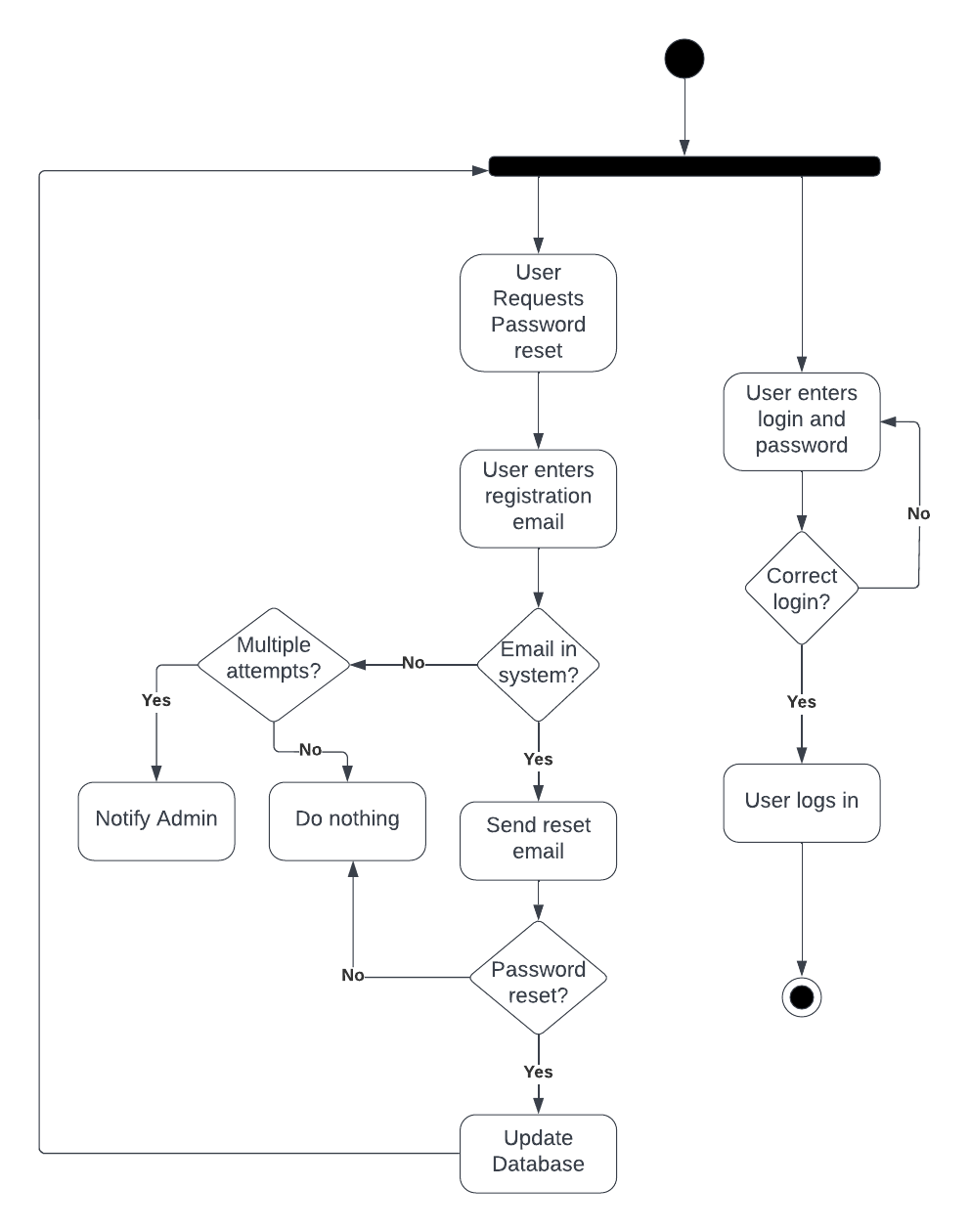
# CS 255 System Design Document

## UML Diagrams

### UML Use Case Diagram

### UML Activity Diagrams

For my activity diagrams, I made one for logging in, and one for scheduling appointments.

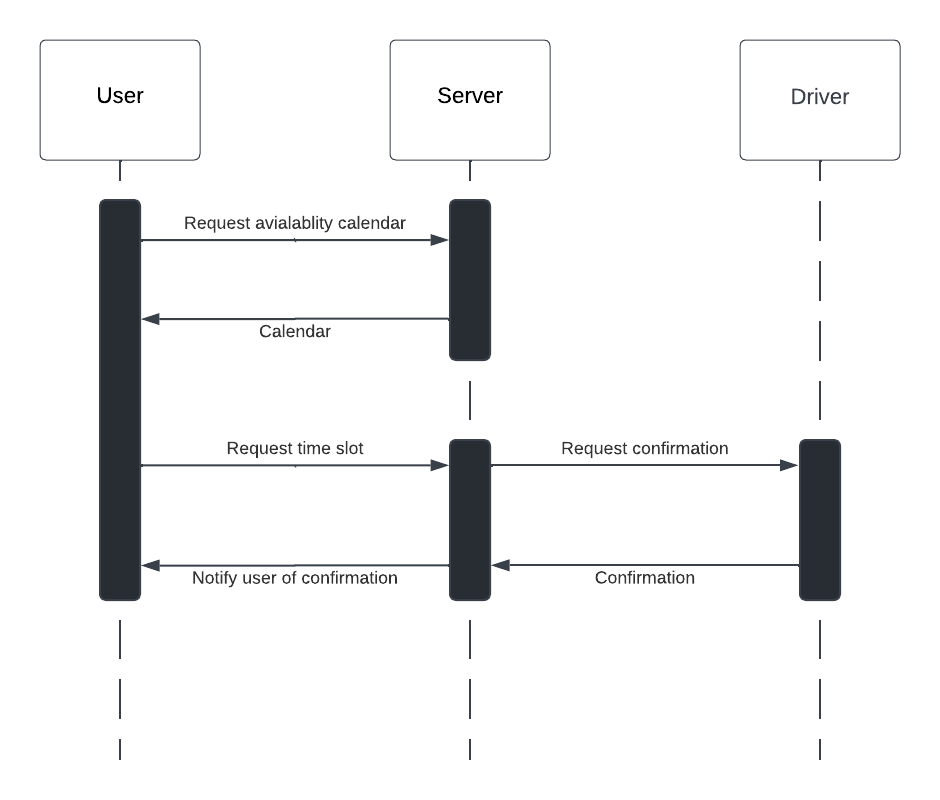
Logging in:

