# 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 develop a system for DriverPass, a company focused on providing drivers training services. The client, Liam, aims to address the need for better drivers training by offering online classes, practice tests, and on-the-road training options to customers.

### 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 facilitate online drivers training, including class registration, scheduling appointments, tracking driving lessons, managing user accounts, and providing access to DMV-related materials. The components needed for this system include user management, appointment scheduling, lesson tracking, package management, payment processing, DMV integration, and smart user interface.

### 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 customers to register for driver training courses, schedule driving lessons, access study materials, and track their progress.
* Should provide a seamless user experience, appointment management, accurate lesson tracking, and secure payment processing.
* The overall goal is to improve the success rate of customers in passing their driving tests and enhance overall satisfaction with the training experience.

## 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, ensuring accessibility from any device with an internet connection. Response times for user interactions should be within a reasonable range to provide a smooth user experience.
* System updates should occur regularly to make sure the latest DMV rules and regulations are captured.

#### 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 be platform-independent, accessible from any operating system or device with a compatible web browser.
* Backend requirements include a database system to store user information, appointment data, lesson schedules, and payment details.

#### 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?*

* User identification should be accurate, noting the difference between different users based on unique credentials. The system should notify the admin instantly of any errors or discrepancies, ensuring timely resolution of issues.

#### 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 allow for user management operations (add/remove/modify) without any other changes to the underlying code. It should adapt seamlessly to any platform updates, and maintaining compatibility with technologies and standards.
* The IT admin requires full access to the system to perform maintenance tasks, manage user accounts, and address any technical issues.

#### 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?*

* User login should require authentication through secure credentials (username/password) to prevent unauthorized access.
* Data exchange between the client and server should be encrypted using industry-standard security protocols.
* In the event of a "brute force" hacking attempt, the system should have account lockout mechanisms to prevent further access.
* Password reset functionality should be available to users who forget their passwords, ensuring they can regain access to their accounts securely.

### 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 customers to register for driver training courses by providing necessary personal information.
* The system shall allow users to schedule driving lessons based on available time slots and instructor availability.
* The system shall track lesson progress, including lesson times, instructor comments, and completed tasks.
* The system shall integrate with DMV systems to provide up-to-date materials and practice tests.
* The system shall process payments securely for course registration and other services.

### 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 cater to different users, including admin, instructors, and students. Admins should be able to manage user accounts, view appointment schedules, and access system settings. Instructors should have access to their schedules, student information, and lesson details. Students should be able to register for courses, schedule lessons, access study materials, and track their progress. The interface should be accessible through both web browsers and mobile devices, ensuring flexibility and convenience for users.

### 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 users have access to an internet connection and web browsers or mobile devices. The system assumes that users will provide accurate and up-to-date information during registration and appointment scheduling. It is also assumed that the IT admin has technical knowledge to manage and maintain the system effectively.

### 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?*

* Limitations include resource constraints such as time, budget, and knowing technology. The system design may face limitations in terms of scalability and performance under high user loads. Integration with different systems, such as DMV databases, may pose challenges in terms of data consistency and connection.