# 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 system is for DriverPass, aiming to improve driving test preparation with online practice exams and on the road training.

### 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 a system that enables users to book driving lessons, take online tests, and track progress. The goal is to improve pass rates for driving tests.

### 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 should allow students to schedule lessons, take practice exams, track progress, and print reports. Different users (admin, students, etc.) will need different access levels.

## 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?*

* [Insert text]

#### 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 will be a web-based application and must run efficiently on both desktop and mobile browsers.
* Response time for user interactions, such as scheduling a lesson or accessing test results, should not exceed 3 seconds.
* The back end will require a cloud-based database (such as MySQL or PostgreSQL) to store user data, test results, and lesson schedules.
* A secure API is needed for integration with external systems, like the DMV, to receive updates on driving test requirements and regulations.

#### 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 will distinguish between different users by assigning unique roles such as Student, Instructor, Admin, and IT, each with specific access levels and permissions.
* input for login credentials will be case sensitive, especially for passwords to ensure security, but usernames will not be case sensitive.
* The system will inform the admin of issues such as multiple failed login attempts, suspicious activity (e.g., unusual login locations), or any failed transactions or system errors. Additionally, any changes made to user accounts or critical system settings will trigger an alert to the admin.

#### 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 will allow IT administrators to add, remove, or modify users through a dedicated admin interface without requiring any code changes. This will include options to manage user roles, permissions, and account statuses.
* The system will be designed to automatically adapt to platform updates, such as operating system or browser upgrades, ensuring continued functionality without manual intervention.
* IT admins will need full access to manage user accounts, reset passwords, handle security settings, and oversee system updates. They will also have the ability to monitor system performance and generate reports on user activity and system health.

#### 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?*

* Users will be required to log in using a unique username and a strong, case-sensitive password that meets security standards. Two-factor authentication can also be implemented for added security.
* Data exchanged between the client and server will be encrypted using SSL/TLS protocols to ensure secure communication and prevent data breaches during transmission.
* If a brute force hacking attempt is detected the system will temporarily lock the account after three unsuccessful tries and notify the admin. The user will need to verify their identity via email or phone to unlock the account.
* If a user forgets their password, a secure password reset option will be available. The system will send a password reset link via email, which will expire after a set time 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 validate user credentials when logging in to ensure secure access.
* The system shall allow students to schedule, modify, and cancel driving lessons online.
* The system shall track and display student progress for online practice exams, including scores and completion status.
* The system shall enable administrators to manage user accounts, including adding, modifying, or deactivating accounts.
* The system shall allow instructors to view scheduled lessons and record feedback or notes for each student session.
* The system shall generate and display reports on lesson bookings, user activity, and test results for administrators and instructors.

### 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 needs to be user-friendly, intuitive, and accessible across various devices, including desktops, tablets, and smartphones, through both web and mobile browsers.
* Different users include students, instructors, admins, and IT personnel.
* **Students** should be able to register, schedule lessons, take practice exams, view progress, and reset passwords. **Instructors** need to access their schedules, view student information, and provide feedback on completed lessons. **Admins** require the ability to manage user accounts, monitor system activity, and generate reports. **IT personnel** will need access to system security settings, user management, and system monitoring tools.
* Users will interact with the interface primarily through web browsers on their desktop or mobile devices, with responsive design to ensure seamless functionality across all platforms.

### 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 a stable internet connection and modern web browsers to interact with the system.
* The design assumes that users, especially students, have basic technical literacy to navigate the web interface and perform tasks like scheduling lessons or taking practice exams.
* It is assumed that the system will integrate smoothly with the DMV’s database or external resources for driving test updates and requirements.
* There is an assumption that any third-party services (payment processing or email notifications) are reliable and can be securely integrated into the system without issues.

### 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?*

* One limitation in the system design is its dependence on internet connectivity; if a user has poor or no internet access, they will be unable to use the system effectively.
* The system may be limited by the budget, which could restrict the implementation of advanced features like AI-driven feedback for students or high-end security measures such as biometric authentication.
* Time constraints could limit the scope of development, possibly delaying the release of future features like customizable lesson packages or more complex reporting tools.
* The system may also face limitations in scaling if the number of users grows significantly, requiring additional investment in infrastructure and server capacity.

### 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.*

