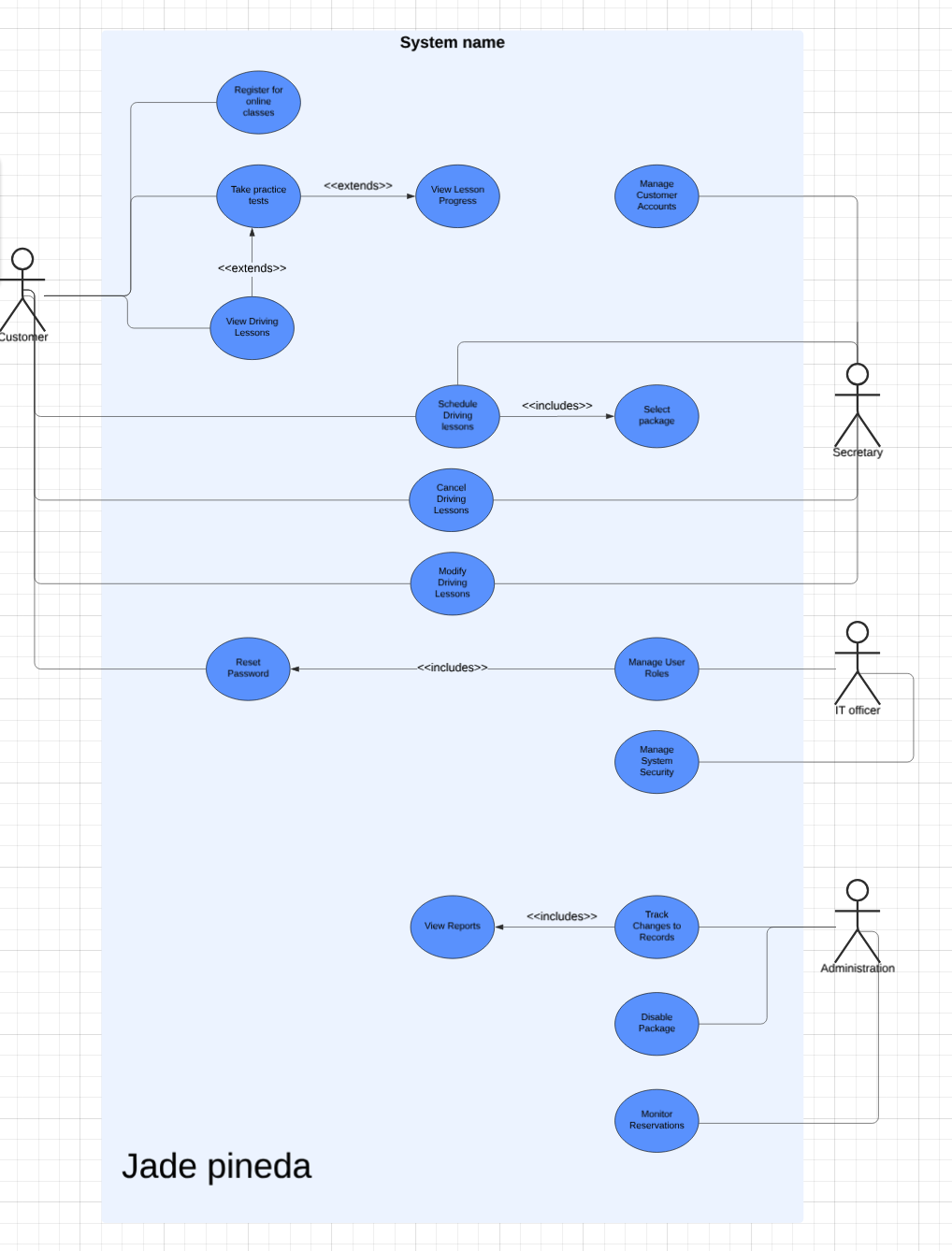
# CS 255 System Design Document Template

