# **CS 255 Business Requirements Document Template**

Kayden Sysum

Doctor Stefanelli

August 3, 2025

## **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?*

* DriverPass would like a system built for their current business model. DriverPass gives drivers the opportunity to practice and take tests in preparation to help obtain a DMV driver’s license. The system has been designed to expand and grow their business with the addition of online user access, scheduling, and payments. This also helps employees manage DriverPass by allowing access to user data which can improve scheduling, bugs, tracking, and their driver.

### **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 has noticed the problem at large, many people fail the DMV test due to lack of preparation. The goal is to offer better practice tests and training.
* The system needs to let users create accounts, schedule lessons or tests, instructor, vehicle selection, support/rescheduling. Cloud is based on functions on phones, computers, and tablets.

### **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?*

* The goal of the system is to provide DriverPass with a completely modern experience that allows customers to schedule, train, test, and track their goals all while being online.
* Create secure user accounts – Develops a login system that allows users to secure private information within their account. Data is stored in the cloud’s database.
* User Interface – Design a clean easy to operate interface that can display information such as progress, notes from instructors (attached to lessons), schedule, and account settings. The final design must be directly approved by DriverPass management/lead.
* Scheduling – build a function so users can see instructors, paired cars, along with the location and time of lesson. The system tracks drivers and students’ lessons.
* Roles – Configure Customer, Employee, and Admin roles, each with access to tools and dashboards relevant to their duties.
* Updates – Establish a process that enables real-time updates from administrators. Alerts are synchronized with designated roles, ensuring timely response to notifications as they arise.
* Management – Allows for students to reschedule, cancel, all in the online dashboard. You can choose selected lessons and payments online.
* Testing – Static and dynamic testing over at least 10 days. The team delivers features agreed upon.
* Delivery – Encompasses the provision of the final, fully functional product, complete with all features approved and rigorously tested. The system offers driving training, online accessibility, and comprehensive data management capabilities for both staff and customers.

## **Requirements**

### **Nonfunctional Requirements**

*In this section, you will detail the different nonfunctional requirements for the DriverPass system. You will need to think about the different things that the system needs to function properly.*

#### **Performance Requirements**

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

* DriverPass must ensure the operations are running smoothly in various web browsers such as Chrome, Firefox, Safari, and Edge. Not only must these run well on desktops but mobile as well. The platform requires stable internet for video streaming content to be viewed.
* The user’s internet will affect the playback quality of a video, but the system ensures support to multiple streams without lag or crashes.

#### **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?*

* A mobile app will be developed for iOS and Android, with a Windows phone version planned. On the front-end, users need primary access to the system through either web-based application, desktop version, and mobile application. Web based applications support Windows and macOS without issue. All web versions will work on any modern device.
* On the back end the system requires a framework that is capable of handling business logic, database, and APIs. The database is secure with SQL as this is strongly compatible with almost all backend systems and handles precise documentation needed for a project of this scale.
* Cloud is deployed across the system making integration like backup, restoration, and logs all fully protected through the system’s integrity. Platforms such as Azure or AWS deploy the systems supporting uptime and load balancing.

#### **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?*

* Accuracy handling is critical for the user’s experience but also keeps a user’s information private. Each user has an ID, and password kept the case sensitive needed for login. For example, ‘Student’ and ‘student’ are two separate accounts where the login form verifies input and displays a unique error message for duplicates or errors such as “Username exists” and “Password Does Not Match”.
* For account creation the system analyzes the database looking for duplicates, rejects existing usernames, and prevents overwriting. Admins are notified if there’s a significant spike in resources such as hardware limitations, high usage, and memory processing. All of this is logged and timestamped for admin diagnostics and is reviewed by the admin as well. Any anomalies in performance, accounts, or down services trigger alerts.

#### **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 is designed for scalability behind the user interface. The user is treated as a larger data structure, this means that students, instructors, and administrators are all acted upon as objects within the framework. Users can be updated, changed (personal information: name, address, date of birth, payment), onboarded by admin, and even terminate an account without having to release a new patch.
* The system adapts to platform updates, for example the Android version may receive an update to address a specific bug to that system. Maintaining versions up to web standards and testing is verified and deployed within the shortest window possible.
* Admins have heightened privileges within the system to manage the interface, logs, status, and user dashboards. Although admins have higher privileges than users, admins do not have access to sensitive information payment methods or any financial details, enforced through role-based access controls.

