# 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?*

* Purpose is to design a comprehensive online & offline system for DriverPass to help the users prepare for their driving tests.
* Client wants to provide on-road driving lessons, online courses, and DMV practice tests.
* The system should support appointment scheduling, lesson tracking, user management, and DMV compliance updates.

### 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?*

* Permits clients to register for driving lessons and schedule appointments.
* Provides three training packages to select from, with different service offerings.
* Provides access to online course content and exam practice.
* Enables the secretary to book appointments and enter students' information.
* Enables admin and IT personnel to manage users and permissions.
* Records all changes made to reservations or user information.
* Key components needed:
  + Reservation system
  + User roles and permissions
  + Package management module
  + Lesson tracking with driver notes
  + DMV update integration
  + Web-based UI with mobile compatibility

### 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 an account.
* Select a training package.
* Schedule, modify, or cancel appointments.
* Access course materials and take practice exams.

Admins and IT staff should be able to:

* Manage users and system access.
* Track appointment and activity logs.
* Reset passwords and block accounts as 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 be operated as a web-based application that is hosted in the cloud.
* It should be available through desktop and mobile web browsers.
* The system should attempt to load every page in under 3 seconds with normal load conditions.
* When online, the data should sync in real-time.
* The system should be updated biweekly or monthly, depending on the need to maintain security and DMV content alignment.

#### 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 use modern web browsers (Chrome, Firefox, Edge, Safari).
* The backend should be supported on either Linux or Windows servers.
* The application should rely on a cloud-based database (AWS RDS, Azure SQL or similar).
* The application must not use client-side installs or plugins.

#### 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?*

* Each user must have a unique identifier in the form of an email and/or student ID.
* Each input field will validate for the correct input format (e.g., phone, email, credit card).
* Input will not be case sensitive unless passwords or ID's require sensitivity.
* The system must log every user activity (reservation creation, edits and cancellations) with a timestamp and user ID.

#### 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?*

* Administrators users need the ability to create, modify, or delete users through the user interface.
* Packages should be able to toggle enabled/disabled without changing any backend code.
* IT administrators should have access to the roles, accounts, logs, and ability to block or allow users to logon.
* The systems should continue to work after browser updates and commonly used updates to operating systems.

#### 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?*

* Access using a secure email and password
* password should be hashed and saved
* a brute-force detection method locks after 5 failed logins
* a user should be able to request a password reset via verification of an email
* all information transmitted between the client and server must be securely sent using HTTPS - controls are role-based (Admin, IT Officer, Secretary, Student).

### 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 the registration of users by providing personal and payment information.
* The system shall let users to log in securely using their email and password.
* The system shall validate user credentials when they login.
* The system shall allow users to reset their password by email verification.
* The system shall provide students the ability to:
* Schedule, cancel, and modify driving lesson appointments if needed.
* Choose from the three training packages during registration.
* View their scheduled lessons, test progress, and scores.
* The system shall allow secretaries to:
* Schedule, cancel, or modify appointments for their students.
* Enter and edit the student's personal and appointment information.
* The system shall allow administrators to:
* Manage user accounts (create, modify, disable).
* Reset passwords and block accounts if needed.
* Make reports of user activities (reservations made or canceled).
* The system shall allow IT officers to access and manage system configurations and view activity logs and audit trails.
* The system shall track & log all changes to reservations and appointments with the user ID and timestamp.
* The system shall connect with the DMV data source to receive updates on test rules and notify staff of changes.
* The system shall display online test progress with the info like: the test name, time taken, score, and status.
* The system shall store and display driver comments for each lesson session.
* The system shall provide a contact form for users to reach staff and receive replies.

### 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.)?*

**General Needs**

* The system shall be a web-based application that is accessible via desktop and mobile browsers.
* The interface should use a cloud-hosted service, this way, the client doesn't have to maintain the service locally.
* The design should be easy to read and use.
* The users should be able to navigate the interface successfully, no matter their technical ability.

**User Roles and Interface Needs**

**Student/Customer Interface**

The following abilities should be available to students and customers:

* Register for an account and choose a training package.
* View a dashboard that represents: When the next appointment is scheduled.
* How the student is progressing through their tests (test name, time, score, and status).
* Notes from driver: lesson times, start/end hour, comments, etc.
* The user should also have the ability to: Schedule, cancel, or update lessons. Reset a forgotten password. Contact support through a form.

**Secretary Interface**

* The secretary's view should allow them to see and edit customer records.
* The secretary should allow them to schedule, cancel and update appointments for the customer.
* The secretary should enter customer data (name, address, phone number) etc.
* The secretary also shows a summary of the general appointments and usage of the packages.

**Administrator Interface (Liam)**

* Administrator Interface (Liam)
* Liam should be able to see all the data from all users.
* Liam should be able to generate and download reports (e.g. user activity, appointments).
* Liam should have the ability to enable and/or disable training packages.
* Liam should have the ability to see system activity logs.

**IT Officer Interface (Ian)**

* Ian should have the ability to manage the users and the configurations of the system.
* Ian should also be able to reset user accounts.
* Ian should receive alerts for any system errors or security issues.
* Ian should be able to see any connections to the DMV for rule/test updates.
* View and manage customer records.
* Schedule, cancel, and modify appointments for customers.
* Enter customer data (name, address, phone, etc.).
* View a summary of appointments and package usage.

**Interaction Channels**

* The primary channel for interacting with the system is web browser.
* System supports responsive web design or developed in a mobile application (tablet or phone) so participants can access features on devices they commonly use.
* System supports outputting data (e.g., reports) to excel or csv.
* Uses (solutions) can receive notifications and reminders (e.g., daily or weekly appointments, DMV notices) through the interface and optional sending via emails.

### 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?*

* Program users will have internet access and modern browsers.
* DMV will keep timely updates in a format.
* Scalability of the system will be considered for future functionality.
* Packages cannot be modified by staff, only turned on, and off.

### 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?*

* Package customization needs developer assistance
* Budget limited to advanced AI or voice assistant implementation
* Real-time aspects are online
* Minimal capability to edit offline

### 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 white grid with multiple colored boxes

AI-generated content may be incorrect.