, Mozilla Firefox, Safari, Microsoft Edge.
5. Development Tools:
   1. IDE/Code Editor: Eclipse, Visual Studio Code.
   2. Version Control: Git for managing source code

**Tools and Framework:**

1. Frontend Framework:
   1. Framework: React.js or Angular for building the user interface.
2. Backend Framework:
   1. Framework: Express.js (part of node.js) for backend development.
3. API Management:
   1. APIs: RESTful APIs to handle interactions between frontend and backend.
4. Security Tools:
   1. Encryption: HTTPS for secure data transfer.
   2. Authentication: OAuth 2.0 or JWT for user login and permissions.

**Infrastructure Requirements:**

1. Cloud Services:
   1. Hosting: AWS, Google Cloud Platform, or Microsoft Azure for scalable cloud hosting.
   2. CDN: Content Delivery Network for fast content delivery worldwide.
2. Database Hosting:
   1. Managed Services: AWS RDS, Google Cloud SQL, or MongoDB Atlas for database hosting.
3. Monitoring and logging:
   1. Monitoring Tools: Prometheus and Grafana for performance monitoring.
   2. Logging Tools: ELK Stack (Elasticsearch, Logstash, Kibana) for logging and analytics.
4. Backup and Recovery:
   1. Backup Solutions: Automated backups using cloud service for data recovery in case of issues.