# 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, DriverPass, wants their system to allow their customers to be able to buy and customize different driving lesson packages. With these driving lesson packages, the client wants their customers to be able to make reservations for lesson times based on their package, with the application being able to be access both online and offline. The application also should be able to connect to the DMV for updated policies.

### 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 solve the main problem of kids failing too many times on the driving tests. The different components of the system they want to design for this problem is a reservation system, driver profile system (including notes), as well as a progress tracker to be able to see when a student is ready to take the DMV test.

### 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?*

* Access to system both online and offline
* Handle different packaging types and prices for reservations
* Interface to allow customers to make, cancel, or edit reservations based on package
* Different levels of access between the boss, IT officer, secretary and users
* Receive notifications to new DMV policies
* Database to store consumer information
* Security to protect user data

## 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 application should be able to run in a browser, both on desktop and mobile. The application should be fast enough

#### 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 application should run on Windows, MacOS, Android, and iOS. To support keeping up with the customers and all the different types of packages available, the system would need a database to store this information on.

#### 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?*

* [Insert text]

#### 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 regular Admin(such as owner) should be able to make changes without the need of coding knowledge, while the IT admin should be able to make updates and modifications to the system whenever 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?*

* The user would be prompted for a password whenever attempting to log in. Although not part of the timeline, an optional function suggested could be two-step verification. If an account is found to be compromised, the customer should be notified and an effort should be made to protect as much information from the hacker as possible. If a user forgets their password, their should be able to choose a “forget a password” option on the application or contact an admin who would be able to change it.

### 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 username and password when attempting to log in
* The system shall display different lesson packages to purchase
* The system shall

### 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 user should be able to interact with the interface with both mouse and keyboard, and touchscreen.
* The interface should display user information and lesson packages
* The interface should be able to handle payment options after selecting a package to purchase

### 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?*

* The application should run off of the cloud system
* The application should have a fallback/timeout for people who can’t connect to the application or stay inactivate for too long

### 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 application would need online/web access for full capabilities
* The application would depend on a server to run, meaning any significant updates would possibly take the application offline for the duration of the update

### 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.*