#### **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?*

* Security is the top priority in the DriverPass system, which uses a robust login authenticator that meets industry standards for handling usernames and passwords (industry standard 10 characters, including numbers, upper/lowercase letters, and symbols). The system also has two factor authentication known as “2FA” which adds another level of protection by requiring an additional password before logging in.
* Data is encrypted with transport layer security that uses HTTPS servers to prevent malicious attacks and interceptions.
* There is additional security to the system that limits the number of tries a user can attempt login. For example, if unexpected brute force hacking commences a user can fail up to a maximum of five times within an hour-long period before temporary lockout occurs. An email is sent to the user after providing details on resetting a password or recovering an account. Admins are alerted when a system trips on account of suspicious activity.

### **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 shall validate user input upon login.
* The system shall enforce strong password creation and store passwords = securely using hashing algorithms.
* The system shall support two-factor authentication for added protection.
* The system shall track and display user's progress toward completion of training and testing.
* The system shall store all user data in private accounts securely.
* The system shall allow users to schedule driver tests and driving practice based.
* The system shall display instructor and car availability for reservation.
* The system shall allow users to make secure online payments.
* The system shall manage employees and schedules.
* The system shall log all user activity and data updates for auditing purposes.

### **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 is designed to adhere to customers, employees, and administrators through roles in the DriverPass system. The system is accessed from various devices at any given time the interface must be able to respond, and function across all the supported platforms. This encompasses comprehensive mouse and keyboard capabilities, as well as complete touch screen functionality on mobile devices and tablets.
* Navigation is straightforward, which keeps the personalized dashboard simple and easy for new users. Clearly labeled tasks and instructions paired with icons, interactive guides, and task status are all included in the user experience. Forms may be signed by accessing the dropdown menus, which provide a range of customization options, including interactive scheduling features that display both availability and reservations. Minimal data entry errors allow for a smooth modern feel.
* Customers have a log in view to a customized dashboard unique to each user’s needs. From here there are displays of current training, completed lessons, lesson grades, instructor notes, and the number of sessions until completion bar. A user can update the address, phone number, email, and any payment details at any given time in the account settings. In the interface there are several dropdowns including rescheduling, cancellation, lessons, and locations.
* Employees have a unique dashboard that shows assigned lessons, vehicle assignments, and student profiles. The instructors have access to a list of availability, vehicle report, request changes, and add notes tailored to any specified lesson for one-on-one feedback.
* Administrator accesses a full control panel with complete system oversight. They can create or delete users, manage, and assign roles, manage vehicles, manage instructors, staff schedules, address alerts, run diagnostics, analyze data, reset passwords, if necessary, overseas transactions (details are kept private and secure), and all DMV announcements. Administrators operate behind the interface to ensure optimal system performance.

### **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?*

* Technology Access – Assumes both customers and staff have access to the system or a relatively modern device that can operate within a web-based application. The system has no legacy features supporting hardware, software, or any outdated browsers unless proven needed.
* Stable Connection – The system is designed for users to have a stable connection, optimization for slower connections will be made, partial offline functionality.
* Code Unknown – Coding language used to write the system currently unknown.
* Hosting – Unknown where the host server will be. The best options were AWS and Azure.
* Aesthetic – color scheme and presentation are unknown.
* Legal Regulations – unknown data protection, assumes system complies with GDPR or CCPA

### **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?*

* Limitations come up depending on how much time the team is given and with what size of team it requires in relation. For example, if the team is smaller and builds an entire system in a set time frame some features have the potential of being scaled back to ensure deadlines are hit.
* Database has a lack of bandwidth to the host then some features in the interface may be tweaked for a smooth user experience. Something as small as simpler animations or low resolution can help especially to users with poor connectivity.
* No budget for the project. If the budget is running high or if expectations cannot be met, then a meeting with DriverPass management must be held for renegotiation.

### **Gantt Chart**

A gantt chart with a row of rectangular objects

AI-generated content may be incorrect.