# 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?*

* Liam is the owner of DriverPass.
* The goal of Driver pass is to help train students to pass their driver’s test.
* Liam fills this is the ample time to achieve this due to the void market and the margin of students who fail at the DMV.

### 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?*

* Liam wants his application to aid students so that they can pass their driver’s test, which currently is an issue for students that go to the DMV.
* Liam would like to see data both online and offline, making data downloadable
* Schedule appointments for his 10 vehicles using different package
* Liam wishes to have roles assigned to users for security purposes
* Liam wants the ability to alter packages
* DMV Compliance.

### 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?*

* Make reservations by phone calls,
* Liam can alter packages
* DMV Compliance (almost like an API)
* View packages
* View data online and offline
* Students should be able to their test progress
* Student information (like name, address, etc.)
* Driver notes
* Driver photo
* Student photo
* Special needs.
* Wireframe should be followed to achieve ample experience while really focusing in on UI/UX

## 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.*

* Scalability
* Capacity
* Reliability
* Maintainability
* Servicabnility

#### 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?*

* I think this application would be good as both a website and an application, if it was written in a speedy front end like React and React Native
* There should be minimum run time, making React much more appealing as it does not refresh as links are clicked on.
* As far as the back end there are numerous back-end servers that can be ran, I recommend the MERN stack as it will keep client and servers separate.
* System updates should run based off the needs of improvement and the diagnostic of bugs.

#### 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?*

* Database to store all data (student info, instructor info, etc.) this can be done using MongoDB
* I recommend that this be ran on windows
* To connect to the database I recommend sticking with the MERN stack

#### 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?*

* There should be some sort of report a bug feature that allows users to ping the admins of issues,
* To separate clients from users, there should be roles given out. Normal registration should give base user access

#### 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?*

* Utilizing admin rights and SQL, admins can add/remove/modify user information. This should only be granted to the admins of the application.

#### 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?*

* Some sort of token should be given to user upon registration like a JWT token. Username and password will be the key to accessing this token.
* Once the user clicks sign in it should match the username and password in the DB and verify it. If it is wrong, it should return an error from the server.
* To combat against brute force hacking the account should be locked after a certain number of attempts, normally 3-5 attempts.
* If a user forgets their password or username there should be a password reset or username retrieval tool; that use security questions or a verified email.

### 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.”*

* User’s token should be verified when prompting to login after they enter username and password
* Users should be able to see and schedule openings for lessons and schedule with specific instructors.
* Users should be able to modify their schedules
* Users should be able to click and see their progress
* Users should be able to see any related training course
* Users should be able to view, select, and modify their training package.
* Admins should be able to update training packages.

### 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 should be able to interact with only their data
* Interface should have functional buttons
* Interface should follow clean UI/UX
* Interface should be able to interact with it in mobile and web
* Interface should show and allows users modify their schedule and package
* Interface should allow users to see their progress
* Interface should show students instructor
* Interface should show driver notes, special needs, test progress, driver and student photo

### 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 ability to upload and modify driver and student photos.

### 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?*

* Scalability is always limited depending on how you set up the interface. The more you zoom in the more the interface begins to alter and containers then start stacking on top of each other.

### 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