**CS-255 System Design Document – DriverPass**

*October 19, 2025*

Alex Akers

Alex.akers@snhu.edu

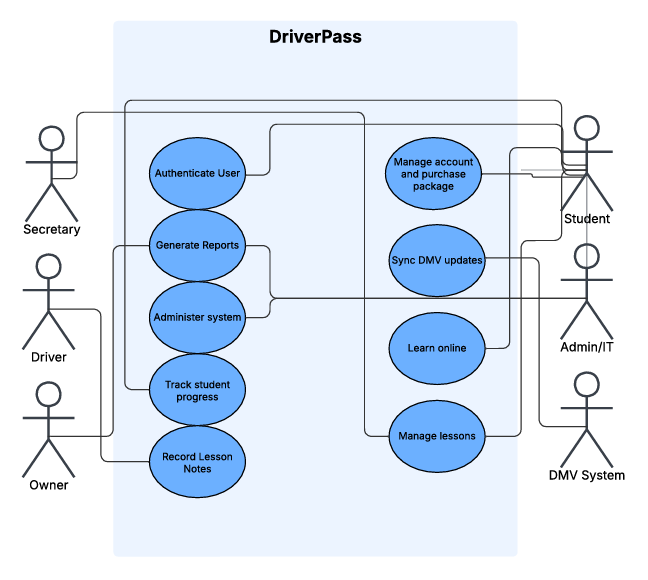
CS-255

# 1. Introduction

This document shows how the DriverPass system will work. It uses simple visuals and short paragraphs to explain the design. The goal is to show how a student, secretary, driver, owner, and admin use the system to schedule lessons, take practice tests, manage accounts, and how the data and services support those actions.

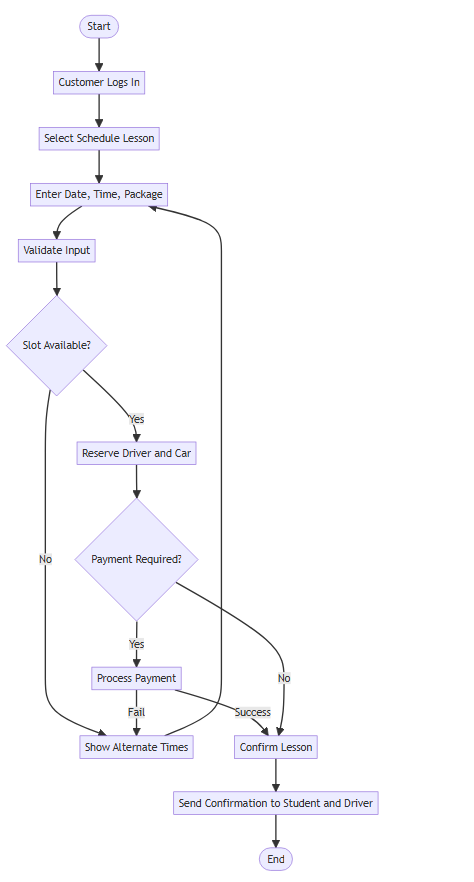
# Use Case Diagram

*Covers FR: scheduling, online tests, account mgmt, reporting, DMV sync.*

This diagram shows the main user roles (Student, Secretary, Driver, Owner, Admin/IT) and the DMV system, and how they interact with core functions like scheduling lessons, taking practice tests, managing accounts, and syncing DMV updates.

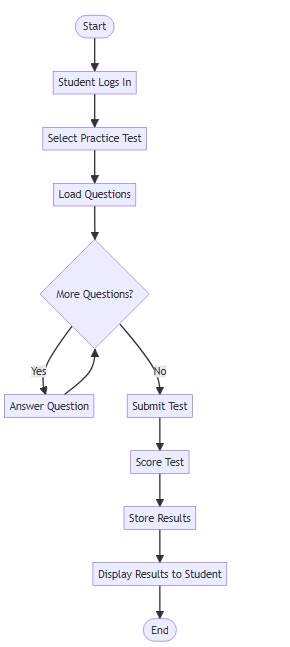
# Activity Diagrams

## 3.1 Schedule Lesson (Activity Diagram) *Implements scheduling + availability + notifications.*



From login to confirmation, the flow checks availability, reserves a driver and car, handles payment, creates the reservation, and sends notifications.

