# 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?*

* Client is Driverpass, a company that helps train students to pass their DMV driving exams.
  + Their purpose is to provide a service that trains drivers to pass their driving exam through online classes and practice tests.
* The application should provide scheduling functionality to allow appointments to be made and tracked online.
  + Utilizes a roles and permissions system with:
  + Admin: password resets, overrides for appointment information.
  + Secretary: viewing customer information, booking appointments on behalf of the customer, updating appointment status, modify (but not delete – security risk) appointment.
  + Customer: purchase appointment package through online web application, view course progress.

### 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?*

* Data should be accessible but not modifiable while offline.
* Should have a system to track user roles and permissions.
* Components needed:
  + Authorization
  + Database for storing appointment/customer info
  + Customer-facing web UI
  + Internal-use scheduling system
  + API to receive updates from DMV

### 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 book reservations for driving lessons. The reservation system should keep record of:
  + The package the customer purchased:
    - Package One: Six hours in car with trainer
    - Package Two: Eight hours in car with trainer, in-person lesson
    - Package Three: Twelve hours in car with trainer, in-person lesson, online course access
  + Who made a reservation.
  + Which instructor and car the reservation was made with.
  + Who canceled the reservation.
  + Who last modified the reservation.
  + Personal info of the customer
  + Pickup/drop-off location.
* A package should be disable-able by a non-developer if discontinued.
* Customers should be able to reset their own passwords.
* Customers should be able to book appointments over the internet.
* Customer should have the option to leave comments about their visit.
* Drivers/employees should be able to access a page that tracks:
  + Online course progress
  + Time of lesson
  + Start hour
  + End hour
  + Driver comments

## 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?*

* Web-based (customer-facing)
* Application-based (internal use)
* Performance is not likely to be a concern for such a lightweight application but should be as fast and efficient as possible.
* Should receive updates from DMV API.

#### 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?*

* A database will be required.
* Application platform should be discussed with client, built to fit whatever OS they’re currently using.
* Web application should be compatible with Linux, Mac, and Windows OS as well as Android and iOS mobile operating systems to ensure the broadest possible reach.

#### 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?*

* Authentication will use a username (email) and password system. Users will also be required to provide a phone number to reduce the likelihood of a single user being able to create multiple accounts.
* Emails will not be case-sensitive, passwords will be. Passwords will also require a number and special character.
* If an API call fails, such as when a connection to the online database cannot be established, the admin should be informed.

#### 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?*

* IT admin should:
  + Be able to manually reset passwords
  + Adjust permissions for other users
  + Manually modify existing appointment data
* Should be built in a modular fashion to allow admin to modify packages, etc., in future versions.
  + Microservices architecture to allow individual components to be swapped out.

#### 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?*

* Users need an email and password to log in.
  + Emails should be verified before an account is activated.
  + If a user logs in from a new location, two-factor authentication using their phone number should be used.
* Data exchanged will be encrypted and users will connect to the site using HTTPS.
* User accounts will be locked for 4 hours after 5 unsuccessful login attempts. An email will be sent to the user to inform them if this happens.
* Users who forget their password will be sent a password reset link to their 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 validate user credentials before displaying tasks/allowing a task to be performed.
* The system shall track appointments, the status of the appointment, and all relevant information thereof.
* The system shall securely handle sensitive user data such as credit card and address information.

### 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.)?*

* Customers will access application through a web interface.
  + Should allow booking/canceling an appointment online through a secure intake form.
  + Should allow for tracking of online course progress.
* Secretary will access application through local program.
  + Should allow booking/canceling an appointment when online.
  + Should provide an intake form that guides secretary through onboarding process of new customers.

### 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?*

* Should confirm what platform will be used to run the application internally (on the device used by the secretary).
* Assuming the DMV provides an API that allows for automatic updates.

### 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?*

* Data will only be modifiable when online.
* Dev will be required to add/remove modules, at least until a more modular architecture can be implemented.

### 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.*

A screen shot of a computer

Description automatically generated