# 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 for this project is DriverPass
* DriverPass wishes to offer driver training services through a completely new system
* Looking to address high failure rates among license tests for new drivers

### 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 desires a system to access and manage users and the services they offer.
* User side includes functionality for online classes, practice testing, and reserving lessons.
* Client side allows management of data and employee access rights.
* Client does not have the desire to purchase or manage hardware/security for this system.

### 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?*

* New user registration system: username and password, password reset ability.
* Reservation system for in-person driving lessons: user applies over phone or via internet.
  + Required info:
  + First Name, Last Name, Address, Phone, State, CC info, pickup location, drop-off (same as pickup.
  + Make, view, modify, cancel appointments online.
  + Customer ability to select/buy from predefined “package” options.
  + Driver ability to leave comments on lessons before or after completion.
* Practice testing and online classes for users:
  + Accessible online via system UI.
  + Connected to the appropriate state DMV.
  + Client is notified when driving rules, policies, or sample questions change.
  + Test progress and completion for customer:
    - Test name, time taken, score, status [not taken, in progress, pass, fail].
* Data access for employees of different roles
  + Manager has full system access over all accounts
  + Ability to grant or revoke access to new or terminated employees
  + Read-only or read/write access roles to data.
* Fully online system
  + Cloud-based with no need to manage/maintain server hardware.
  + Security packages handled by cloud provider.
  + Graphical user interface for customer to interact with using their own device.

## 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 a web-based cloud application accessible online.
* Responses from the system to client should not exceed 10 seconds.
* System is updated nightly; DMV policies for all states queried and notification of new policies sent to system administrator.

#### 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?*

* Front end:
  + System client access available from personal computers running Windows, Mac, or Linux.
  + System client access available from mobile devices running iOS or Android.
  + Client access available through major internet browsers: Edge, Firefox, Chrome, Safari.
* Back end:
  + Server application runs on reliable, managed cloud computing infrastructure such as AWS.
  + Cloud application should respond to calls with HTML for client display.
  + Server must have encrypted and redundant database space available for user personal information storage.
  + Server application uptime > 99.9% to meet client satisfaction.

#### 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?*

* Each user must have a unique identifying username.
* System must distinguish between ‘customer’, ‘employee’, and ‘administrator’ type roles in order to offer the proper access to functionality for each. User is assigned role at account creation.
* Username along with associated password will be case sensitive.
* Administrator should be informed in the case a user forgets a password via an automated request system initiated by user.

#### 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?*

* Administrator has access to modify, create, delete users and roles as needed.
* User role is a property assigned to each user and will be editable by administrator without need for changing code.
* The cloud-based server application must be able to update without interrupting client access, using a continuous delivery model.
* System must check for browser information from client and ensure that an up to date version is running.

#### 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?*

* Client must provide proper username and password combination in order to log in.
* Server application stores password hashes only.
* Password must be at least 8 characters including at least one uppercase character, lower case character, and number to combat brute force hacking attempts.
* Maximum successive login attempts cannot exceed 5 before account is locked and administrator is informed.
* User can request a password reset and will have to verify their email address or phone number, one of which is provided at account creation.
* Communication between client and server should occur using HTTPS for security and encryption of data transfer.

### 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 allow the user to create a new account or log into an existing account.
* The system shall validate user credentials and role upon logging in.
* The system shall allow the user to create a new in-person appointment online.
* The system shall allow the user to take online driving classes and practice tests.
* The system shall allow employee users to modify appointment or test data in user accounts.
* The system shall allow employee users to create new appointments for users who have submitted information via phone.
* The system shall allow administrator users full access to modify employee or user roles.

### 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.)?*

* System interface accessible through internet browser for all users.
* ‘Customer’ user interface needs
  + Online test section
    - View tests
    - Take a new test
    - View test progress
  + Information section
    - View first and last name, address, city, state, zip code, phone number, and email along with any other needed information.
  + Driver notes
    - View upcoming appointments for in-person lessons
    - View notes from previous in-person lessons
  + View or edit special needs
  + View driver photo
  + View or edit student photo
* ‘Employee’ user interface needs
  + Pull up user account information
  + Modify existing information
  + Create new appointments
  + Add new content
* ‘Administrator’ user interface needs
  + View ‘customer’ or ‘employee’ users
  + Add, change, or remove accounts or roles
  + Update system

### 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 of the system will have a personal computer or mobile device with an internet connection.
* A suitable cloud application service is available to meet the needs of this system server application.
* The system content will be managed by the client DriverPass.

### 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?*

* System is only accessible through internet connection to the cloud application.
* System is not openly accessible without a username or password.
* System exists as software only, the client and customer must provide hardware required to access the system.
* System must be built under the defined budget of the client DriverPass and must be known before development begins.

### 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