# 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?*

* Our client, DriverPass, would like a unique system for driving lessons to make it easier to pass the driving test.
* The system should be able to process the purchase of different driving lessons, along with a user interface for students to take tests and schedule appointments.

### 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 wants to fix the problem of driving students failing the driving test due to not being ready. They want to prepare the students for the test to the best of their ability.
* The system should have a database to store user information and scheduling, along with the availability of driving instructors.
* With the database, a user interface should be created to give the user access to their account information and lesson packages. In the interface, the user will be able to change their settings, schedule driving lessons, take tests online, and communicate with the instructor.

### 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 system should allow both students and instructors to schedule and manage driving lessons, offer a choice of driving packages, and offer online tests for students.
* Users will also be able to reset their password if their account has been compromised or they forget it. On the employee side, the system should allow for downloadable reports to be edited offline and keep track of employee edits.
* The system should also receive DMV notifications when regulations are being changed.

## 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 web based and available on all current web browsers and applications.
* It will have to be fast to allow users to see their updates in real time. A responsive system will ensure no appointments end up double booked.
* Updates should occur regularly with notice given to users in advance. If users are unaware of updates, they may think something is wrong with their account when trying to log in.

#### 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 run on Windows and Mac primarily, with the possibility of a Linux port if there is high demand.
* Running the system in the cloud would save on storage and improve speed.
* The backend should have a database to store all user information, along with test results, driver schedules, and appointment information.

#### 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 users will be differentiated with temporary cookies to ensure only one instance of a user account is active.
* Account names do not need to be case sensitive, but passwords do. Having multiple accounts with the same name aside from the case of the first letter could get confusing.
* An error report will be sent to administrative users daily to fix any issues in the system.

#### 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 student accounts can be made through staff and administrators, but only administrators can add, remove, or modify staff accounts.
* The system will need to keep all data in a database, or databases, for updates to pass.
* An IT admin would need full access to the system and user profiles. To help with troubleshooting and bug fixes, all access is needed.

#### 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?*

* All users need their unique account names and passwords to log in. Passwords should be case sensitive.
* The option for two factor authentication could also be provided if a user wants to be extra cautious.
* Blocking brute force hacking can be easy if there is a limit to the amount of password attempts before the account locks.
* The number of attempts should be low enough to stop hackers, but not so low as to lock the account of anyone who forgot their password.
* If the password is forgotten, a user can click the “Forgot Password” option on the login page, and a link to reset the password will be sent to the email address linked to the account.

### 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 allow users to make changes to their accounts.
* The system shall allow users to schedule and change appointments.
* The system shall offer tests to students for practice between appointments.
* The system shall keep all user data secured.

### 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 should be adaptive based on the device a user is on. A mobile device will have a narrower screen, but a desktop will show a wider display.
* The different users are admin, staff, and student.
  + Admins have full access to all accounts. Staff have access to their own accounts and the accounts of their students. Students have access only to their own accounts.

### 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 assume all students will have access to a device and email account. With that, I can assume a user is knowledgeable enough to be able to use a web browser and access the internet.

### 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?*

* Our main limitation is time. We have 15 weeks to complete the system for DriverPass. We are also limited to what our team knows. If something comes up that no one has done before, it could take some time to get around it.

### 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.*

A screenshot of a computer

Description automatically generated