# CS 255 Business Requirements Document

## System Components and Design

### Purpose

* The aim of this project is to create an online system for DriverPass, a company focused on helping students prepare for their DMV driving tests. DriverPass wants to provide students with online courses, practice exams, and on-the-road training options. The system will enable scheduling and management of driving lessons, track progress, and generate relevant reports.

### System Background

* DriverPass identified a problem where many students fail their driving tests due to inadequate preparation. To address this issue, DriverPass aims to develop a training platform that provides more comprehensive learning opportunities. The system will need to support online classes, practice exams, and scheduling of driving lessons. It will also handle reservation management for both online and in-office appointments, allow for user account management with different roles for administrators, staff, and customers, and track user activities, such as lesson reservations and test progress.

### Objectives and Goals

* The system should provide online practice exams and on-the-road driving lessons. It must allow customers to schedule, modify, or cancel their driving lessons and enable DriverPass staff to manage appointments and assign driving instructors to customers. The system will also need to track and report user actions, such as reservations, cancellations, and changes. Additionally, the system will offer various training packages with different services, such as driving sessions, in-person lessons, and online classes. System administrators must be able to manage user roles and reset accounts when necessary.

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

* The system needs to run as a web-based application, allowing users to access it from any device with internet connectivity. It should operate with minimal latency, processing requests and responses in real-time to ensure a smooth user experience. The system should be designed to handle multiple users simultaneously, scaling as needed to support high traffic, especially during peak hours like pre-exam periods. System updates should be scheduled quarterly, with emergency patches released as necessary.

#### Platform Constraints

* The system should be cross-platform compatible, functioning on Windows, Unix, and macOS. Additionally, the system will require a robust backend supported by a relational database such as MySQL or PostgreSQL to manage user data, including exam scores and training records. The application will also need to integrate with cloud services for scalability and data backup.

#### Accuracy and Precision

* To distinguish between different users, the system will assign unique user IDs and roles (student, instructor, admin) upon registration. All inputs, such as usernames and passwords, will be case-sensitive to maintain accuracy. The system will notify the administrator of any errors related to user authentication, system malfunctions, or data inconsistencies.

#### Adaptability

* The system should allow administrators to add, remove, or modify user profiles through an intuitive admin interface without requiring code changes. It should be adaptable to new platform updates (e.g., operating system or browser versions) through periodic testing and patches. The IT admin will need access to system logs and user data for troubleshooting and maintenance purposes.

#### Security

* Users must log in using two-factor authentication to ensure secure access. Data exchanges between the client and the server will be encrypted using SSL/TLS protocols. In the event of attack, the system will temporarily lock the account after five failed login attempts and send a notification to both the user and the admin. If a user forgets their password, they will be prompted to reset it through a secure email verification process.

### Functional Requirements

* The system will validate user credentials during login, allow students to access practice exams, provide a dashboard for instructors to track student progress, allow students to schedule on-the-road training sessions with instructors, track and store student exam performance data for review, notify students of upcoming training sessions and exams, and enable administrators to manage user roles and permissions.

### User Interface

* The interface must be accessible from both mobile and desktop devices. The primary users include students, instructors, and admins. Students should be able to easily navigate through the available practice exams, view their progress, and schedule training sessions. Instructors need access to student progress data and the ability to provide feedback through the interface. Administrators should have a clear, efficient interface to manage user roles, system settings, and access logs.

### Assumptions

* We assume that all users will have access to stable internet and modern devices capable of running a web-based application. Additionally, we assume users are familiar with basic web navigation. For the back-end infrastructure, we assume the client has the necessary server capacity and IT personnel to maintain the system.

### Limitations

* One limitation is that the system relies on users having a stable internet connection, which may not always be available, especially in remote areas. The system may also face budget constraints that limit the frequency of updates or enhancements. Time constraints could impact the thoroughness of testing, leading to potential bugs or security vulnerabilities during the early phases of deployment.

### Gantt Chart

