# 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?*

* Client is DriverPass: a company designed to help students pass the driver’s test
* The system needs to keep track of reservation information
* The system should connect to the DMV system to provide updated questions about DMV policies

### 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?*

* They need access to the system from anywhere, including offline
* Keep security and access in mind (Ian wants full access and admin rights)
* Track who makes reservations, who cancelled it, last modified, activity report (able to print)
* Customers should be able to make reservations through system (each lesson is 2 hours)
  + entering a date and time
  + call or visit to schedule appointment with secretary
* There are multiple drivers and cars, so each customer needs to match with driver, time, and car
* Users:
  + Liam (Boss)
  + Ian (IT Officer)
  + Secretary (answers phone, makes appointments)
* 10 cars, each car has driver – 3 packages:
  + Package one: six hours in a car with a trainer
  + Package two: eight hours in a car with a trainer and in-person less to explain DMV rules
  + Package three: twelve hours in a car with a trainer, in-person lesson, access to online class
* Each package is spread out over 2 hour lessons
* Packages should be customizable
  + Wants the ability to disable a package at any time (high priority)
* Has “reset password” feature
* Connect with DMV for updated policies
* Pages:
  + Input form to fill personal info
  + Contact us
  + Contact the student

### 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?*

* Users can schedule, change, and cancel appointments
* Users can reset their password
* Users can see updated practice questions about current DMV policies
* Customers should have contact page to talk to DriverPass, and the DriverPass users should have contact page to talk to student
* Liam should have access to disable packages
* Reports to see tracking activity on who changes reservations

## 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?*

* It should be a web-based application that can be run from any browser or mobile device
* All major user operations, like logging in and scheduling appointments, should happen within a couple of seconds
* System uptime shall be 99% or greater during business hours.
* Practice test should be synchronized with the DMV updates at least monthly, it not more

#### 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 support Windows, macOS, Android, and iOS within modern browsers
* A relational database like MySQL should be used to store the users, vehicles, reservations and other data
* The back end should use secure web services like AWS or Azure

#### 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?*

* Different users can be distinguished by email or a username that is unique to each user
* The email would not have to be case sensitive, but the username will be case sensitive as well as the passwords
* The system should inform the admin of security issues or multiple failed transactions

#### 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 admins should be able to add, remove, or modify users through the admin interface without needing to change the code in the backend
* The system will need to be scalable to allow for new vehicles or package types in future updates
* The IT admin needs full administrative access to perform resets, block users, and manage security roles

#### 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?*

* The user shall log in with a unique username and an encrypted password
* All communication between the client and server needs to be secured through HTTPS with SSL/TLS encryption
* Accounts should lock after multiple failed login attempts and require email verification
* Users can reset their passwords by clicking a “Reset Password” button that sends them an email with instructions
* The admins should also be able to manually send out password reset emails

### 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 register and manage their profiles
* The system shall allow customers and receptionists to create, modify, or cancel 2-hour driving lessons
* The system shall assign each lesson to an instructor, time slot, and vehicle
* The system shall automatically track who made, canceled, or updated each reservation
* The system shall allow IT admins to manage all accounts, reset passwords, and block access
* The system shall allow the owner to disable or edit packages
* The system shall retrieve DMV updates and refresh policy and test question data
* The system shall allow for activity reports
* The system shall provide a Contact Us page for customers and a separate page for the staff to contact the students
* The system shall display a progress tracker for online tests

### 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.)?*

* Users:
  + Students: Access lessons, view schedules, take practice tests, and track progress
  + Secretary: create, edit, and cancel reservations on behalf of students
  + Instructors: Add notes and feedback for completed lessons
  + IT Admin: Maintain accounts, permissions, and system security
  + Owner: access reports, disable packages, and view analytics
* UI Needs:
  + Web-based design optimized for desktop and mobile users
  + Dashboard showing key metrics for upcoming lessons, test progress, and driver notes
  + Navigation menu with pages for scheduling, tests, reports, and contact forms
  + Form validation with error handling and feedback

### 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?*

* DMV provides API access for policies and updates
* All users have reliable internet connections
* Secretaries and instructors are trained to use the new interface
* Payment data is processed securely through third-party integration

### 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?*

* Offline access will significantly hinder access to most actions
* Future customization will require more iterations from developers
* Integration with the DMV system relies on its uptime and how it formats the data
* Budget and time limit the initial rollout
* Heavy usage may slow performance until the system can be properly scaled

### 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.*

