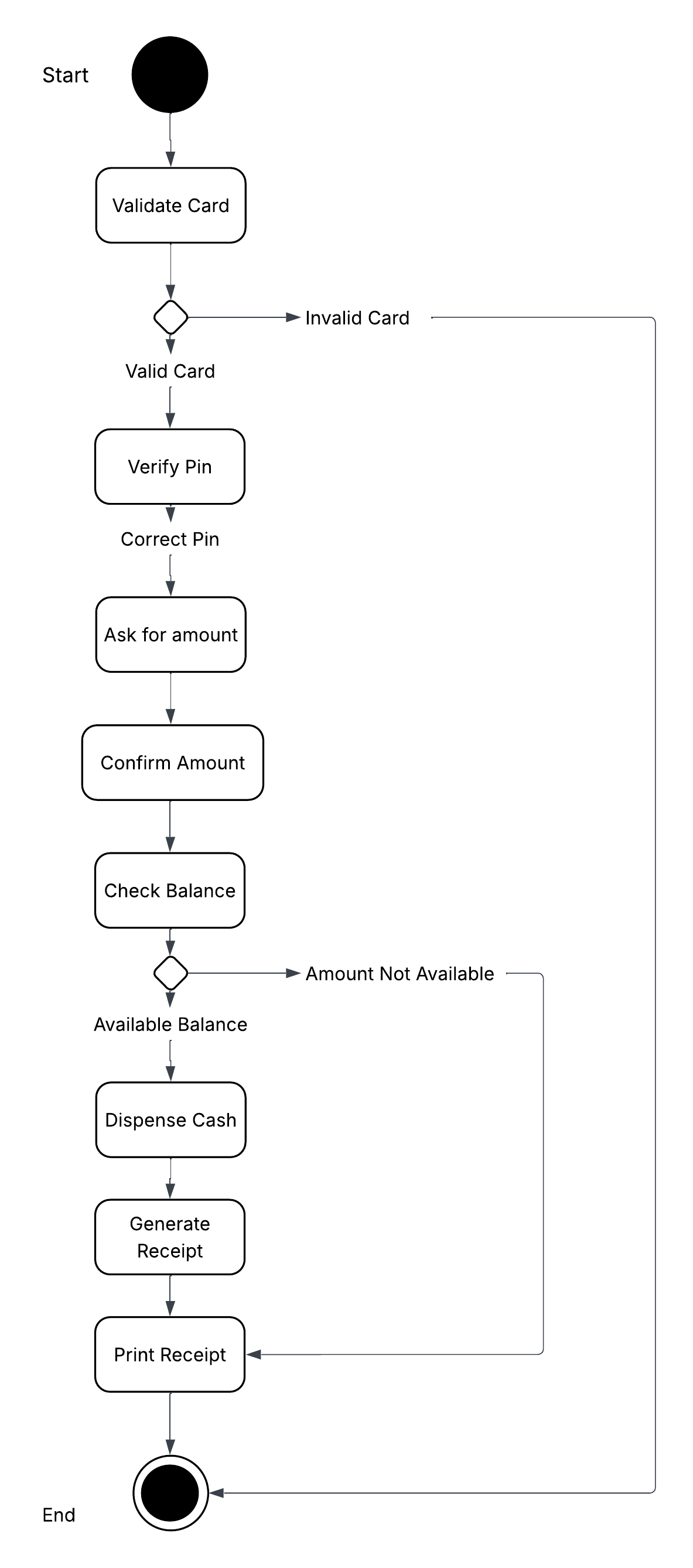
# CS 255 System Design Document

Amelia Sivick

## UML Diagrams

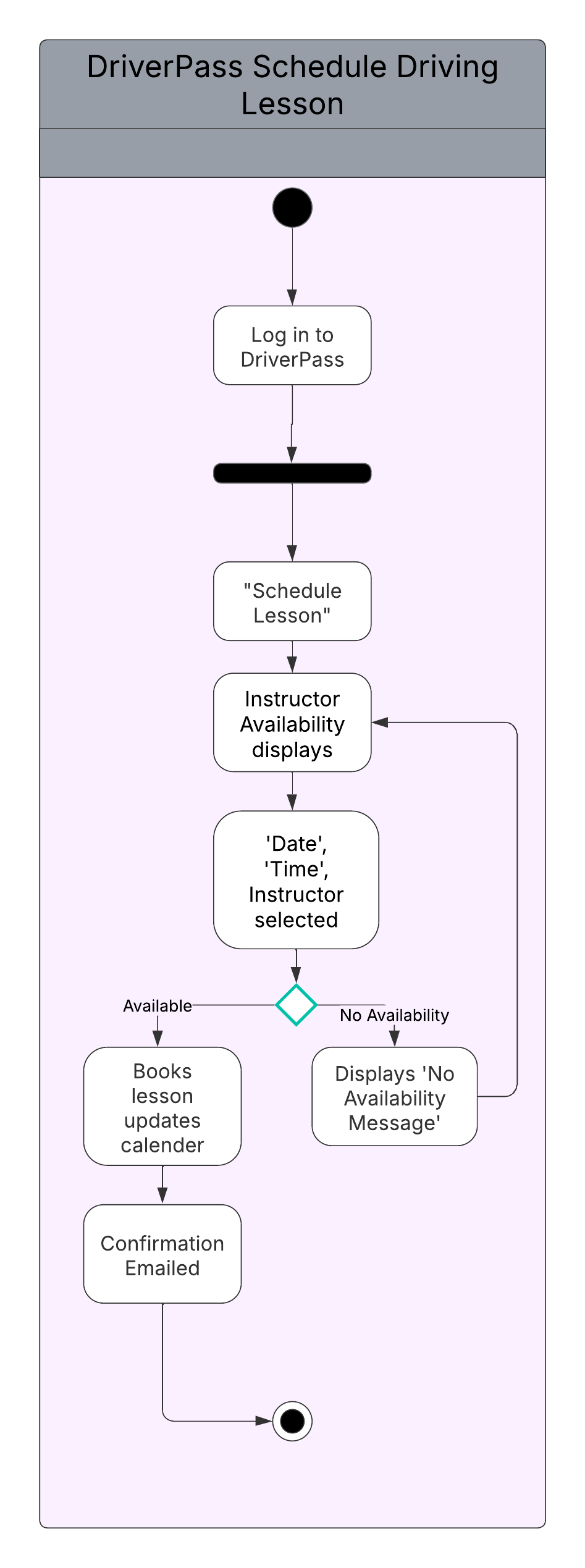
### UML Use Case Diagram

[](https://lucid.app/lucidchart/18a478d4-7a9d-4a6b-b25b-139c69787c82/edit?crop=content&page=0&signature=d5bb3a1389c49c54de2049b07ff33cdf071b7f46fb92668d2cd9edbb1c934258)

