# CS 255 System Design Document Template

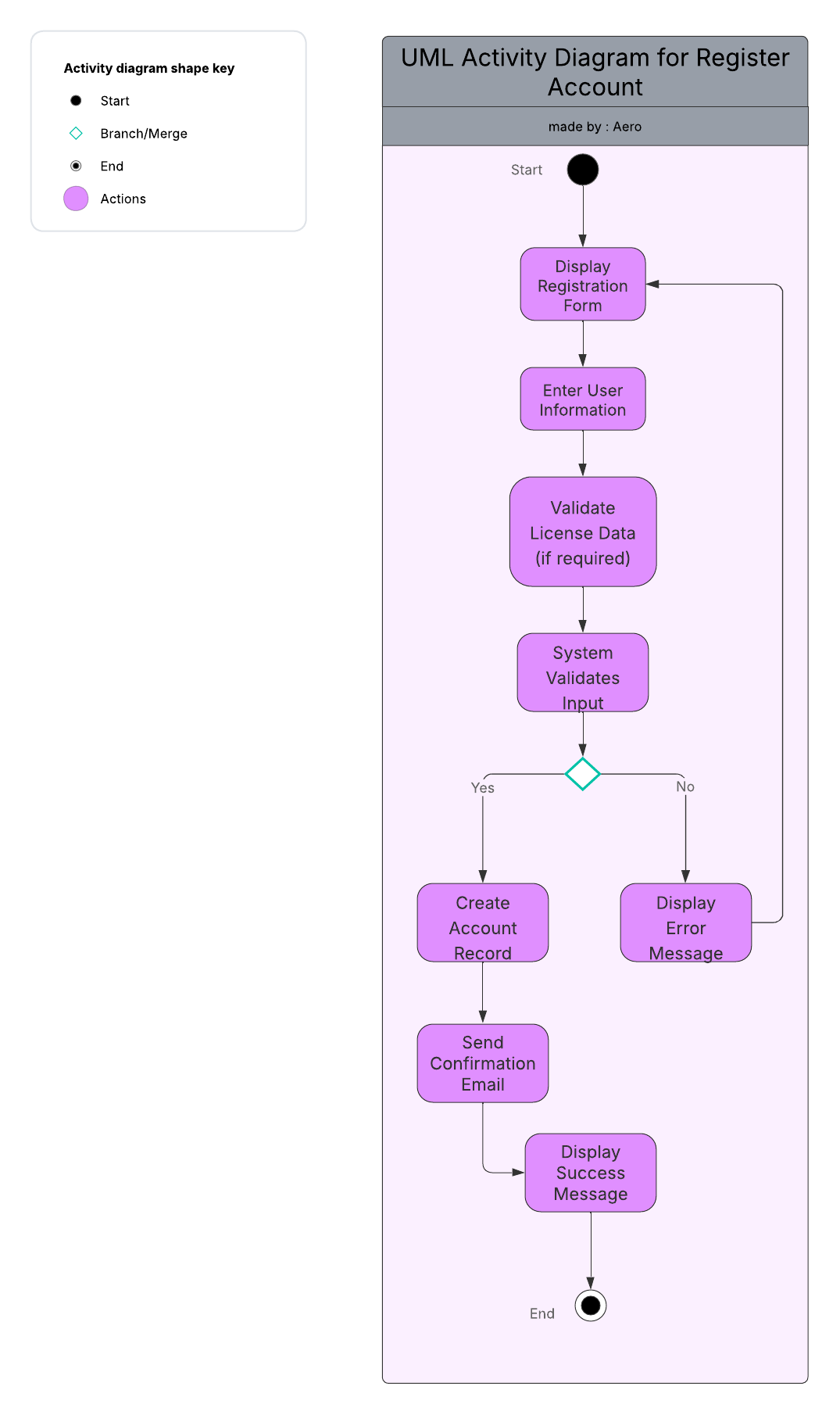
