# 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 the DriverPass project is to address the deficiency in driver training by providing comprehensive online practice exams and on-the-road training to better prepare students for their driving tests. The client, DriverPass, wants a system that facilitates online access to practice exams, scheduling of driving lessons, tracking of reservations and appointments, managing user accounts and permissions, ensuring security and compliance with DMV regulations, and providing a user-friendly interface accessible via web and mobile devices.

### 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 aims to revolutionize driver training by offering a comprehensive system that caters to the needs of students preparing for their driving tests. The system will consist of various components, including but not limited to online practice exam modules, appointment scheduling functionalities, user account management tools, compliance monitoring mechanisms, and a user interface designed to facilitate easy interaction and navigation.

### 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?*

* Develop a user-friendly system that allows students to access online practice exams.
* Implement a robust scheduling mechanism to facilitate the booking of driving lessons.
* Ensure compliance with DMV regulations by regularly updating exam materials and rules.
* Provide a secure platform for user data and transaction processing.
* Deliver a scalable system that can accommodate future enhancements and modifications.

## 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 efficiently in both web-based and mobile environments.
* Response times for accessing practice exams and scheduling appointments should be minimal.
* Updates to exam materials and system features should occur regularly to maintain relevance and accuracy.

#### 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 compatible with various platforms, including Windows, macOS, iOS, and Android.
* Backend support for the system may require a database for storing user data and exam information.

#### 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 authentication should be accurate and reliable to prevent unauthorized access.
* The system should provide real-time notifications to administrators in case of any anomalies or errors.
* Input validation mechanisms should ensure data accuracy and consistency.

#### 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 easy modification of user roles and permissions without requiring code changes.
* Updates to platform technologies should not disrupt system functionality.

#### 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 credentials should be securely stored and encrypted.
* Secure connections and data encryption protocols should be implemented to protect user data during transmission.
* Brute force attack prevention mechanisms should be in place to safeguard against unauthorized access attempts.
* Password recovery mechanisms should be available to users in case they forget their passwords.

### 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 access online practice exams covering DMV rules and policies.
* - The system shall track students' progress and display their exam history, including exam names, time taken, scores, and status (not taken, in progress, failed, or passed).
* - The system shall facilitate scheduling of driving lessons, allowing students to book appointments for two-hour sessions.
* - The system shall support multiple packages for driving lessons, including Package One (six hours with a trainer), Package Two (eight hours with a trainer plus in-person lesson on DMV rules), and Package Three (twelve hours with a trainer, in-person lesson on DMV rules, and access to online class with content and practice tests).
* - The system shall allow customization of lesson packages by the administrator.
* - The system shall enable students to schedule appointments online through their accounts or by contacting the secretary.
* - The system shall match students with available drivers and cars based on their selected packages and preferred appointment times.
* - The system shall store student information, including first name, last name, address, phone number, state, and payment details (credit card number, expiration date, security code).
* - The system shall provide a form for students to input their information during registration.
* - The system shall allow password reset functionality for students who forget their passwords.
* - The system shall notify administrators of any changes to reservations or appointments.
* - The system shall be capable of receiving updates from the DMV regarding rules, policies, and sample questions.
* - The system shall alert administrators of any updates received from the DMV.
* - The system shall have a user-friendly interface accessible via web browsers and mobile devices.
* - The system shall display progress indicators for online tests and driving lessons.
* - The system shall allow administrators to manage user accounts, including adding, modifying, and deleting accounts.
* - The system shall ensure data security by encrypting user data and providing secure login mechanisms.
* - The system shall support secure communication between clients and servers to protect sensitive information.
* - The system shall implement measures to prevent and handle brute force hacking attempts.
* - The system shall include password recovery mechanisms for users who forget their passwords.
* - The system shall provide real-time notifications to administrators in case of system errors or anomalies.
* - The system shall allow for future enhancements and modifications to accommodate changing requirements and regulations.

### 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 intuitive and easy to navigate for both students and administrators.
* Students should be able to access practice exams and schedule appointments seamlessly through the interface.
* Administrators should have access to tools for managing user accounts, scheduling appointments, and monitoring system activity.

### 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?*

* The system assumes that users have access to internet-connected devices capable of running modern web browsers.
* It is assumed that the client will provide necessary resources and support for system development and maintenance.
* The system design assumes compliance with relevant data protection and privacy regulations.

### 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's performance may be affected by factors such as network connectivity and server load.
* Resource constraints may limit the scope and complexity of system features and functionalities.
* Time and budget constraints may impact the speed of system development and deployment.

### 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]