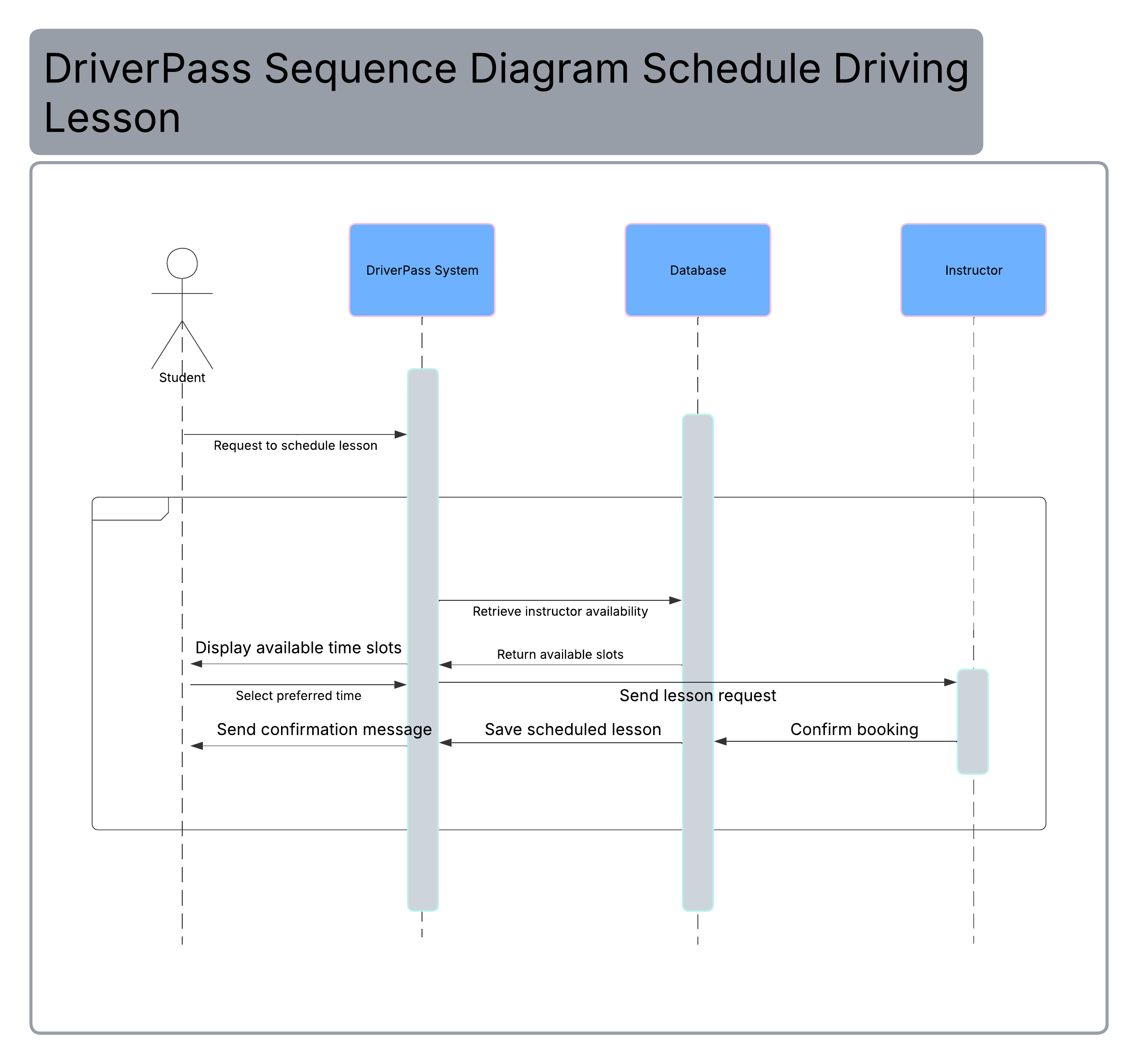
### UML Activity Diagrams

### *[CS255_Module7ActivityDiagram1_AmeliaSivick](https://lucid.app/lucidchart/828ac7bf-30f7-436f-a483-c4f370f53651/edit?crop=content&page=0&signature=90f0a5f172d14be0e0bf42dd71cfbe0078ca40b82824a2ee02c1eb83d740e8c1)DriverPass Register New Account*