## UML Diagrams

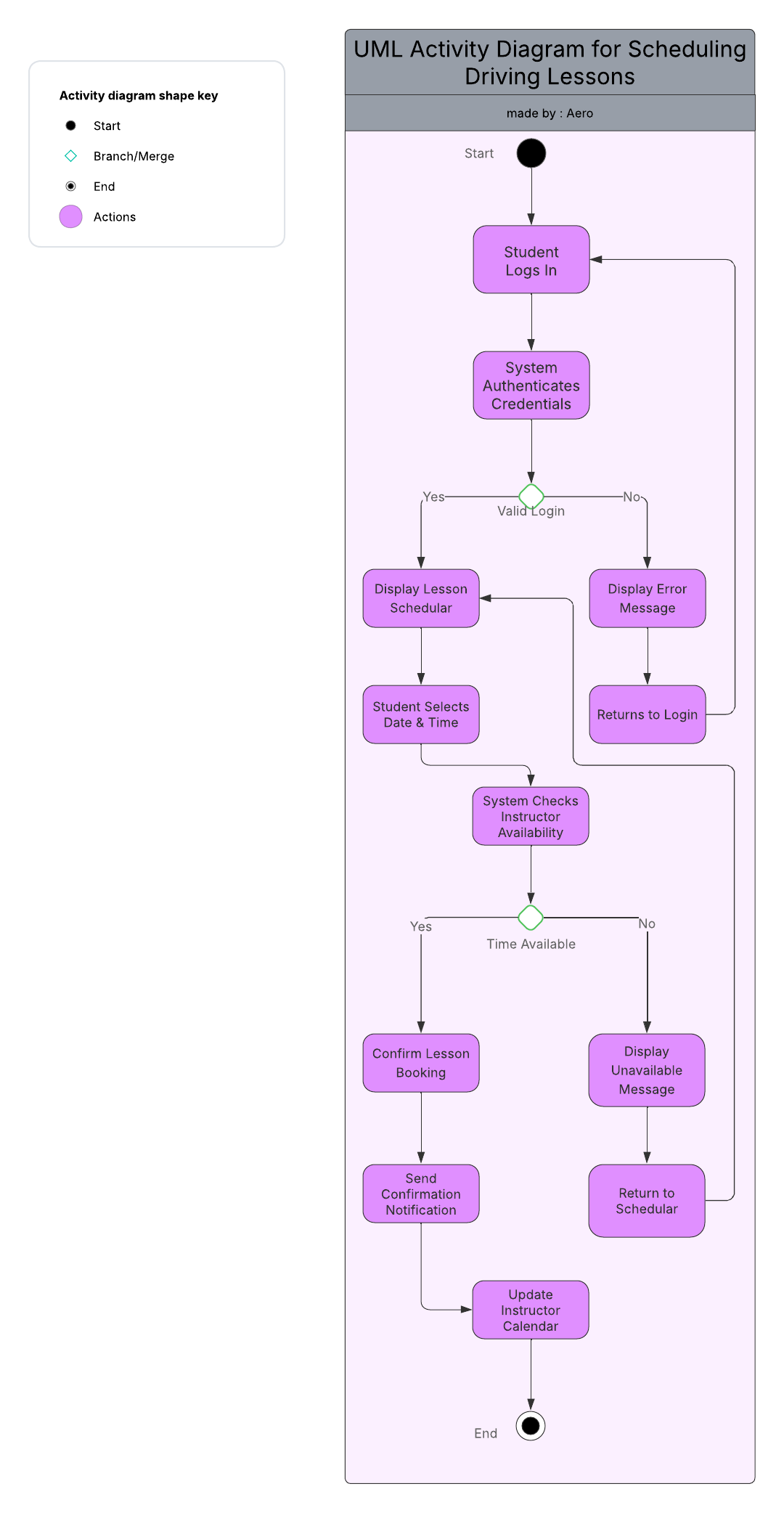
### UML Use Case Diagram

A diagram of a driver pass system

