# 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 is looking to create a system to allow student drivers to take classes and practice tests prior to getting their driver’s license. They would like to offer three packages to those seeking assistance in preparation for the driver’s test. Through these packages, they offer in-person driving lessons with trainers, educational materials, and practice tests.

### 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 believes there’s not enough resources for student drivers and this makes them more likely to fail their driving test. In order to fix this problem, they will be offering various services which they would like to be accessible (at least in terms of registration) online. This will require a system to be developed that can be used regardless of operating system.
* DriverPass is looking for a cloud-based solution. Their focus is on coordinating drivers with customers and giving courses, not managing servers and security. This indicates a serverless system to avoid future server maintenance.
* The server should utilize a relational database management system to organize information regarding customers, employees, courses, and driving lessons.
* A web interface should be made to connect to the system by logging in using one of three levels of security clearance: customer, employee, or manager.

### 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?*

User-Interface capabilities

* Customers can:
  + create a new account or edit their existing account without assistance from DriverPass.
  + create reservations for lessons online, in-person, or over the phone.
* Employees can:
  + register customers by collecting the following information over the phone: first name, last name, address, phone number, state, credit card number, expiration date, and security code.
  + access the following details regarding a driving lesson: driver information, customer information, car information, date, time, as well as who made the reservation, modified it, or canceled it.
* Managers can:
  + do anything employees can as well as perform a password reset or block on any account they wish.
* All accounts above can:
  + securely login to the server through some form of authentication using a web browser.

## 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 will be accessed online using one of the commonly used web browsers, i.e. Firefox, Google Chrome, and Safari. It should be responsive to user input, such as keystrokes and mouse movements. Specifically, entering information in a text box, filling out forms, etc. The system will be updated on an as needed basis. DriverPass has made it clear that this is just the first iteration of the system, the first-draft.

#### 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 in a Unix environment. This will not only make for quick development times, but also lower costs by a considerable margin. Linux distributions such as Fedora and Ubuntu provide a method to boot a preconfigured server stored in the cloud (serverless). The server should utilize a database to provide query results to the user of the application. A simple MySQL instance should suffice. Provided the server is accessible via API endpoints, the server will also be accessible from any commonly used web browser.

#### 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?*

* The system may distinguish between different users using a unique identifier. This could be a combination of their name and birthdate, or a username chosen by them. Regardless of the method chosen, the identifiers have to be 100% unique across the entire system to avoid query conflicts. The system should have many tests to ensure functionality before presenting it as a feature. Nonetheless, errors are bound to happen. The system should inform admin of a problem when

#### 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?*

* Changes to user information can be done via API endpoints from a web-based application. These endpoints will execute SQL queries to mutate user information according to the user’s request. Administration may also be able to change some information in a user’s account, either by request or automatically. For example, administration may choose to remove an account depending on it’s age, or interaction with the company.

#### 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?*

* In order for a user to log in, the system must first authenticate the user’s credentials which are stored in the database. If the user can match their credentials, then they can login. As a general rule, any data being transferred via a port connection (the server-client connection namely) should be encrypted.

### 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 provide API endpoints to be used in a web-application.
* The system shall form these endpoints in such a way that changes can be made to user information by either administration or the corresponding user.
* The system shall encrypt data transferred over the network to ensure data privacy.
* The system shall authenticate user credentials before providing system access to the user.

### 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 be simple and easy to use by anyone. There should be labels for inputs to clearly indicate what information belongs in a text-box and what happens when a button is clicked. The interface should support three user types: manager, employee, and customer. The following security clearances apply to the interface:
  + A manager may manipulate employee and customer information as well as their own information.
  + An employee may manipulate customer information as well as their own information.
  + Lastly, a customer may manipulate their own information alone.
* Aside from account data, the system should also present data based on the user type. The specific type of information and the way it is laid out is yet to be known. Although, Liam did provide a basic mock up of what the interface may look like. Mainly the information that should be accessible by the customer, employee, and manager regarding a driving lesson. Other necessary items include the login page and account details page. Optionally, there may be a FAQ page for any user regardless of login status to view.

### 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?*

* While security is important, it does not hold the same importance as systems that operate on banking or medical data. This enables us to utilize a remote server, rather than hosting one on local machines.
* This is a new project, so the system needs to be adaptable to unforeseeable requirements. To accommodate, a serverless context was chosen to provide flexibility and scalability.
* DriverPass has acquired the funding to pay for remote engineers to maintain the site, and potentially implement new features upon their request.
* DriverPass has acquired the funding to pay for the cloud storage and virtual machines necessary to run the server and database.

### 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?*

* When choosing cloud-based solution with a serverless environment, the core limitation is budget. Engineers must be paid to maintain the site, and the services providing cloud storage and virtual machines must be paid for as well.

### 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.*

