# 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 client is “big boss” Liam. He wants to develop a system to help train people and help them practice for their driving tests; he wants to be able to schedule and provide training to the customers / users remotely, offline or online, and to track the reservations, appointments, and information pertaining to the users of the service.

### 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 software needs to be able to track user information, such as what their identification is, who they are practicing with, what they’re studying / what tests they are provided, as well as basic information like when their appointments are, when they make reservations, modifications, cancellations, or otherwise interact with the service. They want to be able to view and read all of this data from the service, and to have full control over modifying user accounts, in the event that access is lost or needs to be removed.

### 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?*

In the end, this service is meant to be an accessible application for training and helping teach driving students how to pass their tests, and get some practice behind the wheel / on the road with qualified instructors.

**\*The system should allow the customer to choose the service.**

**\*The system should allow them to take their tests online, practice or official, and allow them to see their statistics (ie grade, time, attempts).**

**\*The system should also allow moderators / instructors to view and give feedback.**

**\*Moderative tools for the system, up to date tests and answers, and readily available study and practice for customers / students.**

## 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 needs to be readily available to run on multiple platforms, and through multiple different services, eg chrome on windows OS, or safari on mobile, on linux, mac, etc.
* It should receive regular updates to the material, as laws are always changing and updating, and so to are the requirements for test takers.
* The system, while it would be optimal to run fast, and convenient for the users taking the tests, ultimately does not have a huge concern for performance. Stability will be paramount but displaying questions and answers, links to websites and study resources is not exactly a memory or performance heavy requirement to reach. If and when there are timed tests though, there needs to be some level of speed expected.

#### 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 able to run on pretty much any device with access to a web browser.
* Windows
* Mac
* Linux
* Android
* Apple
* The back end needs a database to store and supply the test information and history, but that is about all it needs to do.

#### 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?*

* The system distinguishes between users via the information that they provide upon creating their account. An email address / password combination would be more than sufficient. Providing a user name or user id would remove one layer of abstraction from this and have a unique identifier for each customers.
* The system would probably only need to raise alarm if there is a breach or hack of some sort. Eg repeated attempts on a user’s account, or if the user requests a password reset. Then again, all this could mostly be automated and handled without admin input.
* The other case where an admin is necessary would be in cases where the customers might be looking to change instructors.

#### 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 needs to be fairly robust. It will need to have a continuously updating pool of test, study, instructor, and student information. The actual structure of the website ought to remain fairly rigid however. Simple is sweet.
* It will need to have a strong back end that can update along side multiple different platforms seamlessly.
* It actually makes a fair bit of sense to limit the administrative access, with the exceptions of them handling feedback and complaints / requests for instructor changes. An admin might only need access to the user identifiers, names, progress, and other basic information in order to do their job, and it doesn’t make a huge difference for them to have any more.

#### 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?*

* A user needs to have a password and username.
* SHA-2 series encryption would be a very good thing to have for the service. This would make it very very very difficult for a breach to occur on a service like this one.
* If there is a brute force hacking attempt, it would make sense to lock the user out or require them to provide an alternative source of verification, such as an ID or phone number verification.
* If a user forgets their password, then the aforementioned alternative verification source would be used and verified by the administrators to reset their password.

### 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.”*

* (Supposing the user is already signed up), The system will prompt users for their username and password, offering reset options if needed.
* The system will take input from the consumers, and validate their information; in the event that they are not already signed up they will be prompted to create an account.
* The system will validate the user’s input and redirect them to the relevant page depending on their status as an instructor, admin, or student.

### 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.)?*

* There are three users that interact with the interface. Students, teachers, and administrators.
* All should be able to see the login page, their account information, their notes / feedback (instructor or user), and grades / exams if they have them
* Instructors should be able to see their student’s relevant information, the notes and feedback they have left, and all other relevant instructor options
* Admins should be able to see everything, as well as the options to update and change user data.

### 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 students will all actually work
* There will be updates and changes to the law
* The course work will be easy to obtain and accurate
* The system will not have outages
* Each user will have strong internet access

### 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 is built around driving tests. This naturally means the DMV is involved in it, which naturally means everything is going to happen really slowly. The system being designed around it might make certain aspects of the study occur slowly, perhaps slower than the real world rules are updated.
* The system can’t always be up and active, at least it’s unlikely.
* Some students might drop in and out, and these sorts of scenarios are difficult to deal with without significant administrative intervention.

### 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.*