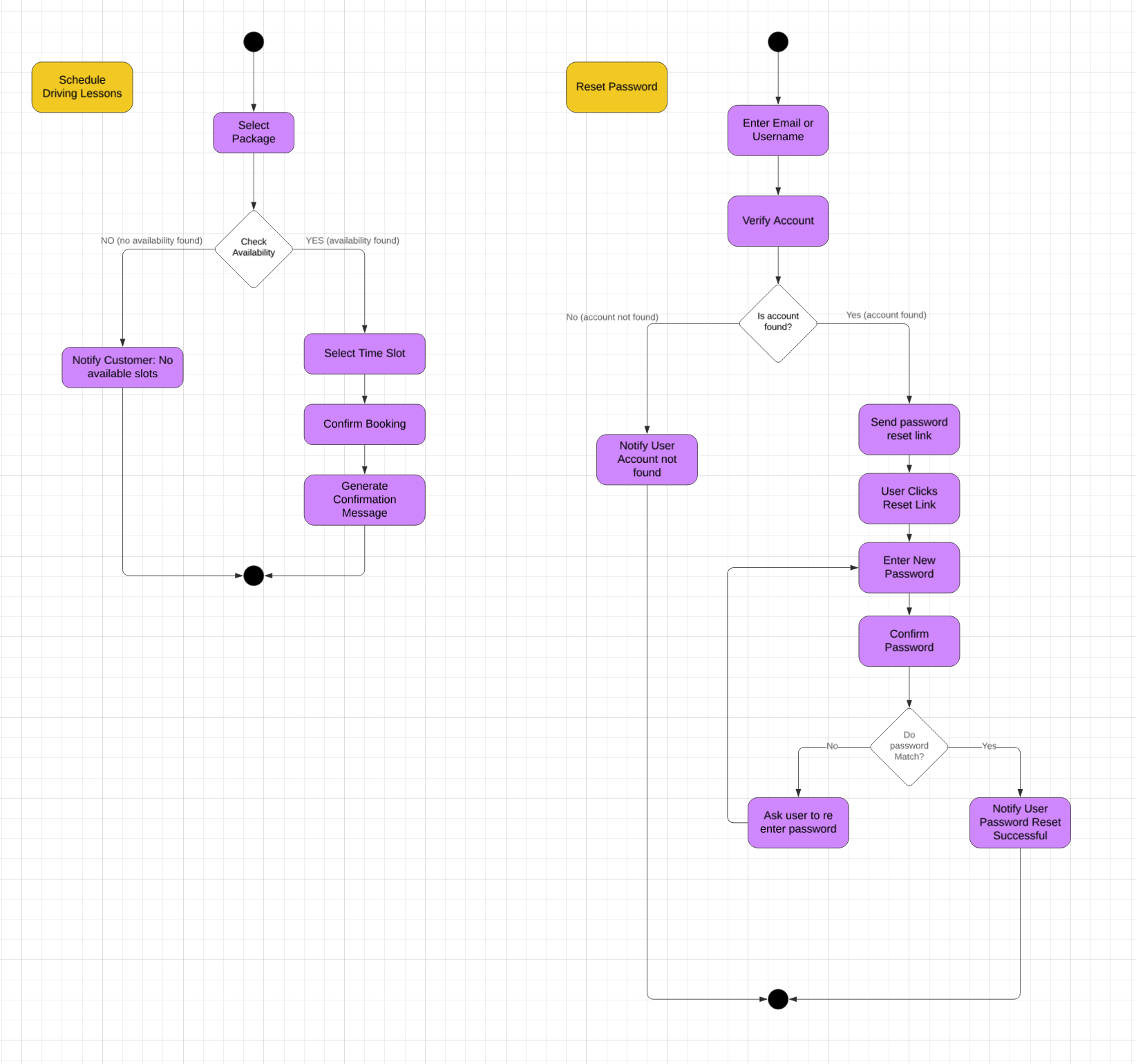
This template lays out all the different sections that you need to complete for Project Two. Each section has guidance to prompt your thinking. You will need to continually reference the interview transcript as you work to make sure that you are addressing your client’s needs. There is no required length 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

