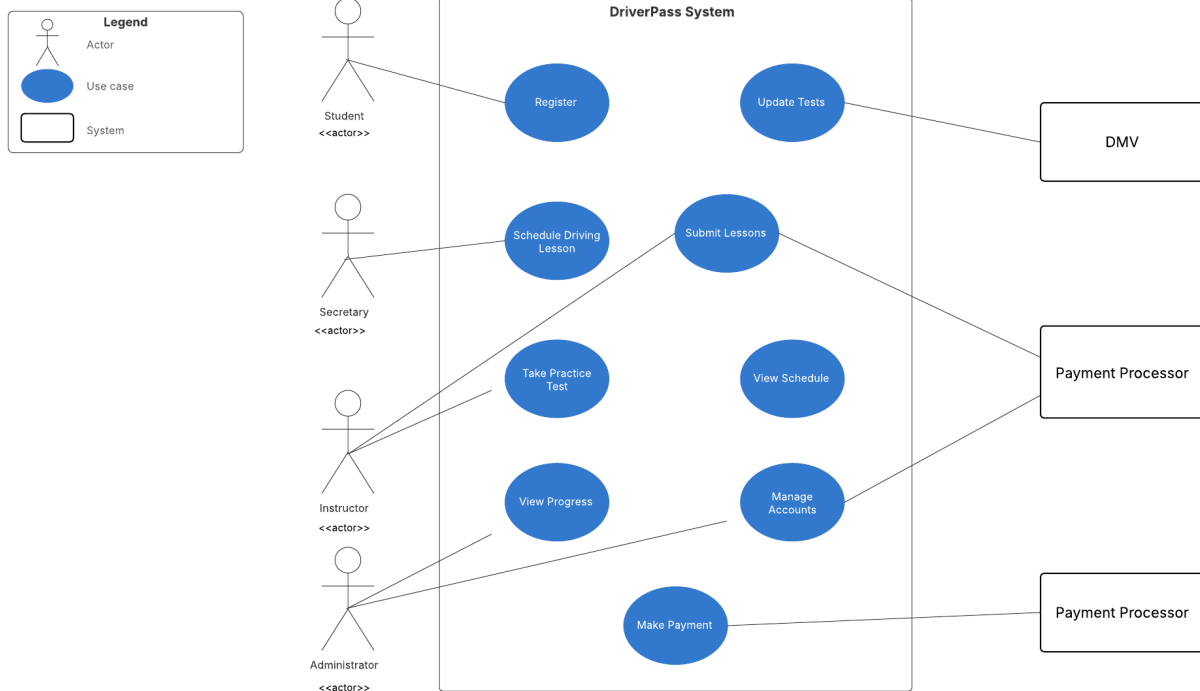


CS 255 System Design Document Template

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.