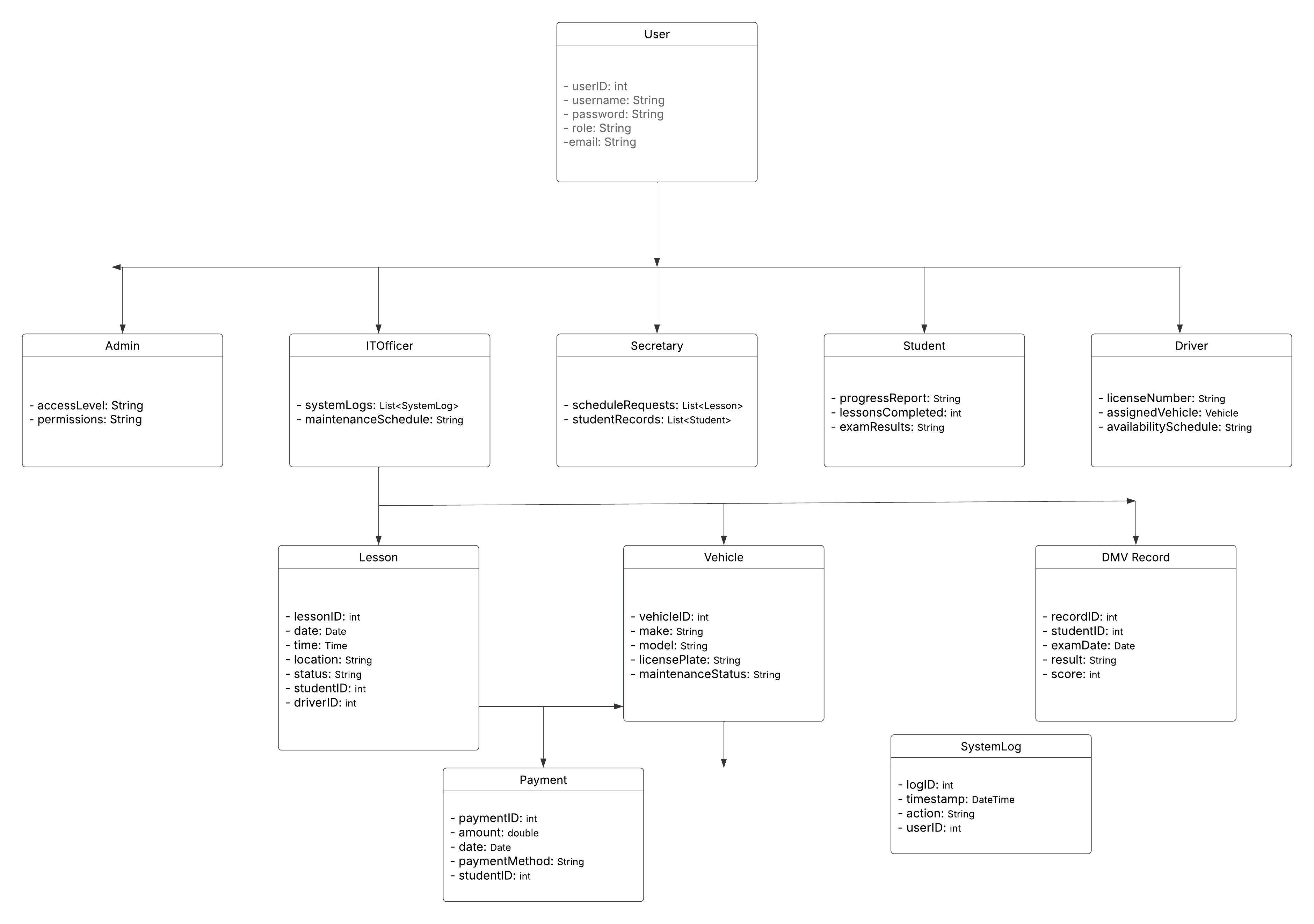
*DriverPass Schedule a Driving Lesson*

*[](https://lucid.app/lucidchart/828ac7bf-30f7-436f-a483-c4f370f53651/edit?crop=content&page=0&signature=90f0a5f172d14be0e0bf42dd71cfbe0078ca40b82824a2ee02c1eb83d740e8c1)*

### UML Sequence Diagram

[](https://lucid.app/lucidchart/828ac7bf-30f7-436f-a483-c4f370f53651/edit?crop=content&page=0&signature=90f0a5f172d14be0e0bf42dd71cfbe0078ca40b82824a2ee02c1eb83d740e8c1)

### UML Class Diagram

[](https://lucid.app/lucidchart/828ac7bf-30f7-436f-a483-c4f370f53651/edit?crop=content&page=0&signature=90f0a5f172d14be0e0bf42dd71cfbe0078ca40b82824a2ee02c1eb83d740e8c1)

## Technical Requirements

*To support the design and operation of the DriverPass system, the system needs to be secure, scalable, and accessible across devices, which means the hardware, software, and infrastructure must work together.*

*On the hardware side, the application should run on a reliable web server with enough memory and processing power to handle user traffic and data logs like 8 GB of RAM. Users will access the platform from desktops, laptops, tablets, and smartphones, so compatibility with modern browsers is a requirement. The network should support fast, secure data transmission, and there should be backup systems in place to protect against data loss.*

*For software, the backend could be built using a Java-based framework like Spring Boot, paired with a relational database such as MySQL or PostgreSQL to manage user accounts, lesson schedules, and DMV updates. The front end should use responsive web technologies (HTML5, CSS, and JavaScript) to ensure a smooth experience on any device. The system should use encrypted connections, hashed passwords, and role-based access controls. Tools like Lucidchart will help with modeling system behavior, while GitHub or GitLab can manage source code and collaboration. Development would likely happen in an IDE like IntelliJ or Eclipse, with project tracking handled through platforms like Jira or Trello. Testing tools such as JUnit and Postman would help validate both backend logic and API functionality.*

*Cloud hosting is the best fit for DriverPass. Platforms like AWS or Azure offer scalability, high availability, and built-in redundancy. A cloud-hosted database ensures reliable storage and protection against data loss. Security infrastructure should include HTTPS for all communication, regular audits, and update cycles to keep the system compliant and secure.*