# 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, DrivePass, would like to address the need for comprehensive training and preparation for DMV Driving Tests. DrivePass’ purpose is to provide that training to clients preparing for the driving test in the form of online classes, tests, and on-the-road training by creating a system that enables the company to manage reservations, store client and company related data, and provide a platform that houses their online training material and 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?*

* DrivePass has requested a database that is accessible from mobile or desktop devices and can export reports in the form of CSV files.
* The system should have a role-based user permissions, enabling an admin to delete or modify existing users and passwords.
* The system should also track any changes make to records, including changes to reservations such as the last person to change it and who canceled it. The activity must be available as a report that can be exported, providing a log of recent activity.
* Reservations must be available to make by phone or online through a customer’s account.
* When making a reservation, the system needs to store when the reservation is scheduled for, including the date and time.
* The system needs to be able to track and manage reservation details such as the user, the driver, the car, and the date and time of the reservation to ensure availability of those resources.
* The system will have an integration with the DMV to receive any updates to policies, training questions, and rules and provide a notification when such an update is received by the system.
* There will be a front-end interface that is user-friendly and displays details such as reservations, tests, and other account-related information. Tests should show a status of either, “not taken”, “in progress”, “failed”, and “passed”. The “Driver’s Notes” section should display any details or notes provided by the driver that is matched with the lesson time, starting hour, and ending hour as well.
* The interface should also include pages for an input form, contacting the business, and contacting the customer as well.
* The system needs to be run through a cloud-based system to ensure the company does not need to be concerned with managing components such as security and backups.

### 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?*

* Allow the owner and IT Officer to manage users, passwords, and reservations in the system.
* The system will provide an interface in which the customer can create a profile and provide account related details such as their name, phone number, etc.
* The interface should also include a page those displays Driver’s Notes, Tests and their current status, customer info, special needs, and a photo of the driver and student as well.
* The system will offer a way for customers to create reservations, designating the date, time, pickup, and drop-off location which should be the same as the pickup location as well.
* The system must have an integration with the DMV to receive updates
* Allow a user to create, cancel, or modify a reservation.
* Provide customers with access to training materials and lesson plans
* Provide a method of accessing data securely whether directly or remotely, so reports can be run.

## 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 need to run in a web-based environment that is compatible with all primary browsers, such as Firefox, Safari, and Chrome.
* The system should be accessible on a web-based environment from mobile and desktop devices.
* The system should experience minimal delay when being used and be responsive, providing quick load times and provide a seamless experience for the user.
* Online courses and test must ensure the experience is smooth and responsive, with little to no delay.
* Updates should be issued as needed, including when the DMV provides any notifications for changes made to best practices, learning material, and other essential information.
* Automatic updates should be employed where possible when updates are provided by the DMV
* A notification needs to be sent to the appropriate administrative users when such an update is registered from the DMV.

#### 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 be platform agnostic, capable of running on Windows, Linux, and Unix based operating platforms that can use a web browser.
* The system should be a web-based system deployed over a cloud system, capable of running on any operating platform that is able to access the website.
* The system shall use a relational database management system from the cloud-based system provider to handle back-end data management and changes.
* The system should utilize a cloud system, such as AWS, Azure, or GCP to ensure proper scalability and to reduce managing requirements.

#### 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?*

* Different users should be distinguished based on registered user names and email addresses.
* For the purpose of username or email address credentials, input should not be case-sensitive to ensure potential login issues related to user input are mitigated.
* Administrators for the platform will be notified of any issues that include blocked logins from repeated unsuccessful attempts, password reset requests, or when unusual activity is identified – such as repeated attempts from suspicious IP Addresses or potential security risks.
* The system should employ role-based user identification to distinguish standard users and administrators, granting the appropriate privileges to each.

#### 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?*

* The system should user management from administrators, potentially through the use of a RDMS for the purposes of user modification, such as adding, removing, or changing details pertaining to a specific user.
* User permissions should also be eligible for modifications through the same methods as mentioned in the previous requirement provided.
* Modularity and scalability should be taken into consideration when developing the platform to ensure system updates can be deployed without issues arising.
* Version control systems should be utilized to ensure system updates can be deployed without longstanding issues arising.
* IT Administrators should have full access to the system, including the ability to modify users, incorporate changes to the curriculum, and disable certain packages based on availability.
* Changes to packages, whether in the form of adding or removing modules, should be available to designated developers or system analysts.

#### 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?*

* Login attempts should be executed through the use of login credentials consisting of username or email, and passwords.
* Passwords should adhere to best practices requirements, being at least a specified length, containing varying characters, and complexity.
* Best practices such as SSL, certificates, or TSL encryption should be used to ensure any data exchanges or connections are secure.
* In the event of a brute force hacking attempt, the user’s account shall be locked until an administrator on the system is able to handle accordingly after being alerted to the locked account.
* The system shall provide the option to reset a password for a user, utilizing their provided email address as a means to do so.
* Multi-factor authentication should be reviewed and implemented if appropriate for the system’s needs.
* The use of security questions should be implemented in the event a user needs to recover their account.

### 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 authenticate or validate user credentials when logging in
* The system shall provide the means for a user to modify their own account details.
* The system shall provide the means for a user to reset their password, if needed.
* The system shall adhere to role-based permissions to distinguish priveleges between ordinary users and administrators.
* The system shall monitor and store changes made to data records, including appointments and account changes.
* The system shall track what user made changes to data records.
* The system shall track what type of change was issued to data records, such as the time of an appointment, pickup location, and the car selected for the appointment.
* The system shall identify assigned drivers for appointments.
* The system shall provide the option to export an activity report from the system’s logs
* The system shall provide the ability to users to make reservations, including the chance to provide date, location, and time preferences as well.
* The system shall support user registration through the assigned users.
* The system shall integrate DMV updates to receive updates regarding changes to best practices or curriculum changes.
* The system shall utilize a user-friendly interface adhering to accessibility best practices.

### 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 system’s interface will be available for different users on this platform including customers, IT Officers, Office secretary, and Administrators.
* The system’s interface should display online test progress, driver notes, customer information such as name, phone number, address, etc., special needs, and the relevant driver’s photo and customer’s photo.
* There should be a page included in the interface that provides an input form for the customer or secretary to fill out.
* The system should also include a page in its interface providing contact options for the customer to contact the business and the business to contact the customer.
* The user should be able to see the progress of tests and lessons completed in the interface.
* The user should be able to interact with the interface through a browser that is compatible with a mobile device and desktop device as well.

### 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?*

* When a user registers an account, the assumption is made that they enter valid information.
* When a user creates a reservation, they will enter valid information.
* The cloud provider will provide management services to alleviate the managing burden from the company.
* The cloud provider will offer database options.
* Users have access to internet and devices necessary to access the platform.
* The DMV provides a method to give updates in an appropriate time.

### 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?*

* Some limitations may exist from the cloud provider in relation to the services they offer, such as when support is available, when updates are issued, how scalable the service is.
* Changes made to package modules and lesso plans will require a system analyst or developer.
* Performance may be limited by the service offered by the cloud provider or strength of connection to the internet.
* Time constraints may limit available features.
* Availability of staff to work on system architecture or features may vary, limiting the workflow of developing the system.

### 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 screenshot of a computer

Description automatically generated