# CS 255 Business Requirements Document Template

## 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: DriverPass
* DriverPass wants a system that can keep track of client data and allow for scheduling appointments and booking of available packages.
* The purpose of this project is to create a system for scheduling appointments, tracking progress, and tracking the individual clients.

### 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 has noticed that many people fail their driving tests and wish to create a platform to enable people to learn and reduce the number of failures.
* The system they wish to use needs to allow for users to schedule appointments for driving tests, keep track of any notes they are given for improvement, and also any practice exams they take through DriverPass.
* They want this system to be accessible by the users who sign up for the service, but also mildly accessible by different areas of the company to ensure that the correct notes and scheduling are assigned to the correct user. EX: the secretary taking a phone call and scheduling an appointment for driving practice.
* DriverPass also wants the system to stay up-to-date with current DMV rules, regulations, and standards.

### 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 main objective is, when the system is complete, to keep track of each user’s individual progress, purchases, and information.
* It needs to allow users to make purchases from the system, schedule their appointments and even take the tests.
* There needs to be a way and security levels that allow for someone, such as the secretary, from within the company to schedule appointments for clients that call to do so.
* The interface needs to follow the example that the owner provided for what the user sees.
* They need a database to allow for keeping track of users, their progress, and what packages they have purchased.
* The system needs to allow for updates from the drivers in order to add notes to the users they are working with.
* The system needs to send out notifications when the DMV for some locality has updates to their rules or their practice questions.
* Needs to keep track of the locality of the individual users and supply the practice exams catered to that region.
* Supply drivers with information on pickup and drop off locations of the users that have scheduled a driving session.
* In the future there should be a way for packages to be added, removed, and updated.
* Needs to be able to allow the owner to disable a package that is no longer available.
* Needs a way for the IT department to reset user passwords when they are forgotten.
* Allow the users to cancel, modify, or schedule their appointments.
* Has a way to download reports, accessible only to the owner.
* Data is accessible both online and offline, but only allowed to make changes when online.
* Provides information on who cancels or modifies a reservation and print the activity when something goes wrong.
* Connects the correct user information with the correct user account.
* Runs over the cloud.

## 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?*

* Should be web-based, such as being a website.
* Should include cloud access to ensure access everywhere.
* The system needs to run fast enough for user access to access the site without certain amounts of lag time to ensure ease of access.
* System should be updated as often as the client decides to add and implement a new feature.

#### 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?*

* Linux should be the platform of choice for this system.
* The cloud, an example of which is Amazon Web Services, should be employed for managing security and database requirements for the back 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?*

* Each, individual, user needs to create a unique username and password. This is when first accessing the site and signing up on it. These username and passwords would be used for each attempt to access hereafter.
* Input on the password should be case-sensitive. It is rare to have usernames be case sensitive, but passwords need to be to ensure that the correct input is received.
* Additionally, access to the system should employ multi-factor authentication. An example of which is sending a one-time use code to a user’s phone number via text message to verify the person’s identity and increase security.
* The administrator needs to be notified immediately when there are any problems with the system/website to ensure that it can be fixed as quickly as possible.

#### 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 adapt to changes to the platform, which could be simple updates to the code all the way to changes in the DMV makes for driving test and rules of the road. This can be an immediate change or slow changes if there are many of them to be updated at once.
* The system has to allow for modification around the users. This mean that if the users are added, removed, or modified in any way, the system is capable of handling this without having to go in and change the code entirely.
* IT administrators will also need to have full access to the system as they will be the front line to any changes. Such changes can include updates to the system, employees leaving or joining, and users needing password changes and the like.

#### 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?*

* Users will have to have usernames and passwords to get access. It is also a good plan to have multi-factor authentication along with these two elements.
* All data exchanged between the client and the server should be a responsibility of the cloud for both online and offline access of certain features.
* The system needs to disable logins when a certain number of, determined by client, incorrect login information is entered. Standards are anywhere between 4 and 10 times.
* There needs to be a prompt, generally email, for users that forget their passwords. This email would provide a link that would reset their address. One can also use a temporary, one-time user, password provided by email. This would have them taken to a password reset page after they use the password to login.

### 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 the login credentials of users when they log in.
* The system shall be able to allow the users to book reservations for driving practice.
* The system shall provide practice tests to the users.
* The system shall inform the driver which customer they are paired with and where to pick up and drop off.
* The system shall offer 3 different packages for purchasing driving lessons.
* The system shall provide custom access to the users based on their privileges.
* The system shall show the user what tests and work they have completed.
* The system shall allow the secretary to book appointments to specific users if they call in to do so.
* The system shall provide users with updated DMV rules when there is an update to the system.

### 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 must display user’s driving information: progress of online tests, their personal information, if they have special needs, the photo of them and of the driver, and any notes the driver has.
* The interface needs to allow the DriverPass employees to make changes and update the system.
* The interface should allow the user to interact with it from mobile devices that are capable of internet connection.

### 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?*

* There is no budget listed and we are assuming that everything we’re building will be available in the budget.
* We are assuming the timeframe we are using is based entirely off of our own knowledge instead of information provided by the customer on when they would like the project completed.
* We are assuming that we will have easy access to all of this technology.

### 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?*

* We have approximately 11 weeks to complete this project in.
* We were not given a budge.
* Another limitation is that we need more people to build the site to finish it in the timeframe.
* The best approach to this project is an agile approach.

### Gantt Chart

