# 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.

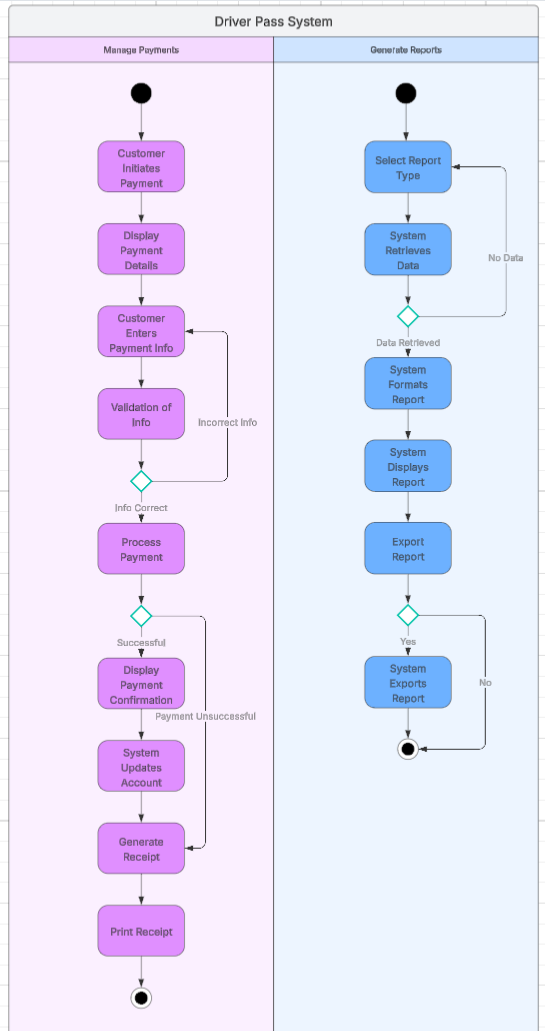
## UML Diagrams

### UML Use Case Diagram

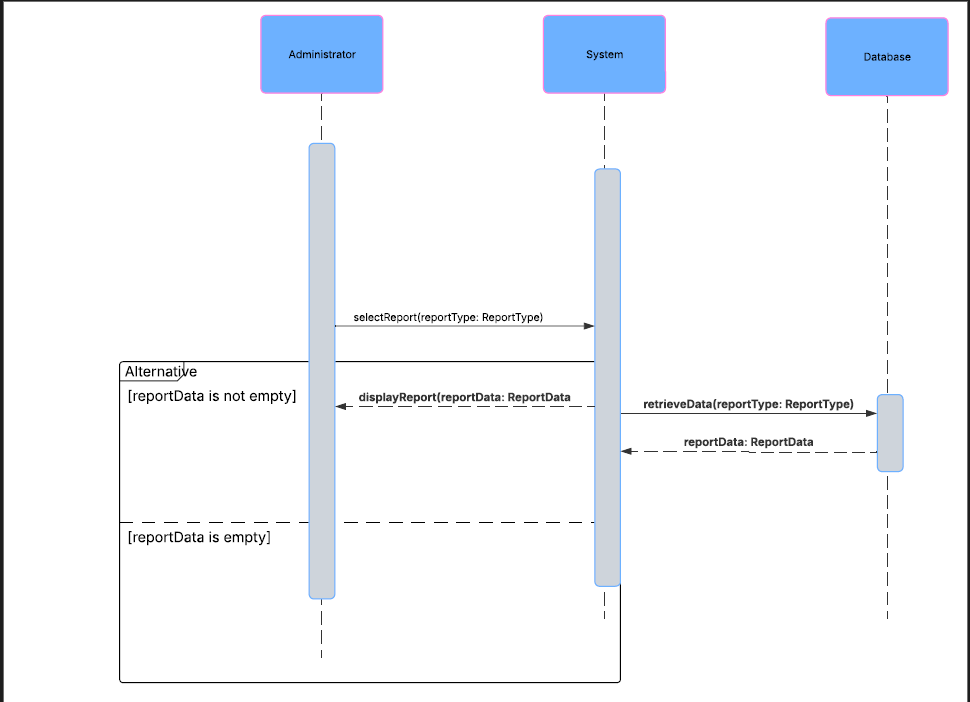
A diagram of a driver pass use case diagram

AI-generated content may be incorrect.