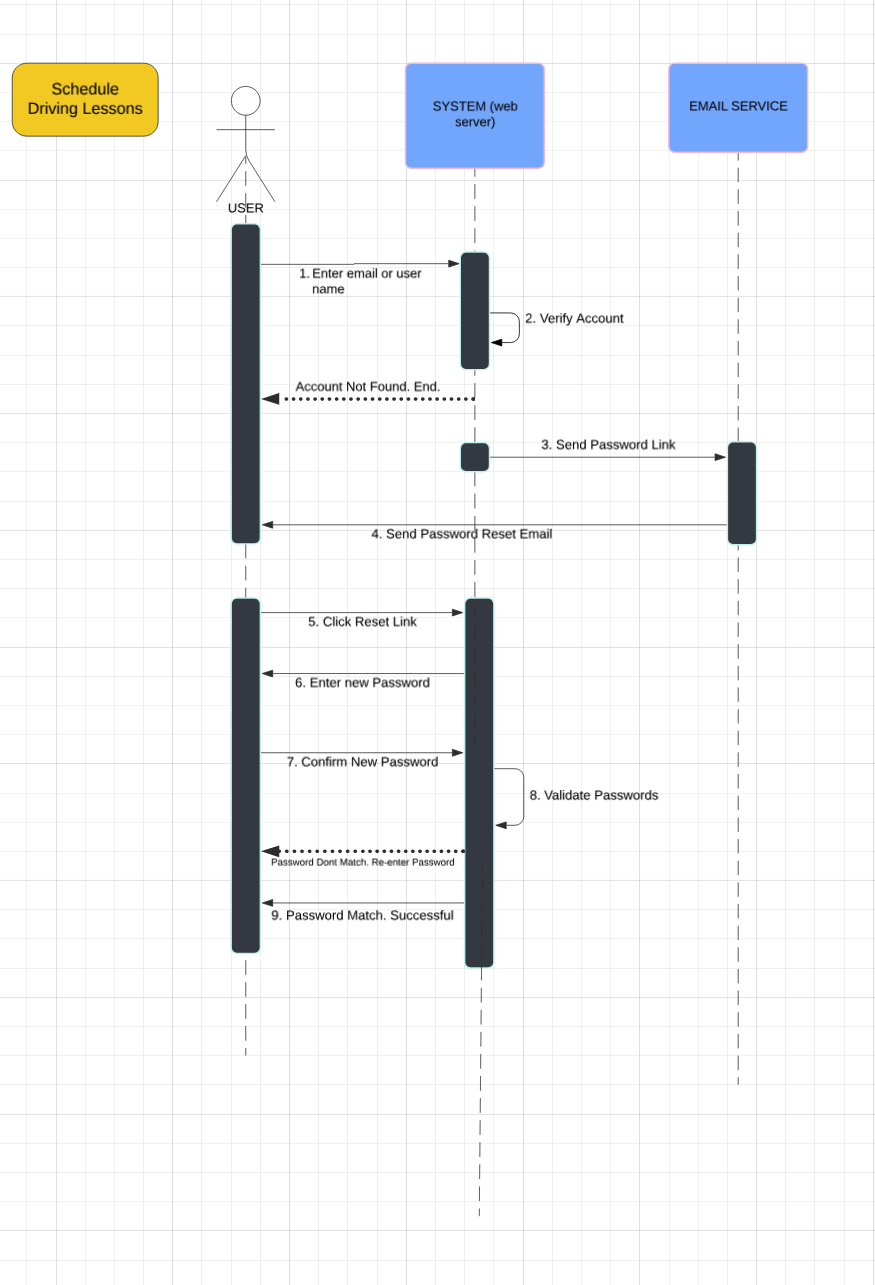
### UML Use Case Diagram

**

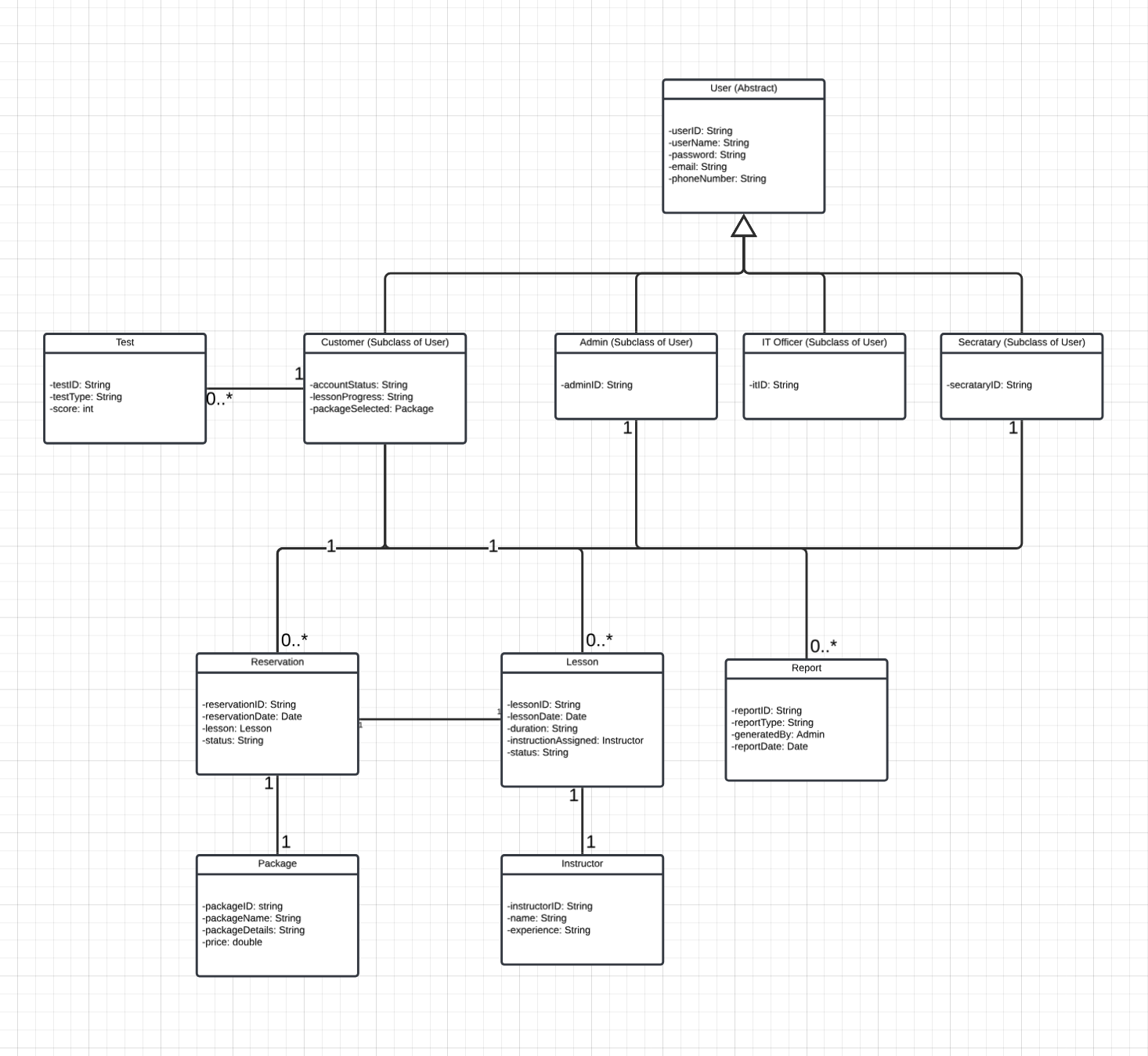
### UML Activity Diagrams

**

### UML Sequence Diagram

**

### UML Class Diagram

**

### Technical Requirements

**1. Hardware Requirements:**

* **Cloud Hosting**: The system will be hosted on a cloud platform like AWS, ensuring scalability, security, and availability, with built-in disaster recovery.
* **Client Devices**: Accessible via desktop, laptop, tablets, and smartphones using modern web browsers (Chrome, Firefox, Safari).
* **Network**: Reliable internet with sufficient bandwidth for real-time interactions and secure data transmission.

**2. Software Requirements:**

* **Web Application**: Backend built with **Java Spring Boot**; front-end developed using **HTML5, CSS3**, and **JavaScript** frameworks like **React** or **Angular** for responsive design on all devices.
* **Database**: **MySQL** or **PostgreSQL** relational database for managing user data, lessons, and reservations, ensuring transaction consistency.
* **Security**: **SSL/TLS encryption** for secure transmission, **RBAC** for role-based access control, password recovery, and account lockout after multiple failed logins.
* **DMV Integration**: API-based integration for fetching updates from the DMV and triggering notifications for new rules and materials.

**3. Tools and Development Frameworks:**

* **Development Tools**: **IntelliJ IDEA** for Java development, **Git** for version control, and **GitHub** for collaboration.
* **Database Management**: **MySQL Workbench** or **pgAdmin** for database interaction.
* **Testing**: **JUnit** for backend unit testing.