# 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 company DriverPass would like to create a system designed to help students practice for their driving exam. The system created would allow students to practice tests, schedule driving lessons, and have access to any extra DMV learning materials. The client wants a cloud based system that offers flexibility, security, and ease of access for customers and employees.

### 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 has noticed an increase in the amount of students failing their driving exams because they don’t get enough practice. DriverPass wants the system to help fix the issue by providing online and in-person training. The system would need to work both offline and online by keeping the user’s data safe and include features like reservations, tracking, and reporting.

### 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?*

* Customers should be able to register for training and practice exams.
* Allow users to schedule, modify, or even cancel driving appointments.
* Implement tracking mechanics for user activity.
* Have instructor feedback.
* Make sure data is secured and there are consistent DMV updates.
* To deliver a cloud based solution with automated backups.

## 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 web-based and accessible for desktops and mobile devices.
* The system should run efficiently enough so that it is optimal for user experience.
* The system should be updated in real time for online users.

#### 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 run on a cloud based system and must be compatible with Windows, macOS, Android and iOS.
* A relational database would be best suited as storing user data with schedules because it keeps user information and schedules organized.

#### 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 accounts should be role based like Admin, Instructor, Customer to control who does what.
* The system should have check inputs to avoid any unnecessary errors, like making sure appointment times make sense.
* The system should keep a history of changes as if something goes wrong, it would be easy to see who did what.

#### 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?*

* Yes, making changes could happen without touching the code. If the Admin does not want anything on the course, they can just disable it. For example, like a training package of sorts.
* The system would be flexible enough to have consistent updates, in case of adding any additional training packages.
* IT staff should have access to user management and system settings in case any problems arise like getting locked out of your account.

#### 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?*

* User logins will have a username and password.
* If someone forgets their password, they should be able to reset it easily with a proper email address.
* The system would encrypt any data to keep personal information safe (I.e, payment information, account information, etc.).
* If someone tries to hack into the account with “brute force” too many times, then the account will be temporarily locked to prevent any access.

### 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 should be able to sign up, schedule lessons, and take practice tests.
* The system should be able to track their progress, including test scores and feedback from instructors.
* The system and updates should sync so that way when new learning material is updated, users have the freshest materials to date.

### 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 should show lesson schedules, test progression, and instructor feedback.
* Customers should be able to book, cancel and reschedule appointments easily.
* The interface should be simple and easy to navigate on both desktops and mobile devices.

### 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 will have internet access.
* The business will provide any digital training materials for users.
* Users will follow the rules and not try to exploit the system.
* Their will be different level of people accessing the system with different technical skills so the system needs to be simple yet intuitive.

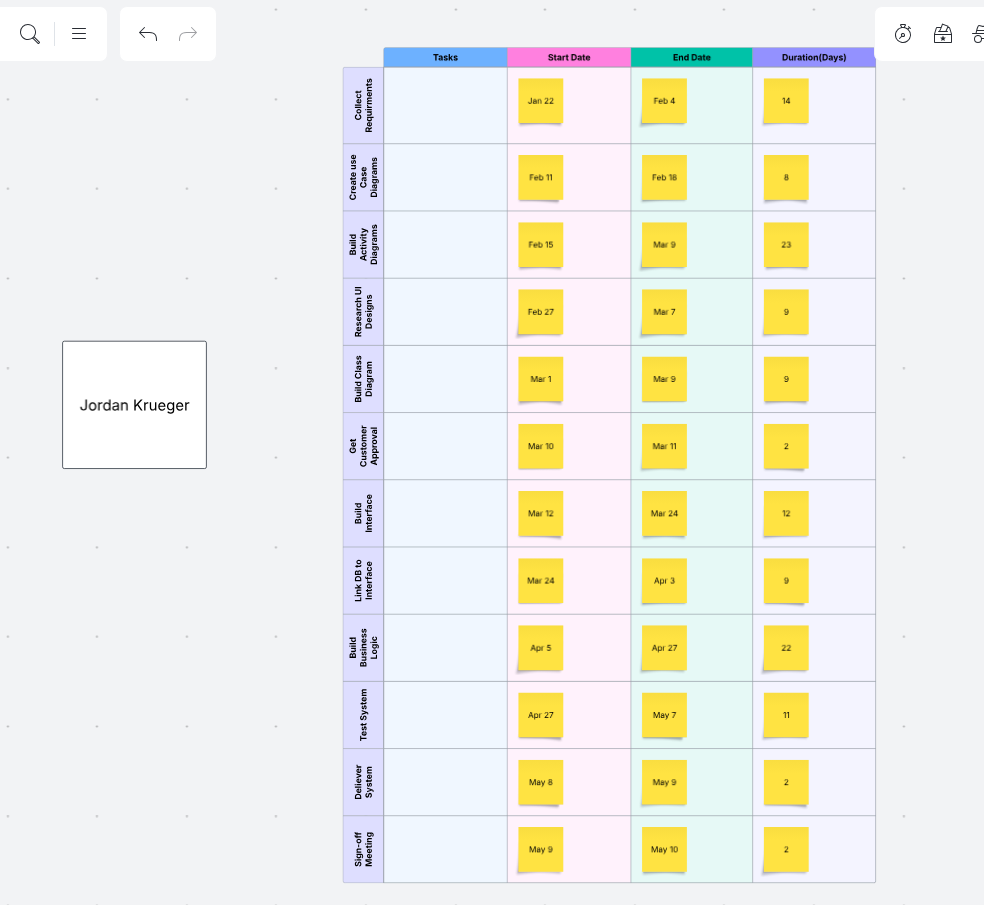
### Limitations

*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?*

* Communication between instructor and student wont be issued within the system, instead out of the system.
* Customer support may be limited, so it may take time for users to have assistance.
* Minor bugs at launch could be an issue with test taking, navigating the UI, or even payment processing.

### 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.*

I’m not sure if you meant stack bar original gantt chart like the discussion post or like a schedule of the gantt chart like shown below so I posted both. Just in case :)

A screenshot of a computer

AI-generated content may be incorrect.