# 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 diagram

Description automatically generated

### UML Activity Diagrams

*A diagram of a diagram

Description automatically generated*

*A diagram of a software application

Description automatically generated*

### UML Sequence Diagram

*A diagram of a website

Description automatically generated*

### UML Class Diagram

*A computer screen shot of a diagram

Description automatically generated*

## Technical Requirements

Technical Requirements for DriverPass:

Technology Stack:

Backend: Node.js, Express.js for server-side development.

Frontend: HTML5, CSS3, JavaScript, and a modern frontend framework (e.g., React or Angular).

Database: NoSQL database (e.g., MongoDB) to store user information, schedules, exams, and results.

Hosting: Deploy the application on a cloud platform like AWS, Azure, or Google Cloud.

Tools Requirements:

Use feature-rich IDEs like Visual Studio Code and IntelliJ IDEA for efficient coding, debugging, and testing.

Version management: Code management, change tracking, and collaborative development with Git.

Automate unit, integration, and UI testing with Selenium or PHPUnit.

Infrastructure Requirements:

To ensure scalability, reliability, and easy management, consider cloud platforms such as AWS, Azure, and Google Cloud.

Website performance and static asset delivery (images, stylesheets) are improved with CDN integration.

Payment Gateway Security: Use PayPal or Stripe for online payment.

Security requirements:

HTTPS/TLS: HTTPS and SSL/TLS certificates secure user data during transmission.

Use multi-factor authentication to secure user accounts.

Secure passwords and payment details with AES.

Performance:

Optimize the system for speed to prevent user frustration during exams.

Use asynchronous programming techniques to ensure smooth user interactions.

Implement caching mechanisms where appropriate to improve performance.

Scalability:

Design the system with scalability in mind to accommodate increasing user traffic.

Utilize cloud-based infrastructure that allows easy scaling based on demand.

User Interface (UI):

Design an intuitive and user-friendly interface with clear navigation and consistent styling.

Implement responsive design to ensure the UI adapts to various screen sizes.