# CS 255 System Design Document Template

This template lays out all the different sections that you need to complete for Project Two. Each section has guidance to prompt your thinking. You will need to continually reference the interview transcript as you work to make sure that you are addressing your client’s needs. There is no required length for the final document. Instead the goal is to complete each section based on what your client’s needs are. Remove this note when you are finished, and replace all bracketed text with the relevant information.

## UML Diagrams

### UML Use Case Diagram

A diagram of a driver pass

AI-generated content may be incorrect.

### UML Activity Diagrams

* ***Login/Register Use Case***
* ***Schedule/edit appointment Use Case***

A diagram of a login account

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

AI-generated content may be incorrect.

### UML Sequence Diagram

*A diagram of a diagram

AI-generated content may be incorrect.*

A diagram of a computer

AI-generated content may be incorrect.

## Technical Requirements

**Hardware Requirements:**

* Web server with sufficient memory and processing power to handle concurrent users (e.g., 8+ GB RAM, multi-core CPU)
* Cloud-hosted database server for data storage and retrieval (e.g., AWS RDS or Google Cloud SQL)
* End-user devices: smartphones, tablets, or computers with modern web browsers (Chrome, Firefox, Safari, Edge)

**Software Requirements:**

* Web application built using Java (Spring Boot) or a similar backend framework
* Front-end developed with HTML, CSS, and JavaScript (React or Angular for responsiveness)
* Relational database (e.g., MySQL or PostgreSQL) to manage users, appointments, packages, and test results
* Email/SMS notification service (e.g., Twilio or SendGrid) for appointment confirmations and updates

**Tools:**

* Lucidchart for creating and maintaining UML diagrams
* Git for version control and collaboration
* Integrated Development Environment (IDE) like IntelliJ IDEA or Eclipse for Java development
* Browser-based testing tools for cross-platform interface testing

**Infrastructure Requirements:**

* Deployed in a secure cloud environment (e.g., AWS, Azure, or Google Cloud Platform)
* HTTPS with SSL encryption for all client-server communications
* Daily automated backups of the database and configuration
* Role-based access control for managing permissions (student, secretary, IT, DMV, admin)
* Authentication system with secure password hashing and automated reset workflows
* Brute-force protection with lockout mechanism after multiple failed login attempts