# 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 wants to build something that makes the process of learning to drive more effective and accessible. They’re focused on helping students succeed not just by passing a test, but by also being prepared for real driving.
* This system needs to bring together two major pieces: an online platform where students can study and take practice tests, and a scheduling tool that handles in-person driving lessons with instructors.
* The goal is to build a web-based platform that’s flexible, secure, and easy to use for students, staff, and admin alike.

### 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?*

* A lot of people fail their driving tests because they either don’t study enough or don’t get enough behind-the-wheel practice. DriverPass sees this as a gap in the market and wants to fix it by combining both types of training into one system.
* They’re offering multiple lesson packages that include a mix of online and in-person sessions. Some packages are simpler with just a few driving hours, while others include full access to practice exams and DMV rule explanations.
* The system needs to support all of that: booking lessons, assigning instructors and vehicles, tracking student progress, and updating materials when DMV requirements change.
* It also needs to work on any device, store data in the cloud, and give DriverPass staff control over who can access what.

### 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?*

**Students should be able to:**

* Sign up and create an account with their basic info and payment details
* Schedule, reschedule, or cancel their driving lessons online
* Take practice tests and see which ones they’ve passed, failed, or are still working on
* View driver notes and see their upcoming lesson schedule
* Reset their password if they forget it without needing to call someone

**Staff and instructors should be able to:**

* Log in with different access levels depending on their role
* Manually book lessons for students who call or visit in person
* See which instructor is driving which student, at what time, and in which car
* Add driver comments after each session
* Enable or disable training packages without deleting them completely

**The system should:**

* Run in any modern browser and store everything in the cloud so local backups or complicated installs aren’t needed
* Automatically receive DMV updates, especially when new test questions or policies are released
* Provide downloadable reports that Liam can review at home or in Excel
* Log activity clearly so it’s easy to see who made changes and when

## 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 should work smoothly on both phones and computers using any modern browser
* Basic actions like logging in or scheduling a session should take less than two seconds
* It should have close to 100% uptime and back itself up daily
* Updates should happen every couple of weeks, with minimal user disruption

#### 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 app should be hosted on a reliable cloud platform (like AWS or Azure)
* Data should be stored and managed using a stable database (e.g., MySQL or PostgreSQL)
* Should run well on all major browsers including Chrome, Firefox, Safari, etc.

#### 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?*

* Different roles should have clearly defined permissions
* All user forms should validate inputs and prevent incomplete or incorrect entries
* Every major action should be logged with user info and timestamp
* The system should flag suspicious activity (like multiple failed logins)

#### 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?*

* Admins should be able to turn packages on or off without touching any code.
* The system should be flexible enough to add new features down the line without a full rebuild.
* IT needs backend access to fix issues, reset accounts, and handle technical stuff without outside help.

#### 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?*

* Passwords must meet strong complexity requirements, and support two-factor authentication
* All data transfers should use SSL or other encryption
* Repeated failed login attempts should lock the account and alert IT
* Users should be able to reset passwords securely via email link

### 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 allow students to register using their name, address, phone number, payment information, and pickup location.
* The system shall verify login credentials before granting account access.
* The system shall allow students to schedule, reschedule, or cancel driving lessons based on availability.
* The system shall match each session with an instructor and vehicle according to schedule and location.
* The system shall provide access to online lessons and practice tests when included in a student's selected package.
* The system shall track and display student progress, including test scores and lesson completion.
* The system shall allow administrators to manage user roles, adjust permissions, and resolve account issues.
* The system shall maintain a log of all major actions, including account changes, bookings, and test completions.
* The system shall allow secretaries to manually enter or update appointments made via phone or walk-in.
* The system shall support DMV content updates and notify administrators when new materials are added.

### 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 interface will be designed for four main user types: students, secretaries, admins, and IT staff
* Students: log in, schedule lessons, take tests, reset password, view driver notes
* Secretary: manage appointments, enter student data, answer inquiries
* Admin: turn packages on/off, view logs, run reports
* IT Officer: full access for system maintenance and account recovery
* Must be browser-based and responsive for phones, tablets, and desktops

### 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?*

* Users will have reliable internet and basic comfort using online forms
* DMV will provide timely updates when their requirements change
* Cloud hosting will include daily automated backups
* Most package options will remain stable in the short term

### 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?*

* Customers won’t be able to create their own custom packages at launch
* DMV integration may not be real-time, requiring occasional manual updates
* Budget or time constraints might delay a full-featured mobile app or AI tutoring tools
* Admin features may evolve over time as user feedback shapes what's most useful

### 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.*

A diagram with multiple colored boxes

AI-generated content may be incorrect.