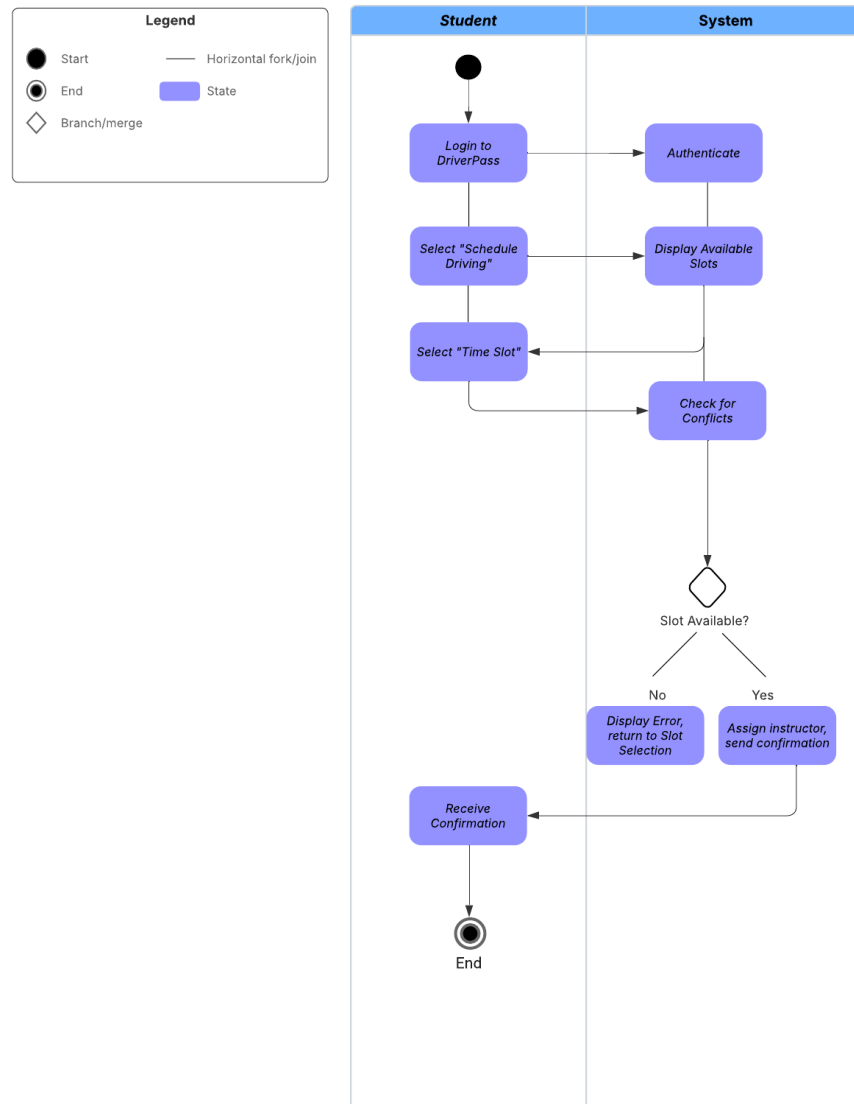
UML Diagrams

UML Use Case Diagram

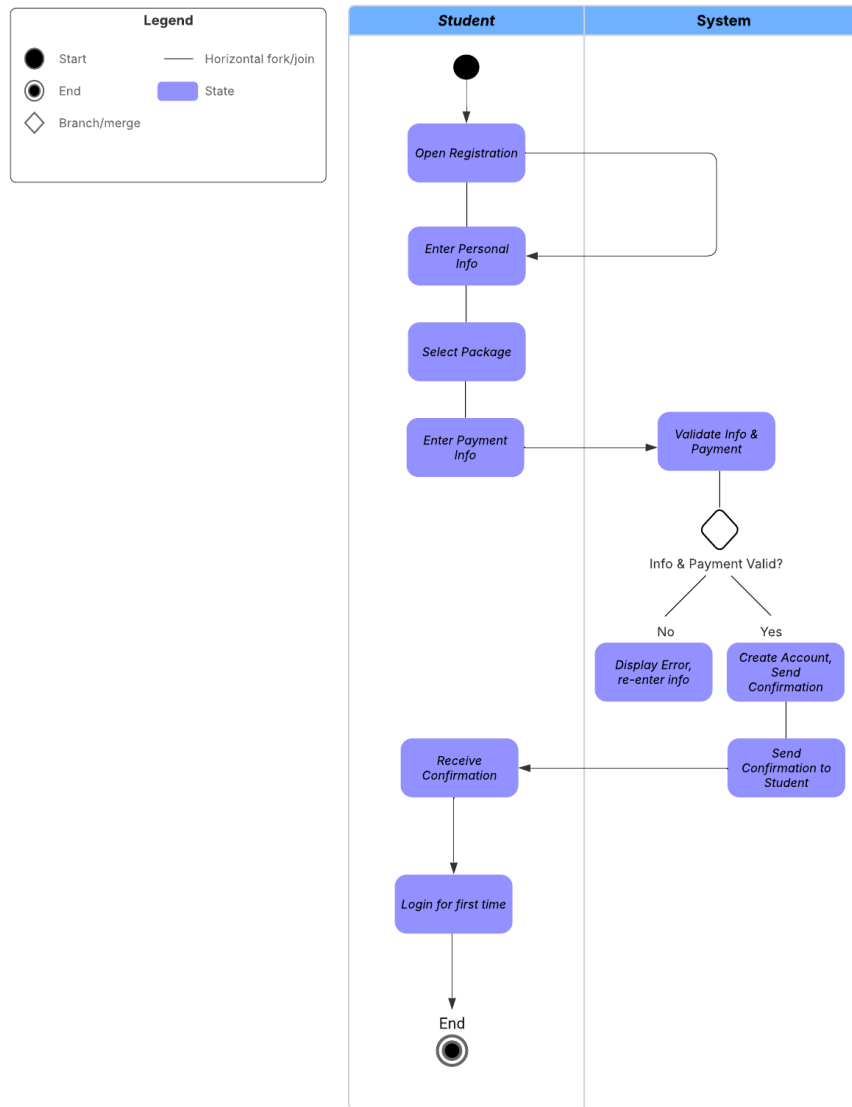


UML Activity Diagrams