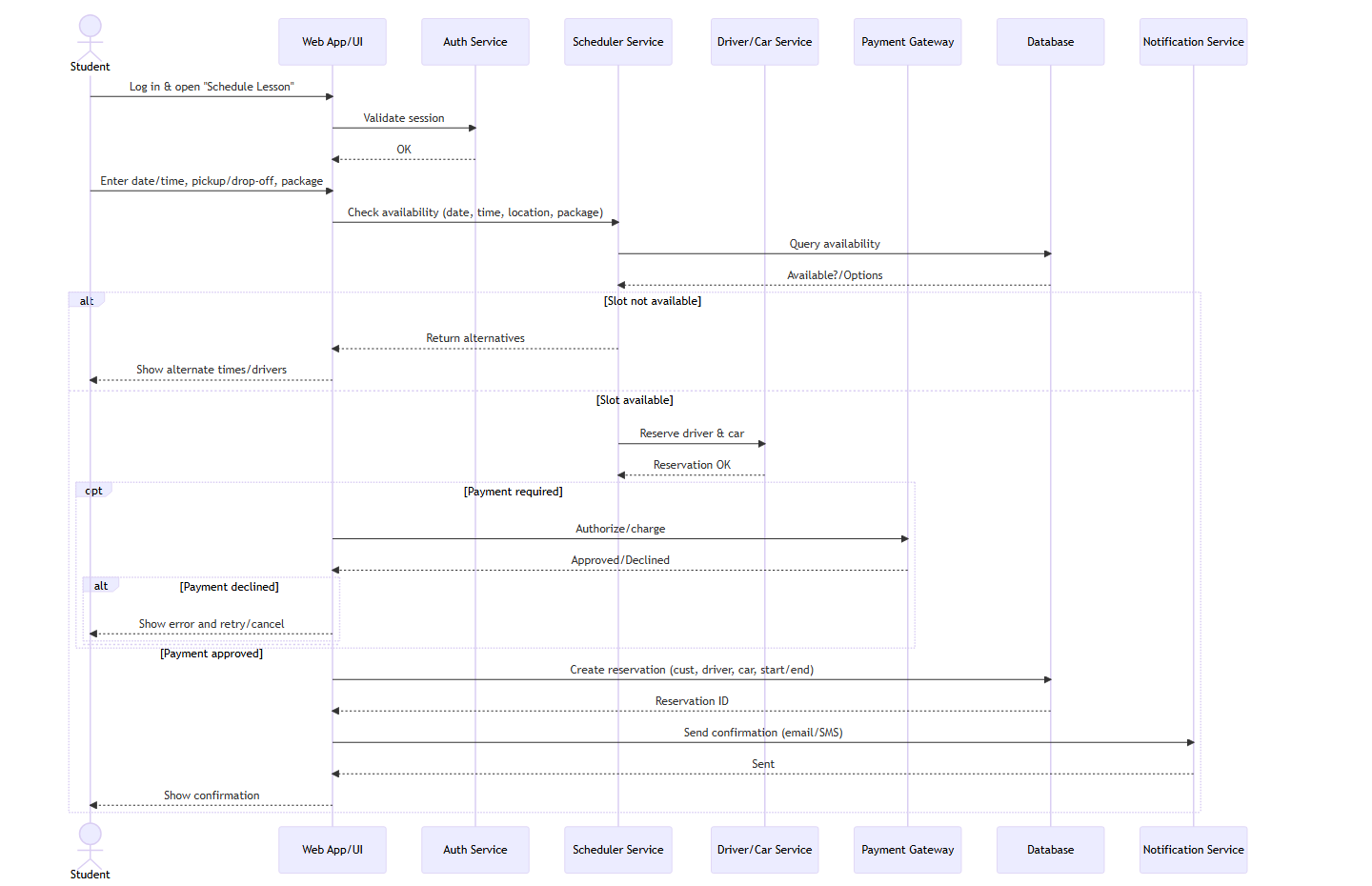
## 3.2 Take Practice Test (Activity Diagram)



*Implements test access, scoring, status updates.*

The student selects a test, answers questions in a loop, submits, and receives a score. The system stores the attempt and updates progress with pass/fail status.