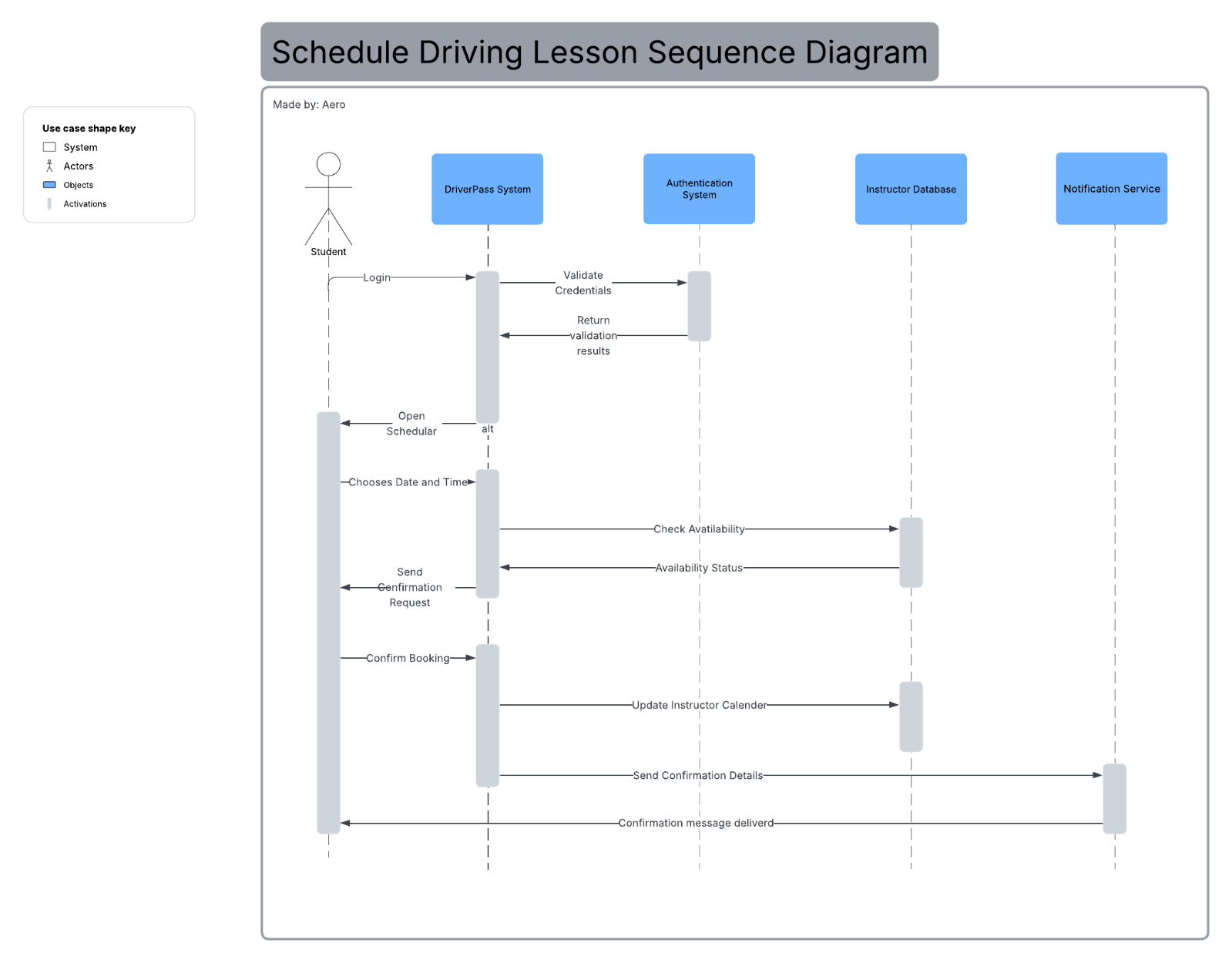
AI-generated content may be incorrect.

