# CS 255 System Design Document

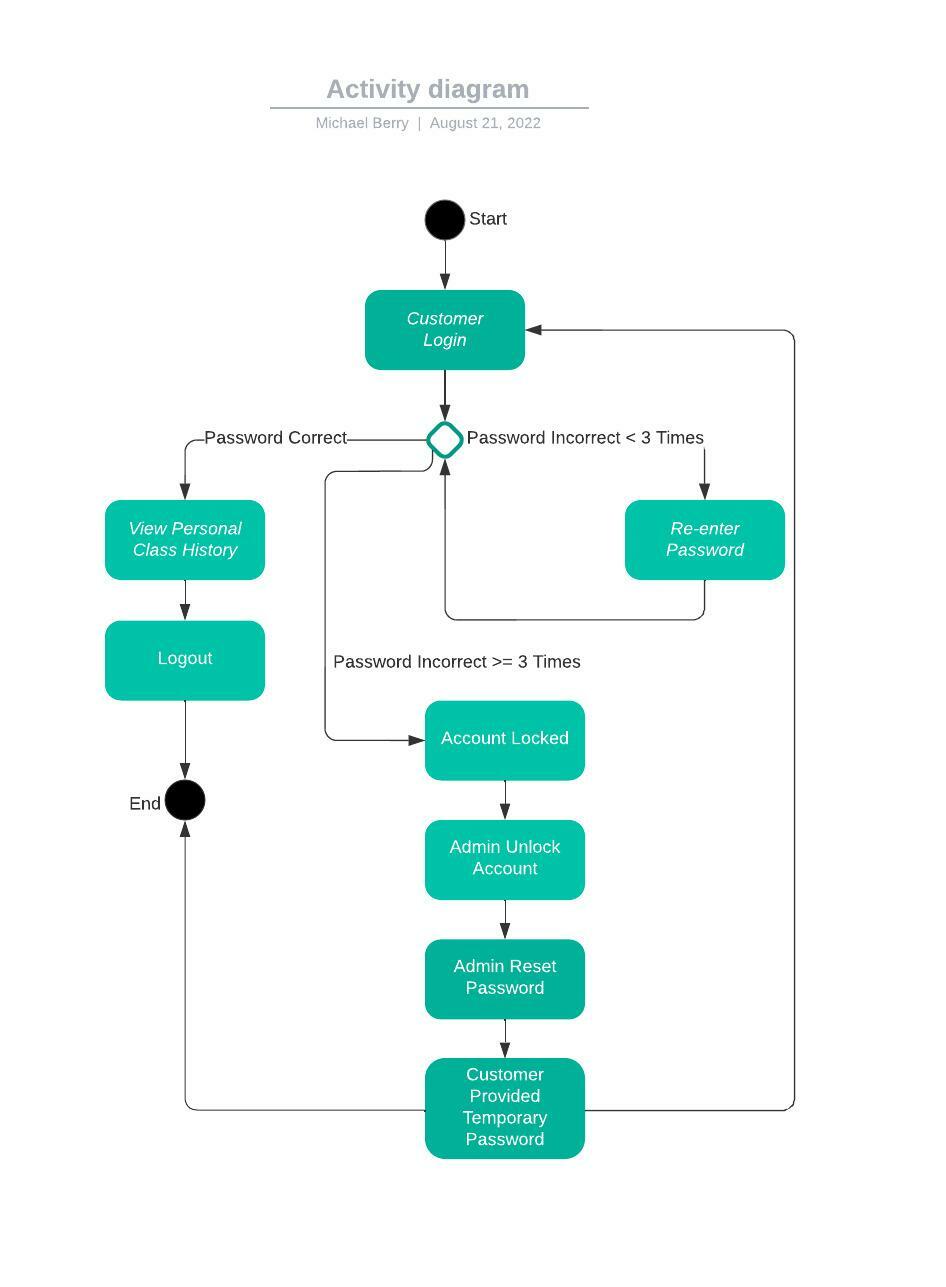
## UML Diagrams

### UML Use Case Diagram

*Diagram, schematic

Description automatically generated*

### UML Activity Diagrams



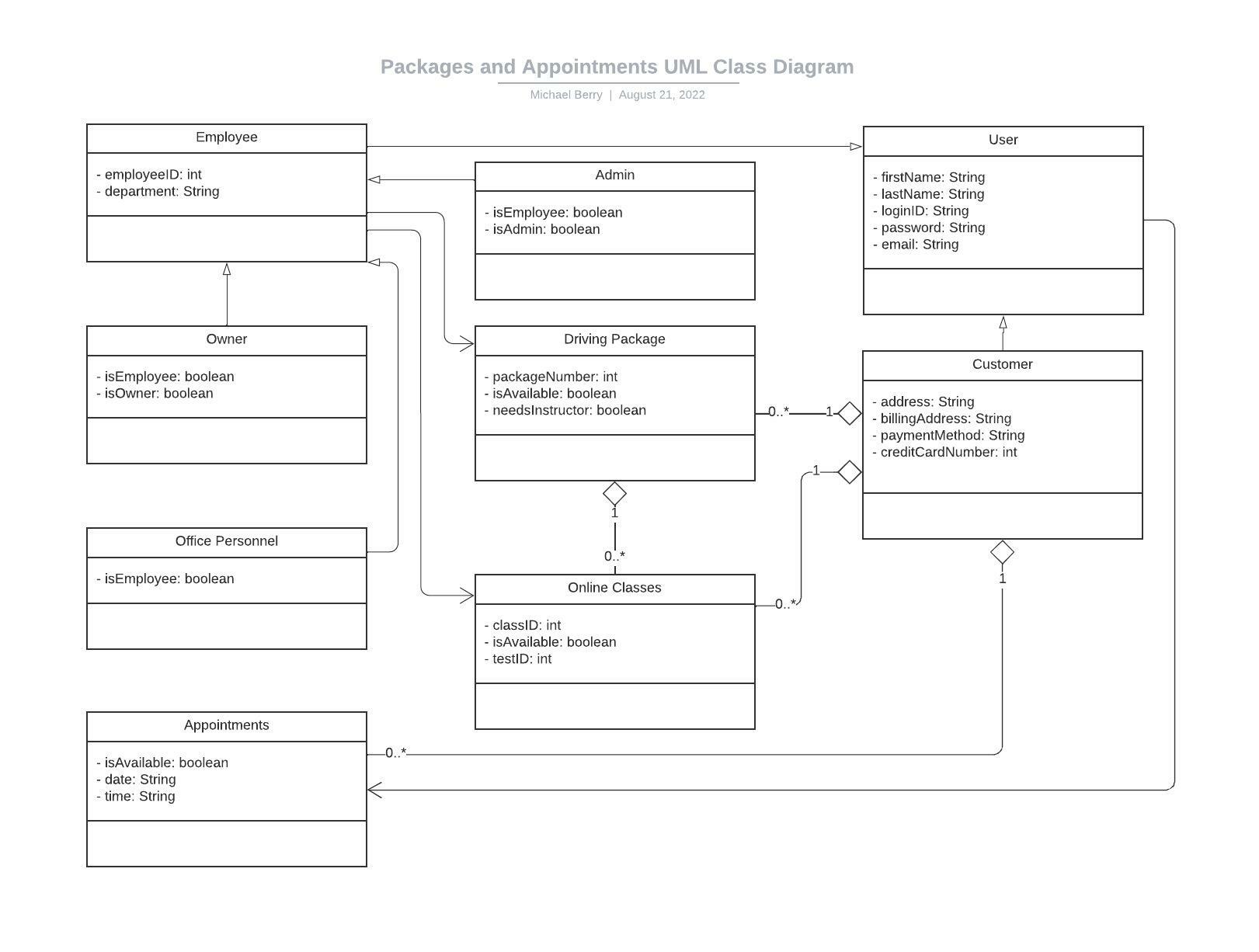
### Diagram Description automatically generated

### UML Sequence Diagram

Diagram

Description automatically generated

### UML Class Diagram



## Technical Requirements

The client has requested that the system run in a cloud environment and be accessible through both desktop and mobile devices. The system should update any time there is new user information added or a profile updated. The system should automatically back up at the end of each day. Currently, there are no specific performance requirements pertaining to the speed of the system. The system should be able to run on Windows, mac OS, Unix, and mobile platforms, as the client has requested accessibility from virtually any device. The back end will require access to a customer information database and a means to perform system updates and maintenance.

To distinguish between different users, we will implement a user profile system accessible with a login ID and password created by each user. For convenience and security’s sake, I recommend case-sensitive login IDs and passwords. Generic user input fields not used for identification should not be case-sensitive to make the system more user-friendly. A login ID and password created by the user will be needed to log into the system. To secure data transfer the IT admin will need to install an SSL/TLS certificate on the server side to utilize HTTPS. In the event of a brute force attack, the system should lock out the account after three failed login attempts and the IT admin will have to be contacted to unlock the account and assist with password reset. There will be options on the login screen for users to reset their login ID or password if they forget them. One or more forms of identification will be needed to change login ID’s or passwords (i.e., registered email access, security questions, PIN).