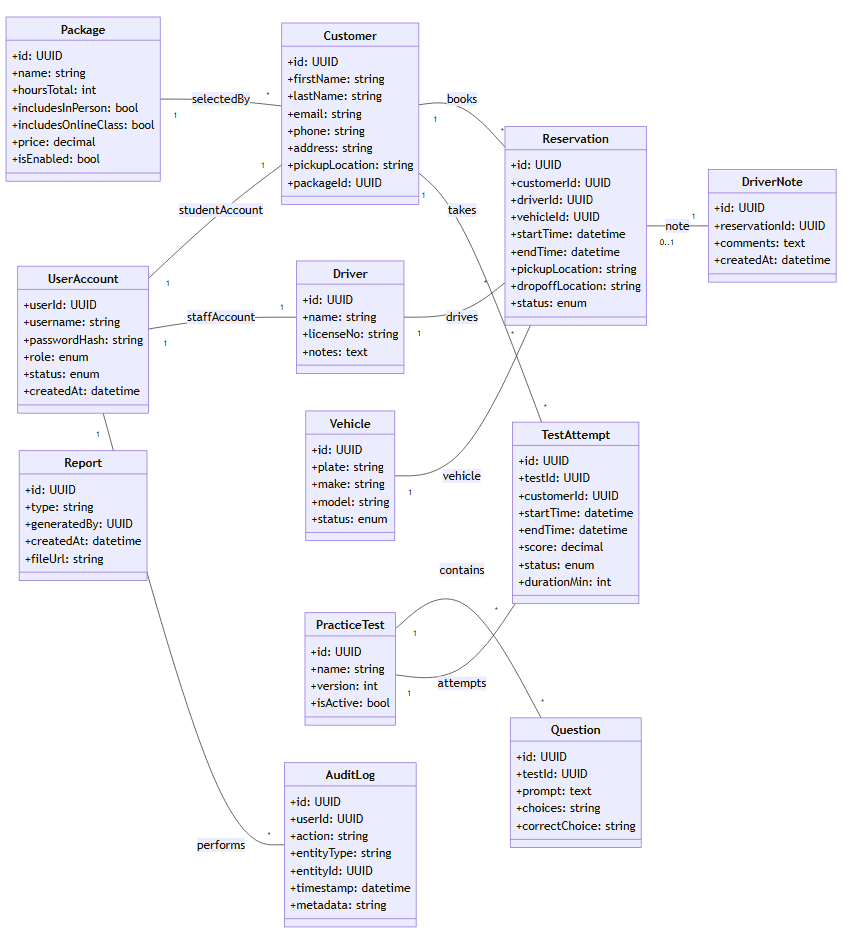
# 4. Sequence Diagram



*Shows message flow for scheduling (auth→availability→payment→create).*

Schedule Lesson. Lifelines include Student, Web App/UI, Auth Service, Scheduler Service, Driver/Car Service, Payment Gateway, and Database. Messages cover checking availability, assigning driver/vehicle, optional payment, creating the reservation, and returning the confirmation.

# 5. Class Diagram



*Defines data needed to support reservations, tests, notes, packages.*

# 6. Technical Requirements

* Web application hosted in the cloud (available 24/7).
* Modern browsers: Chrome, Edge, Firefox, Safari; responsive UI for desktop and mobile.
* Relational database (MySQL/PostgreSQL) for users, lessons, tests, and logs.
* Secure HTTPS, password reset, account lockout after repeated failures.
* Admin tools for enabling/disabling packages and managing users.
* DMV updates integrated by syncing content and questions when available.

# 8. Appendix: Diagram Exports