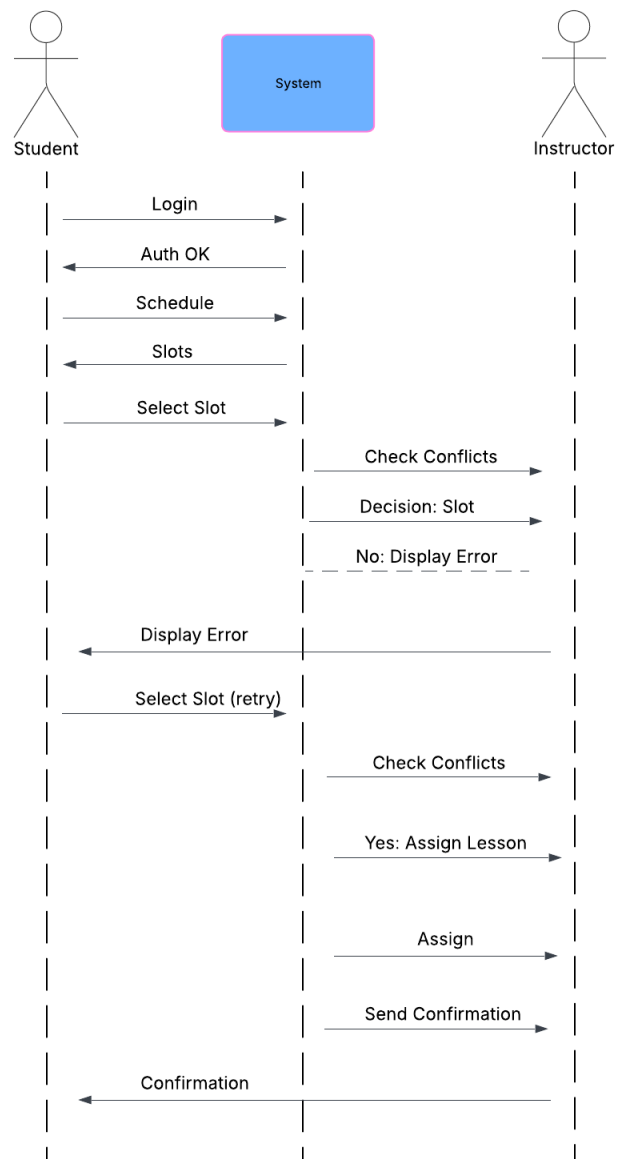
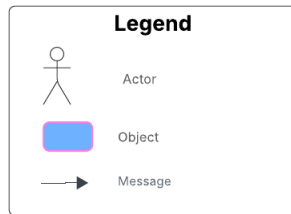
Activity Diagram: Schedule Driving Lesson Use Case