Changes can be made to a user account by office personnel or the user without changing code. The system will run over the web through a cloud environment using cross-platform code such as Java, so platform updates should not be a problem. The IT admin will need root-level access to perform updates, maintenance, and minimal security functions. Most security will be handled by a third party.

The system will require the following functionality:

* The system shall validate user credentials when logging in.
* The system shall track changes to records in the system.
* The system shall print record activity report.
* The system shall display information about driver lessons packages offered by DriverPass.
* The system shall allow customers to make reservations for driving lessons, date, and time on their account.
* The system shall allow customers to modify or cancel reservations while logged into their account.
* The system shall track which customer is scheduled for which car, driver, and time slot.
* The system shall allow the DriverPass owner to disable reservations for any or all driver lessons packages.
* The system shall, in customer account registration, request the user’s first name, last name, address, phone number, state, credit card number, card expiration date, and card security code.
* The system shall display student information, online test progress, driver notes, driver photo, student photo, and special needs.
* The system shall display separate page for driver notes, showing lesson time, start and end hours, and driver comments.
* The system shall take user account input.
* The system shall display DriverPass contact information.

The client has a template that is to be used in creating the user interface. It needs to display customer information (first name, last name, address, city, state, zip code, phone number, email address, etc.), online test progress, diver notes, special needs, driver photo, and student photo. The user will need to be able to input or update their information, view and purchase DriverPass packages, make reservations to complete their program, view, modify, or cancel reservations, view their progress throughout a purchased program (package), view feedback from their driving instructor, and email DriverPass support. Perhaps a chat function could be suggested later.