### UML Activity Diagrams

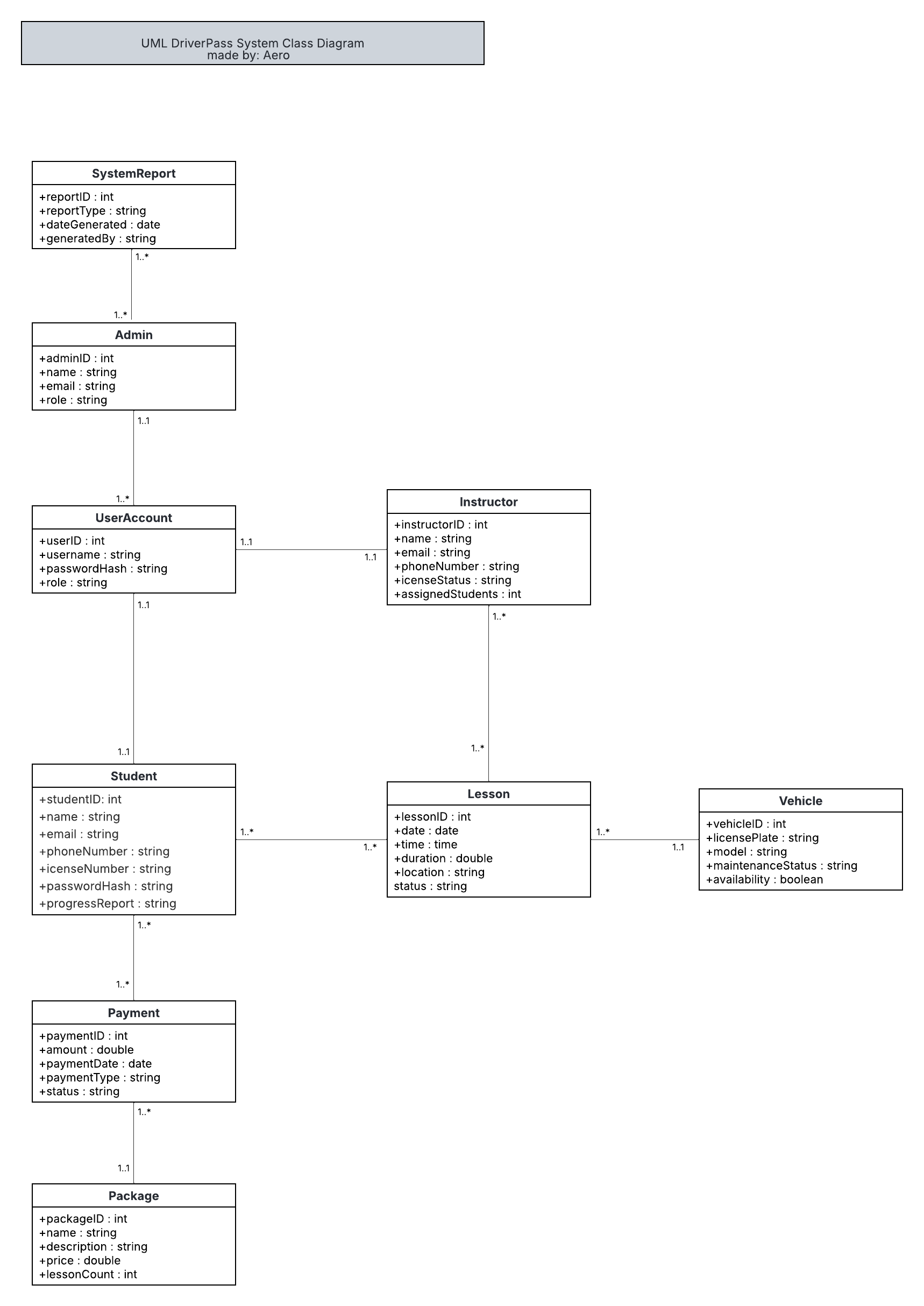




### UML Sequence Diagram

**

### UML Class Diagram



## Technical Requirements

The DriverPass system will require both software and hardware infrastructure to support web-based access, secure storage, and data management. The design follows a client-server model with cloud integration for scalability and security.

Hardware Requirements:

The DriverPass system will operate in a cloud-hosted environment and require standard hardware capable of supporting web-based applications. The client and administrative interfaces will be accessed through desktop or mobile devices with reliable internet connectivity. End users, such as students and instructors, will need devices capable of running a modern web browser to access the online training portal and scheduling system.

For system administration, the hosting server must support secure data storage, and real-time communication between users and the database. The server environment will be managed through a scalable infrastructure that can adjust to increasing user demand without requiring physical hardware changes.

Software Requirements:

* **Operating System Support:** The system should be compatible with common operating systems for users and administrators.
* **Database Management:** Software to store and manage user data, vehicle information, and transaction records securely.
* **Web/Server Framework:** To run the backend logic, handle requests, and serve content to users.
* **User Interface Tools:** Software or frameworks to build the front-end interface for web or mobile applications.
* **Security Software:** Tools for authentication, encryption, and secure communication to protect sensitive data.
* **Reporting/Analytics:** Software for generating usage reports, monitoring system performance, and supporting administrative decisions.

Tools and Infrastructure:

* Development Tools: Environments for coding, testing, and debugging the application.
* Version Control: Tools for managing code changes and collaboration among developers.
* Server Infrastructure: Cloud or on-premises servers to host the application and database.
* Networking: Reliable internet connectivity and network management tools to ensure system accessibility.
* Backup and Recovery**:** Systems to back up data and recover it in case of failure.
* Monitoring Tools**:** To track system performance, uptime, and security incidents.