# CS 255 Business Requirements Document Template

Complete this template by replacing the bracketed text with the relevant information.

This template lays out all the different sections that you need to complete for Project One. Each section has guiding questions to prompt your thinking. These questions are meant to guide your initial responses to each area. You are encouraged to go beyond these questions using what you have learned in your readings. You will need to continually reference the interview transcript as you work to make sure that you are addressing your client’s needs. There is no required length for the final document. Instead, the goal is to complete each section based on your client’s needs.

**Tip:** You should respond in a bulleted list for each section. This will make your thoughts easier to reference when you move into the design phase for Project Two. One starter bullet has been provided for you in each section, but you will need to add more.

## 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 is a company focused on improving driver test success by providing online classes, practice tests, and in-person driving lessons.
* The goal of this project is to build a system that allows students to take practice exams online, schedule and manage driving lessons, and track their progress.
* The system should also let staff (secretaries, instructors) and administrators easily manage appointments, users, and data securely from anywhere.

### 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?*

* Many people fail their DMV driving tests on their first try, which causes frustration and extra costs. DriverPass wants to fill this gap by combining online learning with real driving practice.
* The system must support different lesson packages, scheduling, and tracking of lessons, cars, and instructors.
* It should also connect to the DMV to get updates on rules, policies, and test content, notifying DriverPass when changes occur.
* Users should be able to access reports and data online and offline (e.g., download reports to Excel).
* Security and user role management are key, with different access levels for IT staff, secretaries, instructors, and students.

### 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 system shall allow customers to create accounts, view available packages, and schedule or modify driving lessons online.
* The system shall support scheduling of driving lessons that are two hours long, matching students with specific cars and instructors.
* The system shall enable secretaries to schedule and manage appointments via phone or in-person.
* The system shall track user actions (who booked, modified, or canceled lessons) with clear audit logs.
* The system shall allow administrators to manage users, reset passwords, and block access when needed.
* The system shall enable the disabling of lesson packages so DriverPass can stop new registrations on certain packages without removing them entirely.
* The system shall connect with DMV systems to receive updates and notify DriverPass of changes to exam rules or questions.
* The system shall provide a user interface that displays student test progress (status, scores, times) and lesson details with driver comments.
* The system shall be web-based, cloud-hosted, and accessible from any modern computer or mobile device.

## 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.) does this system need to run in? How fast should the system run? How often should the system be updated?*

* The system must be **web-based** and accessible via common browsers (Chrome, Firefox, Safari, Edge) on both desktop and mobile devices.
* It should respond to user actions within **2 seconds** under normal load.
* The system must support **simultaneous access** by at least 100 users with no degradation in performance.
* Updates to the platform should be deployable **biweekly**, with zero downtime during low-usage hours.

#### 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 should run on a **cloud-based platform** (such as AWS or Azure) to ensure scalability.

 The application must be cross-platform, requiring **no client-side installation**.

 The backend must use a **relational database** such as **MySQL or PostgreSQL** for managing users, scheduling, and lessons.

 Frontend should be responsive and built using **HTML5/CSS/JavaScript** frameworks such as **React** or **Vue**.

#### 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?*

 The system shall enforce **unique usernames and emails** to prevent duplicate accounts.

 Login credentials will be **case-sensitive**, and usernames must meet specific validation rules.

 The system shall notify administrators of **any scheduling conflicts**, data integrity issues, or errors in external system integrations.

 Instructor/student actions shall be **timestamped** and traceable via logs.

#### 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?*

 User roles (e.g., student, instructor, secretary, admin) must be configurable through the admin panel without code changes.

 The system must support adding/removing/modifying users and lesson packages dynamically.

 Admin users shall be able to manage system settings such as scheduling parameters and notification preferences.

 The system must be compatible with **OS updates** and maintain browser compatibility through responsive web design and framework updates.

#### 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 shall require users to **log in with a secure username and password**.

 Connections must be encrypted using **HTTPS**.

 User sessions must time out after 15 minutes of inactivity.

 The system shall implement **account lockout** after five failed login attempts.

 Password recovery must include a **reset link** sent via email and optional **multi-factor authentication**.

 Admins must be able to **disable** or **suspend** accounts due to security concerns or policy violations.

### 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 credentials when logging in.
* The system shall allow students to register, view, and enroll in lesson packages.
* The system shall enable students to schedule, reschedule, or cancel lessons online.
* The system shall support secretaries in scheduling lessons via phone or in-person requests.
* The system shall match students with available instructors and vehicles.
* The system shall allow instructors to update lesson status and add comments.
* The system shall provide progress tracking and display performance analytics to students.
* The system shall generate and export reports to formats like Excel or PDF.
* The system shall allow IT staff to manage user permissions and system configurations.
* The system shall pull updates from DMV feeds and send alerts when new content is available.

### 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.)?*

* **Students** will:
  + Register and manage their accounts.
  + View and sign up for packages and lessons.
  + View lesson feedback and test progress.
* **Instructors** will:
  + View assigned students and lesson schedules.
  + Update lesson completion and feedback.
* **Secretaries** will:
  + Create and modify schedules for students and instructors.
  + Manage customer interactions offline (via phone or walk-in).
* **Admins/IT** will:
  + Oversee user accounts, lesson availability, and system integrity.
  + Monitor usage and security logs.
* The UI must be:
  + **Mobile-friendly**, responsive, and intuitive.
  + Accessible via modern browsers on desktops and mobile devices.
  + Simple and navigable, using dashboards for role-specific features.

### 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?*

 All users will have access to a modern web browser and internet connection.

 DriverPass will provide regular access to DMV updates through a public or authorized API.

 Customer support staff are trained to use the system interface.

 Students and instructors will have basic technical skills to navigate the platform.

 All backend infrastructure will be cloud-hosted with appropriate support contracts.

### 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?*

 Real-time DMV integration may depend on external API availability or state-specific restrictions.

 Budget constraints may limit the initial number of simultaneous users supported.

 The availability of instructors and vehicles may restrict lesson scheduling during peak times.

 The system’s offline capabilities (e.g., Excel exports) are limited to predefined data sets.

 Third-party software integrations may delay delivery timelines if compatibility issues arise.

 Platform scalability may require future upgrades based on adoption and user load.

### 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.*

