# CS 255 System Design Document Template

## UML Diagrams

### UML Use Case Diagram

A graph with text on it

AI-generated content may be incorrect.

### UML Activity Diagrams

A screenshot of a diagram

AI-generated content may be incorrect.A diagram of a computer

AI-generated content may be incorrect.

### UML Sequence Diagram

A diagram with pink squares and black text

AI-generated content may be incorrect.

### UML Class Diagram

A diagram with pink squares

AI-generated content may be incorrect.

## Technical Requirements

*Based on the client interview and the system diagrams created, I've identified the following technical requirements for the DriverPass system:*

***Hardware Requirements***

1. ***Server Infrastructure****: The system requires cloud-based servers to host the web application, database, and other components. This aligns with Ian's requirement for a web-based, cloud solution that minimizes technical management overhead.*
2. ***Client Devices****: The system must be accessible from various devices including desktops, laptops, tablets, and smartphones, as Liam mentioned the need to access data from anywhere using different devices.*
3. ***Printing Capability****: Local printing support is needed for generating activity reports, as Liam mentioned wanting to "print an activity report" to track system activities.*

***Software Requirements***

1. ***Web Application****: A responsive web application that functions correctly on different browsers and device types.*
2. ***Database Management System****: A robust relational database system to store user information, appointment details, package information, test results, and tracking data.*
3. ***Authentication System****: Secure login and authentication system with password reset functionality as requested by Ian.*
4. ***Reporting Tool****: Capability to generate and export reports in formats compatible with Excel, as Liam needs to "download the reports" and work on them at home.*
5. ***User Role Management****: Software to manage different user roles and access rights as mentioned by Ian regarding "different employees at the company with different rights and roles."*
6. ***Notification System****: Email or in-app notification system to alert users about appointments, test results, and DMV updates.*

***Integration Requirements***

1. ***DMV Integration****: Interface to receive updates from the DMV regarding rules, policies, and sample questions, as Liam mentioned needing to "be connected to the DMV" for updates.*
2. ***Payment Processing****: Secure payment processing for customers purchasing training packages.*
3. ***Email Integration****: Email functionality for sending confirmations and notifications.*

***Security Requirements***

1. ***User Authentication****: Secure login process with password reset capability.*
2. ***Role-Based Access Control****: Different access levels for customers, drivers, secretary, IT officer, and owner.*
3. ***Activity Tracking****: System to track and record all user actions, especially for reservations, as Liam emphasized knowing "who made a reservation, who canceled it, who modified it last."*
4. ***Secure Data Storage****: Encrypted storage of sensitive information like credit card details.*
5. ***Data Backup****: Regular automated backups of all system data.*

***Network Requirements***

1. ***Internet Connectivity****: Reliable internet connection for cloud access.*
2. ***Firewall Protection****: Network security measures to protect against unauthorized access.*
3. ***API Security****: Secure communication channels for DMV integration.*

***Performance Requirements***

1. ***Scalability****: Ability to handle growing numbers of users and data.*
2. ***Response Time****: Quick system response for all operations, especially appointment scheduling.*
3. ***Availability****: 24/7 system availability with minimal downtime for maintenance.*

***Development Tools and Environment***

1. ***Development Framework****: A modern web development framework supporting responsive design.*
2. ***Version Control****: System for tracking code changes during development.*
3. ***Testing Environment****: Separate testing environment to verify functionality before deployment.*
4. ***Documentation Tools****: Software for creating system documentation and user manuals.*