# 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?*

* The client is DriverPass, a company helping people prepare for their DMV driving tests.
* They want a system that lets users take online classes, schedule driving lessons, and track their learning progress.
* Customers should be able to book lessons online or by phone, choose from training packages, and view their test scores and lesson history.
* The system should also help staff manage reservations, track driver assignments, and access reports.
* It needs to work on both computers and mobile devices and store everything in the cloud.

### 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 wants to improve DMV pass rates by offering better training, including online tests and in-person driving lessons.
* The problem they’re solving is poor training that causes people to fail their DMV tests.
* Customers should be able to choose from 3 packages (6, 8, or 12 hours of driving + lessons + online content).

The system needs:

* User accounts with login and password reset options
* Lesson scheduling with date/time, driver, and vehicle
* Activity tracking (who made or changed a booking)
* Reports that can be downloaded and viewed in Excel
* A way to collect customer details, payment info, and pickup/drop-off locations
* A way to update content when DMV rules or tests change
* Different access levels for the owner, IT admin, secretary, and customers

### 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?*

Let customers:

* Create an account and reset their password if they forget it
* Choose a training package and make lesson reservations
* View their test scores and progress (status: not taken, in progress, failed, passed)

Let staff:

* Assign customers to drivers, cars, and lesson times
* See a table with lesson time, driver, and any notes or comments
* Track who made or changed bookings and run printable activity reports
* Disable certain training packages when needed

The system should:

* Send notifications when DMV materials are updated
* Be secure and allow different access levels for staff and users
* Be cloud-based with minimal technical maintenance needed

## 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 run in a web-based environment and be hosted in the cloud.
* It needs to load quickly on both desktop and mobile devices (within 2–3 seconds).
* The system should be available 24/7 with minimal downtime.
* Updates to the system should happen automatically and not interrupt users.
* It should be able to handle multiple users at the same time without slowing down.

#### 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 work on any major platform that supports a web browser (Windows, macOS, Android, iOS).
* The backend should include a secure database to store user info, lesson data, test results, and activity logs.
* A web server is needed to run the site, and there should be tools for handling data backups and security.
* The front end should be built using common web technologies (like HTML, CSS, JavaScript).

#### 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?*

* Users will be identified by their username or email, and by their assigned role (customer, secretary, IT, owner).
* The system should treat usernames and passwords as case-sensitive for better security.
* If someone enters the wrong login multiple times or tries to access something they shouldn’t, the system should alert the IT admin.
* Errors or system issues should also be logged and viewable by staff.

#### 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?*

* Staff with the right permissions (like the IT admin) should be able to add, remove, or update users through the system, no code needed.
* The system should still work after browser or platform updates as long as standard web technologies are supported.
* The IT admin should be able to reset passwords, block user accounts, and manage access levels for different staff members.

#### 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?*

* All users must log in with a username and password.
* The system should use secure HTTPS connections to protect data.
* If someone tries too many incorrect logins in a row, the account should be temporarily locked, and the IT admin should be notified.
* Users should be able to reset their password using an email verification process if they forget it.
* All data sent between users and the server, including personal and payment information, should be encrypted to prevent unauthorized access.

### 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 users to schedule, cancel, or modify driving lesson appointments.
* The system shall validate user credentials when logging in and allow password recovery.
* The system shall track and display progress on practice tests, including score and completion status.
* The system shall log all user activity related to appointments, including who created, changed, or canceled them.

### 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 should be web-based and work on both computers and mobile devices.
* Customers will use it to book lessons, choose packages, and check their test progress.
* Staff (like the secretary and IT admin) will use it to schedule lessons, manage accounts, and track activity.
* Each user should only see the parts of the system that match their role.
* The layout should be easy to use, with clear forms and tables for entering and viewing information.

### 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?*

* Some future features like customizing training packages or adding modules were not fully addressed and may be added in a future update.
* Payment processing methods and how credit card info is stored or verified were not fully covered in the design.
* All users will have internet access and a device with a modern web browser.
* Users are expected to know how to use a basic website to schedule lessons and view progress.
* Customers will provide accurate personal and payment information during registration.
* IT staff will be responsible for managing user accounts and handling security issues.
* DMV updates will be shared in a format that can be processed by the system and shown to staff.

### 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 system won't support offline data entry or updates to avoid duplicate records or data conflicts.
* The design does not include built-in tools for customizing or adding new training packages without help from a developer.
* Credit card payment processing has not been fully planned out and may need to be handled by a third-party service.
* The team may be limited by time and budget, so some advanced features (like real-time DMV sync or mobile app versions) might need to wait for future updates.
* Staff will need to rely on their internet connection and compatible devices to use the system properly.

### 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 screenshot of a project management

AI-generated content may be incorrect.