### Scheduling Appointments:

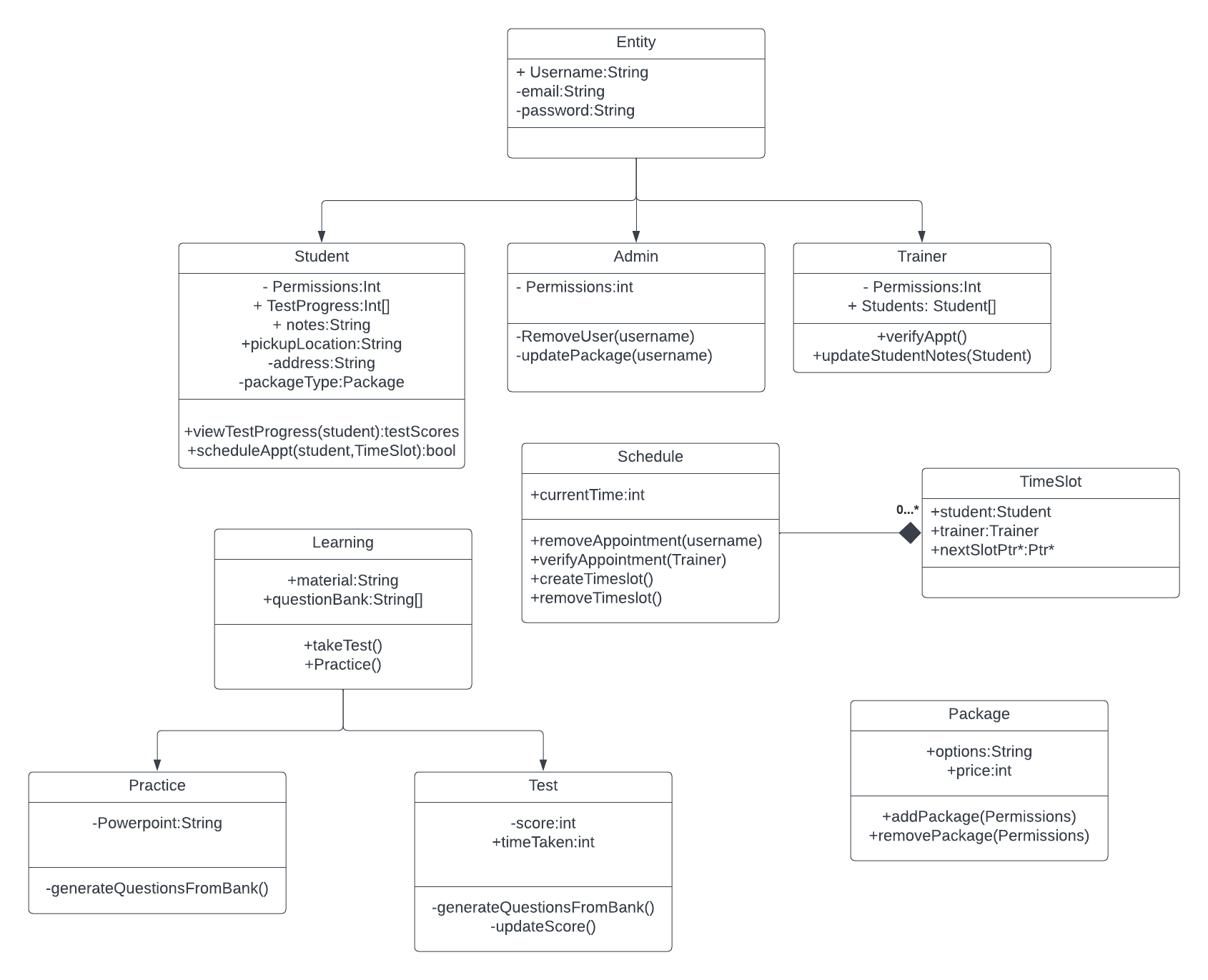
### 

### UML Sequence Diagram

This diagram shows the sequence of events in which the user schedules an appointment with a trainer (Driver).



### UML Class Diagram



## Technical Requirements

For this system to work, it would need a server-client relationship. To do this, we need a server to host the application, as Liam, the owner, said he would like to access the system from anywhere with no application download. The frontend can be built with HTML/CSS with a bootstrap framework. For embedded scripts, REACT or Angular would be fine. For server hosting, to reduce costs and maintenance, going with a serverless stack will reduce stress on Liam running this business. His program is pretty light, so he won’t be paying much for space on a server, also the system should scale with the amount of users logged on. It is assumed the user will be able to log in on a web browser, either Chrome, Firefox, or Edge would be fine. Mobile browsers might act a little weird, as budget and time restraints make it not feasible to test every device. Finally, the user must have an internet connection to access the site.

In terms of development, IDE’s such as eclipse, VScode or apache would be fine for development.