# CS 255 Business Requirements Document

Dusty Cook/Professor Madeley/04-04-2025/CS255

## System Components and Design

### Purpose

*What is the purpose of this project? Who is the client and what do they want their system to be able to do?*

* This project aims to create and execute a comprehensive system for DriverPass, a company dedicated to enhancing student success in driving tests. DriverPass has recognized a significant gap in the market: many students still depend on outdated written exams for preparation, resulting in low passing rates. The objective of this system is to offer students an engaging and interactive platform that features realistic online practice exams and tools for scheduling on-the-road driving instruction. This system is designed to align with the company’s mission of better preparing students for success and facilitating the organization of instructors and lessons.

### System Background

*What does DriverPass want the system to do? What is the problem they want to fix? What are the different components needed for this system?*

* DriverPass aims to tackle a significant issue in driver education: the gap between students' exam preparation and the actual skills required to succeed. Many learners focus solely on old written tests, neglecting the vital hands-on experience needed for success. To address this, DriverPass will provide both intelligent, adaptive practice exams and opportunities to book real-world training with driving instructors. The platform must support user accounts for students, instructors, and IT/admins, manage scheduling, deliver exam content, track progress, and perform administrative tasks. This multi-user system should be user-friendly, scalable, and accessible on various devices.

### Objectives and Goals

*What should this system be able to do when it is completed? What measurable tasks need to be included in the system design to achieve this?*

* By the project's end, the DriverPass system must enable students to register, log in, and take practice exams that closely resemble the format of official driving tests. Additionally, students should have the option to schedule driving sessions with instructors, who can manage their own schedules and monitor student progress. IT personnel and administrators need the ability to oversee user accounts, adjust system settings, and manage content. We will evaluate success based on system performance, user satisfaction, usability, and ultimately, improved driving test pass rates. Important milestones include launching the system, verifying the effectiveness of exam functions, and fully implementing scheduling features.

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

*What environments (web-based, application, etc.) does this system need to run in? How fast should the system run? How often should the system be updated?*

* The DriverPass system needs to function as a web application that is accessible via major browsers on desktop and mobile devices. It should be performance-optimized, with page load times not exceeding three seconds and aiming for 99.9% uptime. Additionally, the system must support concurrent use by multiple users, such as students taking exams and instructors managing appointments. Content, including exam questions, should be updated regularly- preferably every quarter- to ensure it aligns with current driving standards and regulations.

#### Platform Constraints

*What platforms (Windows, Unix, etc.) should the system run on? Does the back end require any tools, such as a database, to support this application?*

* The system needs to operate on a Windows server and be compatible with browsers like Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge. It should utilize a relational database management system, such as MySQL or PostgreSQL, for storing user data, exam results, and scheduling information. The backend development will leverage a reliable programming language and framework, preferably Node.js or Python, to guarantee scalability, maintainability, and security. Additionally, the system must be hosted on a secure cloud platform to facilitate remote access and simplify deployment.

#### Accuracy and Precision

*How will you distinguish between different users?* *Is the input case-sensitive? When should the system inform the admin of a problem?*

* User identity must be carefully managed using unique identifiers like email addresses or student IDs. The system needs to enforce rigorous validation on input fields to ensure data accuracy and minimize input errors. Standardized input formats should be utilized, implementing features such as case sensitivity, password strength checks, and email verification. In the event of system errors or scheduling conflicts, administrators should receive immediate notifications through dashboard alerts or automated messages.

#### Adaptability

*Can you make changes to the user (add/remove/modify) without changing code? How will the system adapt to platform updates? What type of access does the IT admin need?*

* The DriverPass system should be crafted for flexibility. It must facilitate the effortless addition, removal, or modification of users and content via the administrative interface, eliminating the need for direct modifications to the codebase. The platform should be ready to adapt to future changes in browser technologies or industry standards. IT administrators need access to comprehensive system controls, including settings for user permissions, reporting tools, and backup management. Additionally, the system should be structured to integrate seamlessly with future services, like video-based lessons or mobile apps.

#### Security

*What is required for the user to log in? How can you secure the connection or the data exchange between the client and the server? What should happen to the account if there is a “brute force” hacking attempt? What happens if the user forgets their password?*

* The system should implement secure login methods, featuring email/password authentication along with optional two-factor authentication for administrators. All data exchanged between the client and server must be encrypted with SSL/TLS protocols. Additionally, the system needs to have measures in place to prevent brute-force attacks, including locking accounts after several unsuccessful login attempts. In the event a user forgets their password, there should be a secure recovery process that uses token-based email verification. Administrative capabilities must be limited to authorized personnel only, while audit logs maintain a record of all significant activities at a high level.

### Functional Requirements

*Using the information from the scenario, think about the different functions the system needs to provide. Each of your bullets should start with “The system shall . . .” For example, one functional requirement might be, “The system shall validate user credentials when logging in.”*

* The system will enable students to register for and manage their accounts.
* It will verify user credentials during login.
* The system will create randomized practice exams from a secure question bank.
* It will track student performance on each practice exam.
* Students will be able to view their progress and test history.
* They can schedule, reschedule, and cancel driving lessons.
* Instructors will be able to manage their availability and appointments.

• Notifications of upcoming lessons will be sent to students and instructors via email or SMS.

• IT administrators will have tools for managing users, content, and schedules.

* The system will log errors and alert administrators to critical issues.

### User Interface

*What are the needs of the interface? Who are the different users for this interface? What will each user need to be able to do through the interface? How will the user interact with the interface (mobile, browser, etc.)?*

* The user interface should be intuitive and responsive, compatible with both desktop and mobile browsers. There are three main user groups: students, instructors, and administrators. Students require access to exam features, lesson scheduling, and performance tracking. Instructors need to be able to view their schedules, mark attendance, and monitor individual student progress. Administrators and IT personnel require tools for managing users, configuring the system, and generating reports. The design must adhere to accessibility standards to ensure usability for all users, including those with disabilities.

### Assumptions

*What things were not specifically addressed in your design above? What assumptions are you making in your design about the users or the technology they have?*

* We assume that all users can access reliable internet connections and modern web browsers. Similarly, instructors and administrators are expected to possess basic computer skills and will receive training if needed. Initially, the system will be developed solely for web use, with the possibility of expanding into mobile applications later on. DriverPass is expected to supply the content for the practice exams and establish rules regarding instructor availability.

### Limitations

*Any system you build will naturally have limitations. What limitations do you see in your system design? What limitations do you have as far as resources, time, budget, or technology?*

* The project's budget and timeline constraints might limit the addition of advanced features such as real-time chat, AI-assisted tutoring, or a dedicated mobile app during the initial phase. Moreover, the system's functionality hinges on the availability and participation of instructors, potentially impacting scheduling options. The implementation of SMS/email notifications could be contingent on third-party services that incur additional costs. Additionally, backend scalability may face limitations depending on the selected hosting solutions.

### Gantt Chart

*Please include a screenshot of the GANTT chart that you created with Lucidchart. Be sure to check that it meets the plan described by the characters in the interview.*

