# 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 purpose of this project is to design a web-based system for DriverPass, a company that aims to help students better prepare for their driving tests. The client, DriverPass, is represented by Liam, the owner, and Ian, the IT officer. They want a system that allows customers to take online classes, access practice exams, and schedule on-the-road driving lessons. The system should be accessible online from any device, support different user roles (like admin, IT staff, secretary, and customers), and allow secure login and reporting features.

### 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 the system to provide a better training experience for customers who are preparing for the DMV driving exam. The main issue they want to solve is the high failure rate among students who only rely on past written tests. The system should allow for scheduling driving lessons, accessing study materials, tracking student progress, assigning instructors and vehicles, and managing user accounts. Key components will include an online reservation system, course content and practice exams, user account management, test tracking, driver notes, and activity reporting.

### 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?*

* Once completed, the system should allow customers to register for driver training packages, schedule and manage their lessons, and access practice materials. The system should track test results and instructor feedback and display the status of online exams as “not taken,” “in progress,” “failed,” or “passed.” It must allow administrative users to print reports, reset accounts, block access, and manage different user roles. These tasks should be easily measurable by successful user logins, lesson bookings, completed tests, and system-generated reports.

## 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 needs to be web-based and hosted in the cloud so that it can be accessed on desktops and mobile devices from any location. It should perform basic operations such as login, scheduling, and test access in less than two seconds. The system should allow for regular updates, particularly when new DMV rules or practice tests are released. It should be designed for high availability and low downtime to support business continuity.

#### 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 must be compatible with all major platforms, including Windows, macOS, iOS, and Android. It should work on any modern browser and not be tied to a specific operating system. The backend will need a secure relational database to store user profiles, test data, appointment history, and package selections. Cloud-based infrastructure will support scalability and performance.

#### 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 unique login credentials assigned during registration. Inputs like usernames and passwords should be case-sensitive to enhance security. Admins should be alerted if there are multiple failed login attempts, scheduling conflicts, or system errors. The system should also clearly log and report who made, changed, or canceled appointments for accountability.

#### 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 system should include a flexible administrative dashboard where IT and admin staff can add, deactivate, or update user accounts without needing to access or change the source code. The system must use responsive design so it can adapt to browser and platform updates. IT administrators should have full access to system settings, user management, and logs, including the ability to reset accounts and disable training packages if needed.

#### 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?*

* Each user will need a username and password to log in securely. The connection between the client and server will be protected using HTTPS encryption, and sensitive data will be encrypted in storage. In the case of a brute-force login attempt, the system should lock the account temporarily and notify the IT admin. A password reset option should be available to users through a secure email link or verification process.

### 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 users to register, select training packages, and schedule lessons. The system shall provide access to online course material and DMV practice tests. The system shall track lesson times, driving instructor notes, and test scores. The system shall allow secretaries to input and manage student records and appointments. The system shall allow admins to generate activity reports, manage user accounts, and reset passwords. The system shall send alerts when DMV materials are updated, and the system shall allow students to reset their passwords securely.

### 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 must be clean, intuitive, and responsive for use across different screen sizes and devices, including desktop and mobile browsers. The primary users include customers (students), the company owner, IT personnel, and secretaries. Customers should be able to schedule lessons, access practice tests, and track their progress through the interface. Secretaries need access to create or update appointments, input student information, and contact customers. IT and admin users will use the interface to manage users, troubleshoot system issues, and generate reports.

### 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?*

* It is assumed that all users will have access to the internet and a device with a modern browser. It is also assumed that users will have basic knowledge of how to navigate online forms and scheduling systems. The design assumes that the DMV will supply updated materials in a usable format. It is also assumed that customer support will be handled outside of the system, such as by phone or email.

### 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 will not allow nontechnical users to create or customize new training packages independently; this will require developer support. Integration with DMV systems for automatic updates may not be possible and could require manual updates. Budget constraints may prevent advanced features like mobile apps or real-time chat in the initial version. Time constraints may also limit testing or performance optimization before launch.

### 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.*

[Insert chart]