# 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 person's work flow

Description automatically generated*

### UML Activity Diagrams

*[You were asked to choose* ***two*** *use cases and create* ***two*** *activity diagrams, one for each use case. Please insert* ***both*** *of your activity diagrams here. Check to make sure that you included appropriate components and symbols and that your design meets the client’s needs.]*

### UML Sequence Diagram

*[You were asked to create a sequence diagram based on* ***one*** *of the use cases you chose. Please insert your sequence diagram here. Check to make sure that you included appropriate components and symbols and that your design meets the client’s needs.]*

### UML Class Diagram

*[You were asked to create a class diagram based on the different classes and attributes needed for your system design. You are* ***not*** *required to include methods, but you may if you wish. Please insert your class diagram here. Check to make sure that you included appropriate components and symbols and that your design meets the client’s requirements.]*

## Technical Requirements

*[Based on the diagrams you have created, describe the technical requirements of your system. These requirements should address the required hardware, software, tools, and infrastructure necessary for your system design.]*

Hello Class,

I think using the same programmers for both the coding and testing of software is the most effecient and effective way of building out a system. When you have programmers completing multiple tasks withing the development cycle, there is less room for errors and breakdowns in communication.

As an example I can think of for encountering errors in a system, I will use the platform I am using to learn front end development: Scrimba. Recently, Scrimba released v2.0 of their platform, and with it, there were lots of bugs and errors in the system. Luckily they released it with a nice big button to report errors and leave comments about them. They also have a very active discord community where you can submit error reports as well. The best part is that people from the company will personaly contact you to ask questions about the issues and help resolve them.

In the implementation of the DriverPass system, I would ad something similar like a very obvious button for reporting any bugs or broken parts of the system and submit tickets so that users dont feel like they have to jump through hoops to provide feedback.

Being a good team member means good communication skills in both directions. It means making choices that benifit the team as a whole wrather than the individual. Good documentation creates an enviroment where other developers coming into a common code base can understand what they are looking at without having to go out of there way, or get confused.

Log into course

Open practice test

Take test view results

Give feedback

Review results/review feedback