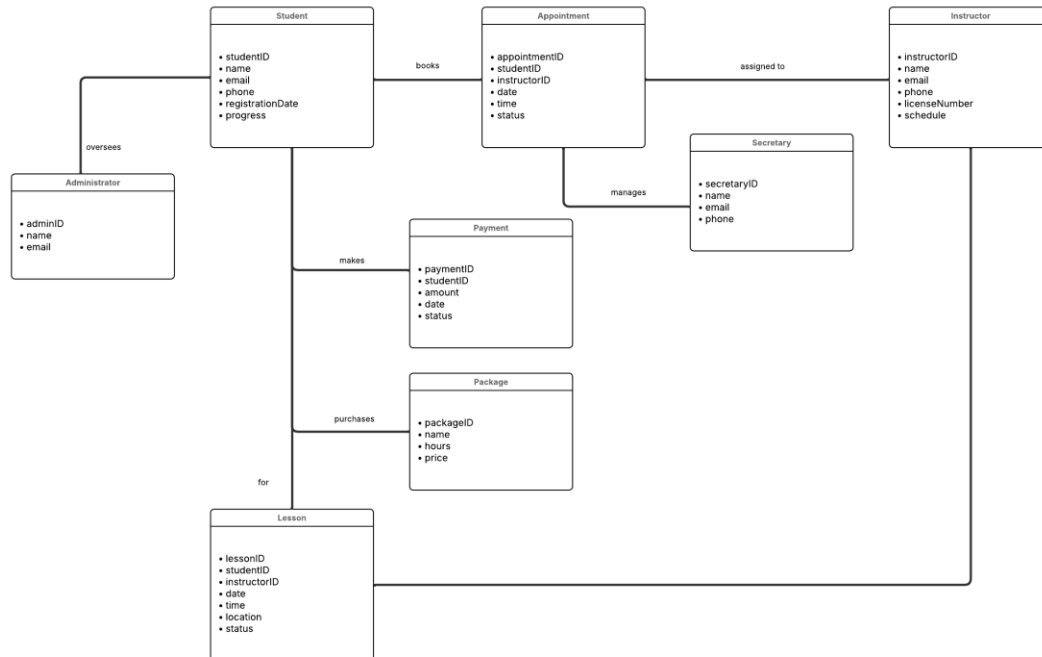
Activity Diagram: Register



UML Sequence Diagram



UML Class Diagram



Technical Requirements

The DriverPass System will be delivered as a modern web-based system for all students, instructors, secretaries, and administrators. We plan to use dependable cloud services like AWS, Azure, or Google Cloud to host the system. This will allow easy user access to the Driver PassSystem via laptop, desktop, tablet, or smartphone, while independently allowing them to work with their own machines. All users need is a standard browser like Chrome, Firefox, Safari, or Edge which eliminates the need for special hardware, or software at their end.

The main data sources will be a reliable relational database like PostgreSQL or MySQL to manage students, instructors, bookings, lessons, payments, and packages. The backend will use a framework that is widely used (Node.js, Python (Flask or Django), Java Spring Boot), so that we will be able to efficiently scale and maintain the platform. All data reported to be sent from the end-users to the platforms will be encrypted, along with ensuring that each user receives only what they require, by their role.

In efforts to keep students and staff informed, automatic emails will be sent as notifications through a service such as SendGrid, AWS SES, or Gmail API for confirmations, reminders, or any other important notifications. We can securely connect the DriverPass System to a reputable

provider, such as Stripe or PayPal, if it processes online payments, allowing protection of financial data and reliable electronic transactions.

The technologies that we decide to use will allow us efficiency in development: Visual Studio Code to code, GitHub for version control, Lucid chart or draw.io to create or update a diagram, and Postman to test our API endpoints. Google Workspace or Microsoft Office will be used for documentation and reporting, depending on what the team prefers and what their workflow supports.

With regards to infrastructure, the focus will be on security, speed, and compliance. There will be routine automated backups of the production database, and all sensitive information and payments data will utilize encryption at rest. There will be a secure authentication system used for users that I would recommend be OAuth at a minimum, and preferably two-factor authentication when possible. The platform is designed for growth with cloud capabilities like load balancing and failover (availability) support planned. We will be aware of the privacy or data protection regulations applicable to our business or any information that we collect, especially student records or payment related information, whether that regulation is from a school body, government, or other governing body, depending on where or how the DriverPass System is used.

To future proof the DriverPass System, it will provide RESTful API endpoints. This means we are prepared for the day that we do want to connect to a mobile app, allow outside integrations, or add reporting options at a later date.

In conclusion, the top considerations for the technical requirements are the foundation for a secure, reliable, seamless system for users whether they are securing lessons, managing schedules, payments, or operating.