# 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 wants the system to do the following:
  + Take advantage of a void in the market to provide training for learning drivers.
  + Provide online education and practice tests for learning drivers to prepare for the written exam at their local Department of Motor Vehicles (DMV).
  + Provide a way for students to schedule in-person and hands on learning to prepare for their driving exam.
  + Provide instructor feedback to students so that they can review and learn.

### 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?*

* Different components that DriverPass is requiring are as follows:
  + Administrative account where administrators can pull reports on all user activities as well as control system permissions for other users. Administrators should also be able to edit which package types are available to students at a given time.
  + Secretarial account so that the user can take and input appointments as well as make changes to appointments for other users who either call in on the phone or walk into the office.
  + Driver accounts so that instructors can provide students with personalized feedback to their students.
  + Student accounts where learners can control their schedule for in-person learning sessions with instructors, in person classes, access practice tests, their personalized feedback, learning materials, and update their personal information and make payments for their classes.

### 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?*

* When it is completed, the system should be able to:
  + Provide self-service scheduling for students.
  + Track progression statuses for students.
  + Track instructors and vehicles.
  + Track user activity for metrics as well as accountability.
  + Providing a central learning environment for students to take online courses.
* These can be considered complete when the following functionality is included in the system:
  + Students can schedule appointments either themselves, or through an individual with secretary permission.
  + The system provides learning materials, resources, and practice tests to students.
  + The system shows students their personalized feedback from instructors.
  + All user activities are appropriately logged into a database from which reports can be pulled.
  + Provide the ability to track instructors’ schedules and appointments as well as which vehicles will be used for 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 should be a cloud-web-based system so that users can access the system more easily than having to download an application that then becomes more difficult to update.
* The system should run as fast as possible, updating the schedules in less than 2 seconds to limit the risk of overbooking.
* The overall system should be backed up and updated at least once a week to avoid potential loss of data.

#### 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 focus on running on Windows and Mac first and can later be updated to run on Linux platforms as these are the most popular platforms, which increases availability for students as well as being able to pull reports to Excel.
* A couple of databases should be used. One that tracks user activity, one that tracks students and their information, one that tracks employees and their information, and one that tracks vehicle information. A calendar built into the system can also assist with maintaining organization.

#### 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?*

* User authentication should be done through case-sensitive unique login and passwords to increase system integrity.
* Administrators should be notified when a user has failed to login 3 times so that they can triage the event as well as aid in a timely manner.

#### 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?*

* Administrators and IT specialists should be able to alter user permissions as necessary without changing code. The web application should need to be manually refreshed for changes to take effect so that progress on a task is not halted at an inconvenient time.
* An IT administrator would require access to the normal administrator permissions as well as back door access to be able to modify code if necessary to complete updates or as immediate technical support.

#### 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?*

* At minimum, a unique case sensitive username and password should be required to log in. The password should have to be updated at least once a year.
* A cloud-based system is required by DriverPass so that the company does not need to deal with the major components of security and data back up and they can focus running the business. User authentication and following the rule of minimal permissions are the biggest pieces of security that needs to be worried about in the system itself.
* The user’s account should lock after 3 unsuccessful log in attempts to help repel brute force attacks, after which an administrator will need to verify the user and unlock the account.
* A user who forgets their password should be able to request to reset it which will send verification through external means (such as email or text) so that the user can be verified and allowed to change 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.”*

* The system shall distinguish between student, secretary, instructor, administrator, and IT administrator accounts and prevent anyone from utilizing functionality that they should not have access to.
* The system shall correctly identify each user by their username and password information.
* The system shall require re-authentication after 2 hours of idleness.
* They system shall produce user activity reports when requested by an administrator or IT administrator.
* The system shall provide students with the option to print receipts after making purchases.

### 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 be through personal computers or mobile devices and needs to be able to handle input and output from where appropriate (keyboards, mice, touchscreens, monitors, speakers, printers).
* The student home page of the website needs to reflect DriverPass’ pre-designed prototype.
* Each account type should have customized options displayed to reflect their role within the company including an option for IT administrators to interact directly with the system code to make updates.

### 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?*

* Users have stable connections to the internet, electricity, and the ability to download data if necessary.
* The user can read and write in English.
* The user has a valid debit or credit card and the means to be able to afford the products being offered.

### 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 dependent on a third-party cloud service provider and the company’s ability to pay for the service.
* The system will become hindered during updates.

### 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