# CS 255 System Design Document

Michael Reynolds

## UML Diagrams

### UML Use Case Diagram

Diagram

Description automatically generated

### UML Activity Diagrams

*Use Case One: User login and book road test.*

### Diagram Description automatically generated

*Use Case Two: Driving Instructor login and check scheduled road tests.*

### Diagram Description automatically generated

### UML Sequence Diagram

A picture containing chart

Description automatically generated

### UML Class Diagram

Diagram

Description automatically generated

## Technical Requirements

The technical requirements are heavily dependent on the software. Due to the multi-platform requirements, the software must be available on mobile/web/desktop. If we consider the functional/non-functional requirements from our business requirements, we can consider technical requirements for them.

### Performance Requirements

* Multi-Platform, Providing the application on both desktop, mobile, and web-based to provide for access from multiple device types.
* The system should run with constant updates, using web sockets to provide updated info at all times.
* The system should constantly be attempting updates with DMV standards, requiring routine maintenance to ensure the application and its resources are up to date.

### Platform Constraints

* The application will need to be able to run on Windows, Unix, Android, iOS in order to be accessible on all platforms. The application should also be web-based to provide access through web browsers.
* The back-end will require a database and need to be run in the cloud, likely with Linux server distribution.
* The database will require middleware and a thorough API for providing database CRUD operations.

### Functional Requirements

* The system shall validate user credentials when logging in.
* The system shall provide users with test/road grading and completion.
* The system shall provide secure access to user account management
* The system shall provide access from multiple devices and web-browsers.
* The system shall provide scheduling appointments and road tests.
* The system shall provide payment processing for online package purchases.
* The system shall provide records of user actions and modifications for administrative purposes.
* The system shall provide a separate dashboard for instructors and admins.
* The system shall provide an administrator product package plan add/delete interface

For hardware/infrastructure requirements the design will require the following:

* Cloud database with cluster backups and manual IT administration
* Local storage availability on client-side device
* Location Services to provide Locality DMV data

Software-wise we’re depending on multi-platform support:

* HTML/CSS/JS on Web
* C# or PWA for windows desktop app
* Swift for MacOS, iOS
* Kotlin for Android

Some of the third-party tools/resources we’ll use to develop include:

* Possibly CircleCI for CI/CD pipelining
* Locality tools to update according to local DMV data
* Framework for payment processing
* CPanel for web app management