# CS 255 Business Requirements Document Template

*Andrei Kourouchin*

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?*

* DriverPass is the client – They would like to have a system that is able to manage information regarding their business, and access that information offline. The data should only be editable online. The system should allow users to purchase driving lesson packages and let users take online tests.

### 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?*

* The problem they are trying to solve is that too many people are failing driving tests.
* There should be a front-end website that allows employees and users to access various features.
* There should be a database in the backend that allows storing information for customers, drivers, and scheduling driving lessons.

### 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?*

* Handle multiple package types and allow the deactivation of certain ones
* Have a feed for DMV rule changes
* Access data offline and have database access to edit data online for correct user roles
* Have different access levels for drivers, users, and admin
* Allow customers or to reset their own passwords, or let admins do it for them over the phone
* Track changes for edits and ensure audit log is accurate
* Give employees access to make reservations for customers, and allow users to make reservations online

## 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?*

* This system should be web based
* Website should be a modern web app, created based on the template the client provided
* System should be updated as needed to ensure all plugins used are patched.
* Load times should be prioritized, and data rendered as received to make the use as smooth as possible

#### 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 platform should run on any modern web browser (Chrome, Firefox, Safari, Chromium), and as such should be able to be run on all types of systems including windows, Unix, and MacOSx.
* UI should scale and correctly render for mobile browsers.
* The backend should have access to a database for data storage, but the database should not have direct access to the front end.

#### 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?*

* Users should have a string related to each user object that lets the server know what type of user they are. EG. Admin, User, Employee, IT
* The input should not be case sensitive unless the input is a password, but we should ensure that we properly handle all data that we process to ensure that we do not allow code injection or other harmful data to affect our systems.
* The system should immediately inform an admin of any serious exception, especially those that cause downtime or allow for code execution that is not typical. Regular errors should be logged and no notification made.

#### 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?*

* We should ensure that we can make updates without damaging the web application, building parts of the software in chunks that we can replace without affecting large parts of the system.
* User changes will be done in the front end tracked in the audit log. If unauthorized changes are made, the IT team should be informed.
* IT admin needs access to the admin section of the application as well as the database and webserver.

#### 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 should be required to use either a password that has complexity requirements or using things like Okta for single sign on
* Optional multifactor authentication using an authenticator app or email. SMS is not a secure way of doing 2FA.
* Accounts should be locked after a few incorrect password attempts, however, should be able to be unlocked by a phone call to the DriverPass to unlock the account after the account is verified with staff.
* Forgotten password should not email a password, it should instead email a link to recreate a password. We should never store passwords in plain text.

### 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.
* The system shall ensure the login area is not unusual, EG, user in NYC, but login is from Bangkok. This should prompt a verification.
* The system should lock accounts after a few failed logins
* The system shall allow users to update their profiles, as well as admins to update user profiles.
* The system shall track all changes made to ensure audit log is complete.
* The system shall keep the DMV update page up to date.
* The system shall allow scheduling of user appointments
* The system shall email temporary password reset links when password is reset.

### 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 will interact with the front end using a browser either on mobile or desktop.
* Users will be able to see their own account, as well as make purchases and see order history.
* The interface will allow admins to disable certain package purchases if too many people purchase them.
* Admins will have access to see all users and edit their profiles. They will be allowed to create appointments on behalf of users.
* The interface will be web based and will be lightweight to be able to be loaded on mobile internet easily.

### 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?*

* I am assuming most customers have an email that they have access to
* I am assuming that users have access to mainstream desktop and mobile browsers.
* I am assuming that users will act in predictable ways and as such am adding certain security features that could get in the way of users that fly a lot.
* I am assuming that the DMV releases information regarding rule changes in some sort of consistent way that we can scrape.

### 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?*

* As this is meant to work on mobile as well, we should strive to ensure the front end is light to be able to be used on the go.
* We have 15 weeks to finish this project
* We cannot handle users in off hours when there is no support. As such, we should have automated features such as password reset through email so that we do not have to always have phone operators.

### 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.*

