# 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 DriverPass and they want a system that allows first time drivers and learners to be able to practice for their driving exam with a professional teacher and to be able to match these teachers with customers based around a scheduling system.

### 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 the system to allow customers to schedule a time with a driving instructor to prepare for their driving exam by selecting from study packages that will be available.
* DriverPass seeks to fix the problem of some students not being prepared for their driving exam.
* The different components needed for the system would be a cloud based user interface for the customers to sign in and register, a backend for the administrator to make changes and a database for current, past and future scheduled lessons to be stored.

### 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?*

* To match Students with instructors based on scheduling requirements and to do this automatically.
* Ability For students to make reservations and for the administrator to be able to change it and see when it was last modified.
* Track which instructor is matched with which student in which car.
* Customer should be able to select one of three packages and ability for the administrator to be able to disable packages.
* Customers can also Schedule over the phone.
* When a customer registers a package and schedules a time the system will also ask for a pickup and drop off location.
* The system needs to be connected to the DMV for rules and guideline updates.
* The system should be able to run off the web over the cloud.
* The system should also lay out the test progress to the student through the account interface.

## 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.*

* The user Interface should have the student information, special needs, the instructor photo, the student’s photo, driver notes and online test progress.

#### 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 will be a cloud-based web application.
* The system doesn’t have any specific speed requirements, but it should run reasonably fast to allow students to make instant scheduling changes.
* The system should be able to updated regularly.

#### 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?*

* With it being a cloud-based system it should run on most systems.
* The back end should only require minimal coding tools for the software team to maintain and update the system.
* Due to it being cloud based security will be handled by the cloud provider.
* A database for the storage of students, instructors and scheduling will be required to support this system.

#### 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 Should distinguish user based on user names, and student names.
* I don’t believe the input should be case sensitive.
* The admin should be alerted if there are any login problems, certain packages being bought too much or if there are scheduling problems or cancellations.

#### 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 team should be able to make changes to users like adding, modifying and removing.
* The system being cloud based platform updates should happen automatically.
* The access of the IT admin should be able to modify base code that allows changes of packages and to modify some user profiles.

#### 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?*

* The requirements for user login should be username and password.
* If the user forgets their password they should be able to contact the IT admin team to get a new password.
* Most security should be handled by the platform.

### 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 update the student, instructor, and admin team of scheduled lessons.
* The system shall automatically select instructors based on availability.
* The system shall let student select form three packages and process payments.

### 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 will layout the student lesson progress.
* The interface will layout the student’s scheduled lessons.
* It will also layout the student’s information and payment processing.

### 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?*

* One assumption would be payment processing and payment plans.

### 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 limitation would be that the client’s IT department wouldn’t have access to the full system.
* Another limitation would be the reliance on an external database to pull all information.

### 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.*

*Chart, waterfall chart

Description automatically generated*