# 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

This project is to create a website where users can sign up for driver’s training packages, input personal information needed by the client, and view information pertaining to them. The company, DriverPass, wants this website to be a hub for online classes, practice tests, and setting up in person driving training appointments. Additionally, the website needs to be accessible from both personal computers and mobile devices.

### System Background

DriverPass wants this application to give up and coming drivers access to online training and practice tests to improve their ability to pass a driver’s test. For this system to work there needs to be a cloud sourced database and server, client side applications or webpage for access, and an admin application for controlling specific features and inputting data on behalf of users. A way to update the training based on DMV information and roles system for the administration side needs to be implemented. The server and database for this application is to be run on a cloud-based system.

### Objectives and Goals

* Create a database to hold users, tests, drivers, cars, and packages.
* Create a user interface for the client side.
* Create a user interface for the admin side.
* Create a roles system to allow access based on need that is adjustable by the administrator.
* Link database to the interfaces.
* Create a method to download reports.
* Create a method to pull updates from the DMV about testing policies and driving rules.

## 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?*

* System could be web-based.
* System should run smoothly for testing and minimal lag for pulling up information.
* System should be updated whenever a driving slot has been filled, a student has chosen a package, or when regulations update.

#### 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?*

* System should be flexible enough to run on mobile or a computers. All major OSs would make the most sense.
* There needs to be a database to track customers and appointment slots.
* System should run on a cloud system to grow and shrink as needed.

#### 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?*

* Customers should have unique IDs in system possibly by email or username.
* Username should not be case sensitive.
* System should inform administrators of problems should a customer report a bug, a portion of the system becomes inaccessible or a third party software that the system relies on fails.

#### 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?*

* Users should be able to be added, changed, or modified within the constraints of the current system.
* Regular patches should be made as major platform or software updates come through.
* IT admin should have the ability to add, delete, or modify users.
* Admin should be able to cancel and reactivate packages and make changes to the appointments.

#### 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?*

* Login should consist of a username and standard password
* Upon a successful login a secure handshake should be established for the duration of the session via encryption and one time key.
* If there are too many password attempts then the account should be locked until password reset or admin override w/password reset.
* Password reset will require an email and security questions answered.

### 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.”*

* System will validate user credentials when logging in.
* System will access appointment database to display to user or administrator.
* System will check for updates to regulations periodically.
* System will show different package options.
* System will process payments through third party application that connects to appropriate banks and pays into DriverPass account.
* System will display current classes and upcoming appointments for user who is logged in or a list of all for the administrator.
* System will allow for administrator to download list of appointments or users.

### 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.)?*

* User interface will show current class, purchase options, and up coming appointments for current user.
* Admin interface will show list of users, all appointments with user, driver, and times shown.
* User interface will display updates to both users and admins.
* User interface should have ability to adapt based on screen size.

### 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?*

* Card payments will be handled through a third party app.
* Classes will take minimum processing power as basic videos, images, and documents.
* Online will be required to purchase packages, make changes to appointments and to take the classes.

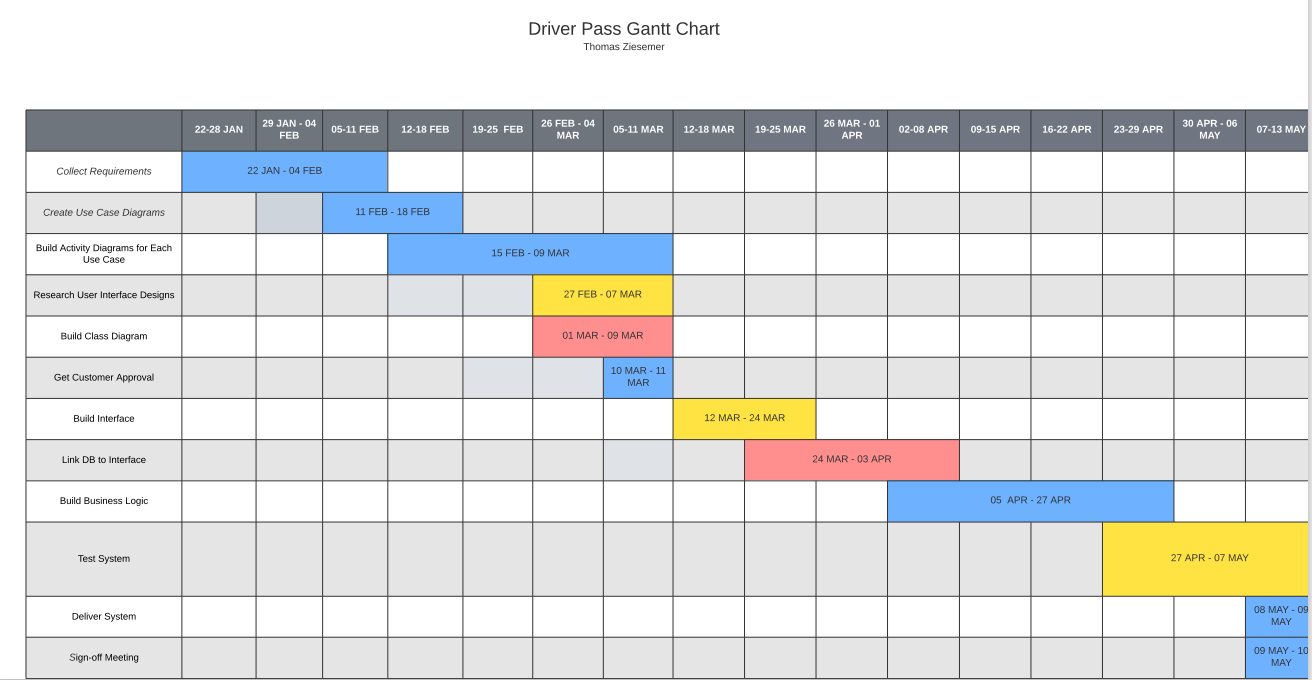
### Limitations

*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?*

* Classes will not be downloadable.
* Only four people working on the project.
* System cannot be heavy on resource as customers of all sorts need access.

### 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.*

**