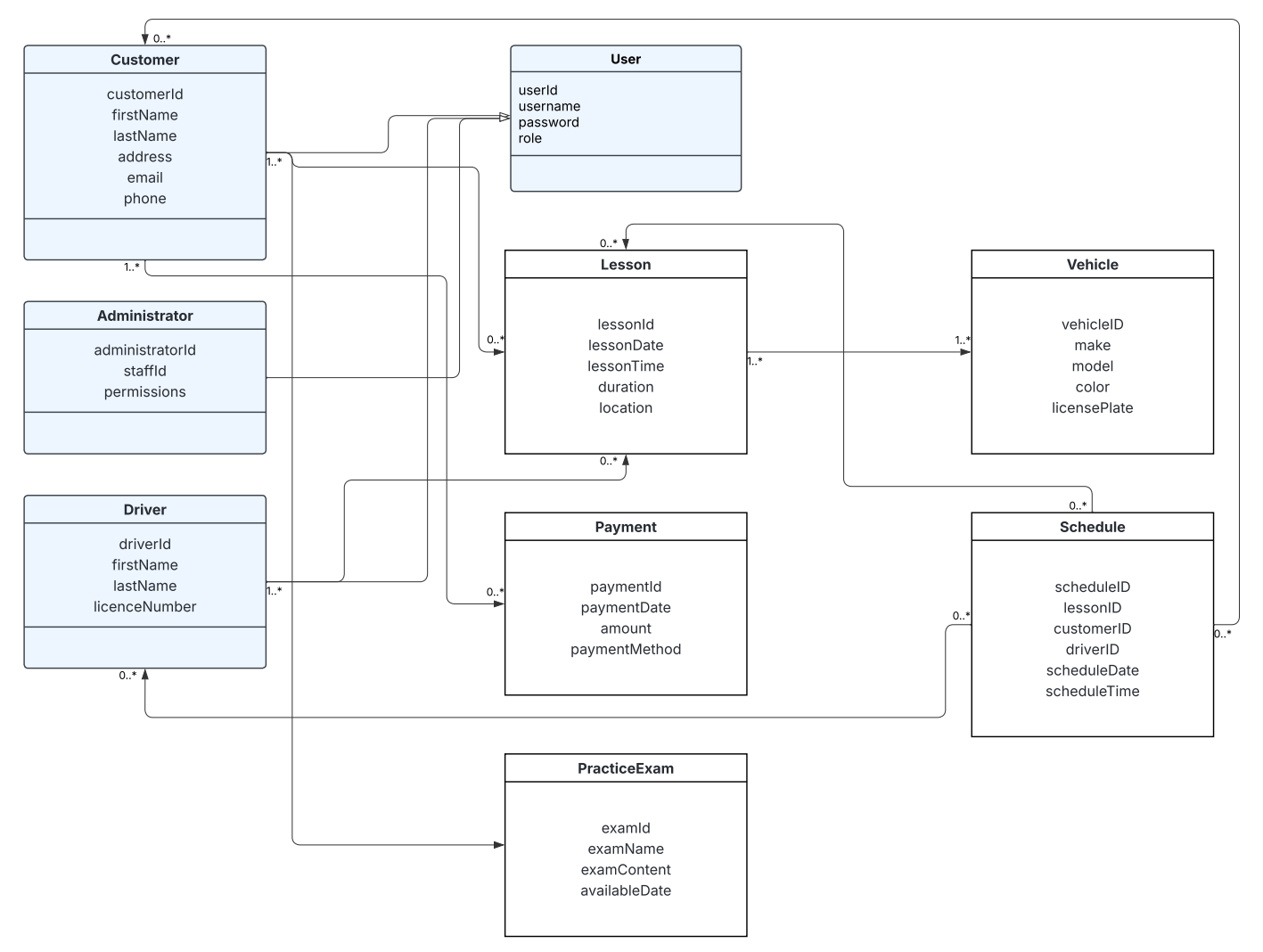
### UML Activity Diagrams



### UML Sequence Diagram

**

### UML Class Diagram



## Technical Requirements

Technical requirements for the DriverPass system:

**1. Hardware Requirements**

* **Servers:**
  + Sufficient processing power (e.g., multi-core CPUs) and memory (RAM) to handle concurrent user requests and data processing.
  + Storage capacity (e.g., SSD or HDD) to store the database, application files, and user-generated content.
  + Network interface cards (NICs) for reliable network connectivity.

* **Client Devices:**
  + Standard desktop or laptop computers with compatible web browsers (e.g., Chrome, Firefox, Safari).
  + Mobile devices (smartphones and tablets) with iOS or Android operating systems (if mobile access is required).
  + Devices should have sufficient screen resolution to display the user interface correctly.

**2. Software Requirements**

* **Operating System:**
  + Server operating system (e.g., Linux, Windows Server) to host the application and database.
* **Web Server:**
  + Web server software (e.g., Apache, Nginx) to handle HTTP requests and serve web pages.
* **Database Management System (DBMS):**
  + Robust DBMS (e.g., MySQL, PostgreSQL) to store and manage system data (user accounts, course content, schedules, etc.).
* **Programming Languages and Frameworks:**
  + Programming languages for back-end development (e.g., Python, Java, PHP) and front-end development (HTML, CSS, JavaScript).
  + Web frameworks (e.g., Django, Spring, Laravel, React, Angular) to facilitate development.
* **API (Application Programming Interface):**
  + APIs for communication between different system components and integration with external services (e.g., payment gateways, DMV).

**3. Tools and Technologies**

* **Integrated Development Environment (IDE):**
  + IDEs (e.g., Visual Studio Code, IntelliJ) to write and debug code.
* **Version Control System:**
  + Version control system (e.g., Git) to manage code changes and collaboration.
* **CASE Tool:**
  + UML diagramming tool (e.g., Lucidchart) to create and maintain system design diagrams.

**4. Infrastructure Requirements**

* **Network Infrastructure:**
  + Reliable network connectivity with sufficient bandwidth to support user traffic.
  + Network security measures (e.g., firewalls, intrusion detection systems) to protect the system.
* **Cloud Hosting (Recommended):**
  + Cloud hosting platform (e.g., AWS, Azure, Google Cloud) to provide scalability, reliability, and security.
* **Security Infrastructure:**
  + SSL certificates for HTTPS to encrypt communication.
  + Security protocols and mechanisms to protect data at rest and in transit.

**References**

Grammarly. (n.d.). *Grammarly (Version 1.2.141.1617)* [Software application]. Retrieved from <https://www.grammarly.com/>

Southern New Hampshire University. (2025). CS255 DriverPass Interview Transcript. CS 255: System Analysis and Design 2025 C-2 (Mar-Apr). Module 1.

Dennis, A., Wixom, B. H., & Tegarden, D. (2015). Systems analysis and design with UML (4th ed.). Wiley

Valacich, J. S., & George, J. F. (2024). *Modern Systems Analysis and Design* (10th ed.). Pearson Education (US). <https://mbsdirect.vitalsource.com/books/9780138180294>