# 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, is looking for a new web application to manage their schedule and track driving students.
* The application should allow customers to create an account and schedule an appointment with DriverPass, based on 1 out of 3 available packages offered.
* The app should be web-based and cloud-based.

### 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 requires an application that can be used by employees and customers to schedule appointments, document student information, and document driving tests.
* The app will require a login component so that users can create and login to their account, as well as change their password.
* There are multiple roles needed, like owner, administrator, editor, user, etc…
* Various user roles need the ability to add, remove, or change reservations.
* The app needs a tracking/logging component to keep track of all changes.
* The app will be hosted via the cloud, and web based.

### 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?*

* The application should be able to allow users to login.
* Customers should be able to purchase a package from DriverPass.
* Customers should be able to schedule, cancel, and change their appointment with DriverPass.
* Employees should also be able to schedule, cancel, and change appointments for the customers.
* Employees need to be able to view which driver, vehicle, and time is needed for the appointments.
* Drivers need to be able to leave notes on the lesson, like the Lesson Time, Start Hour, End Hour, and Driver Comments.
* The application should require the students’ information like their name, number, address, email, etc… This could be filled in by the student or the secretary.
* If no times are available for a selected hour or day, the application must reflect that.

## 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?*

* DriverPass will be a web-based application, likely hosted via the cloud.
* The app will be making database queries and will need to be relatively quick when returning information to users.
* The application will need storage for the database and user-uploaded photos.

#### 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?*

* To host the web application, I’d recommend Windows or Linux server.
* Windows server comes with tools to host web applications, but optionally the app could be hosted via apache server.
* We will need to connect the application to a database to save and query user data.
* The website can be built using a number of languages and frameworks, like ASP.NET for example.

#### 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?*

* Users will likely be based off email or a unique username. They will require a case-sensitive password to login. If a user forgets their password, a new one will be emailed to them.
* An admin could receive a notification when a user’s account is locked out due to maximum login attempts.
* Admins could receive notifications when the application and/or database have lost connection.

#### 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?*

* Customers should have the ability to edit/update their account information. Employees with proper roles should also be able to update customer information from the web application.
* IT administrators should have the ability to access customer information, create employee accounts, manage employee roles, and disable employees.
* The web application and database should be backed up, and platform updates shouldn’t affect it. Maybe a test environment should be enabled for QA to test how platform updates affect the web app.

#### 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 should need a password to login to their account.
* We may want to send an authentication email so that we don’t have an issue with bot accounts.
* The developers should use HTTPS as the protocol to receive and send data. This is important because the data transfer will be encrypted and more secure for sending passwords or other sensitive information.
* Developers should add a “maximum attempts” to login attempts, which should mitigate brute force hacks. In this case, user accounts could either be locked for “X” amount of time, or an admin must unlock it.
* If a user forgets their password, they should have the ability to reset it via 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.”*

* The system shall display the home page containing information about DriverPass and their services, with a nav bar that links to other pages. Notably, the nav bar should show a “register” and “login” option.
* The system shall register users when necessary data is given and password passes minimum requirements.
* The system shall validate user credentials when logging in.
* The system shall allow customers/employees (for customers) to reserve driving lessons, including giving them the ability to select from a schedule of available dates/times.
* The system shall allow customers/employees (for customers) to purchase an education package.
* The system shall be connected to the DMV and display the latest rules, policies, and sample questions for customers to practice.
* The system shall allow customers to take online practice tests, which will also display on their profile for Drivers to view.
* The system should contain a page that shows the customer’s online test progress, their information, the Driver and student photo, and the Driver’s notes.
* The system should query the database and generate reports for employees with allowed access to view.

### 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 interface should be simple to understand for new and existing users to the system.
* Users will be based on roles, and can either be a customer, employee, administrator, or owner/manager.
* All users should be able to register/login and reset their passwords.
* Customers should be able to view and purchase DriverPass packages, input their information, study and take online tests, and schedule driving lessons.
* Employees should be able to create customer accounts, fill in their information, set their package type, and manage customer/driver schedules.
* Administrators should be able to create employees and set their roles, get notifications for application problems, and manage the application (like backups).
* The owner should be able to create and access reports, view driver schedules, and view active users.
* The user interface will need to be dynamic so that its size and functionality adjusts based on the monitor or device 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?*

* Users have a computer or device with a browser.
* Users have an internet connection.
* Users have an email address for account creation.

### 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?*

* Clients in certain regions may have slower access to the website.
* The client doesn’t have an on-staff developer that can help debug or create modules.
* It’s possible that not all client devices will be able to open the website.

### 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

Description automatically generated