# 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?*

* DriverPass is the client
* Provide online classes, online practice tests and in-person road training.

### 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?*

* Reduce the rate at which people fail driving tests
* Components of the system will include:
  + web application (front end)
  + API server (backend)
  + authentication and authorization (security)
  + databases (data retention)
  + reporting component
  + scheduling component
  + tracking component
  + learning component

### 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?*

* Allow customers and employees to login
* Secure system by restricting access on the system based on employee role or package purchased.
* Customers should be able to modify existing appointments online (cancel, modify and create)
* Customers should be able to reset password online through self service
* Retrieve policies, rules and/or sample questions from state DMV
* Learning system to provide courses and practice tests to authorized customers.
* Reporting system so client can run reports on the various aspects of the system.
* Client should be able to track activity on appointments

## 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 will be a web application hosted on a cloud platform.
* An API server hosted on a cloud platform should serve data to the web application
* Resources should be allocated so performance is optimal providing customers and the client with a seamless experience.
* System will be updated by using sandboxing to ensure the changes do not introduce bugs or vulnerabilities.

#### 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?*

* Most cloud platforms run on a Linux systems with Kubernetes containers
* Multiple databases will contain all the data accessed by the client and their customers
* OAuth 2.0 authentication

#### 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 account will be distinguished by a username which is unique to each account
* Usernames are not case-sensitive as they will be converted to lower case when the system is queried for an existing account
* Passwords are case sensitive
* The system should notify admins immediately upon an errors , bugs or access problems.

#### 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?*

* Changes to users should be made without changing the code.
* The application must be tested against platform updates before going live with the updates.
* Any updates to the system or functionality change must be approved by DriverPass
* The IT administrator must have complete access to the system, they should be able to access the cloud platform in which the application and API server lives.

#### 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?*

* MFA should be used to as an additional level of security
* OAuth 2.0 will provide secure authentication if chosen during account creation
* Security questions will be used to reset user passwords and unlock user accounts
* After three failed login attempts user account will be locked, after 30 minutes has elapsed users attempting to login with a locked account will be notified at which time they will verify their username and the system will send a link with a link to unlock account.
* Forgot passwords or password resets will result with an email being sent with a link to reset 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.”*

* The system shall authenticate customers and employees when logging in.
* The system shall present customers with available date and times for in-person road lessons
* The system shall reserve date and time selected by customers
* The system shall provide client with reports
* The system shall display to customers online course materials and practice tests
* The system shall track all activity on appointments
* The system shall restrict access to unauthorized areas of 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 be responsive such that it runs on desktop, tablet or mobile
* Customers should be able to login to the system
* Customers should be able to create, edit or cancel appointments
* Customers should be able to take online classes and practice tests
* The different users are Customers and Employees
* Employees need to have the ability to edit customer accounts and appointments

### 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?*

* A budget was never discussed in the meetings, so the assumption is made the application and databases will be deployed to a cloud platform.
* Notifications of appointments are not mentioned so it is assumed upon scheduling an appointment the system will send email notification with a calendar invite to the customer.

### 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?*

* One very important limitation we have is time, there is roughly five months to create the application
* Probably the biggest limitation is the development team, a well-rounded team will be needed to complete this application and should include a tester, database designer and developer.

### Timeline Description automatically generatedGantt Chart