# 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 purpose of the project is to render consulting services for DriverPass in order to help them develop an program that will enable people to more easily and frequently pass their DMV driving 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?*

* The system will provide online classes and practice tests as well as on-the-road training. Users should also be able to schedule driving tests with the DMV through the program’s online interface. This interface also provides users the ability to access and change their information. In addition to all of this, the system will keep track of user progress and the availability and status of training drivers and cars.

### 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 system will be connected to the DMV to stay compliant to ensure the lessons and practice tests are current with the DMV’s expectations. When the system is complete and up-and-running, it should enable driving students to pass the DMV driving test much more frequently, which can be measured statistically against the frequency of test passing among people who didn’t use DriverPass. It will also keep track of students’ lesson plans for driving lessons, as well as their proficiency in the DriverPass tests. The system must be connected to the DMV via the Internet and to cloud servers for data management. A hierarchical system of authorizations will enable efficient management of the entire program and DriverPass as a whole. The system must also track the current to-the-minute state of teacher drivers, cars, and pickup locations.

## 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 must be ready for quick and immediate updates to the statuses of all data objects used to keep track of customer and company information within the system (dependent on user authorization level). This requires secure connections to the cloud for data storage as well as to the DMV and company computers and user internet-connected devices for data retrieval.

#### 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?*

* I believe that a Linux distribution will be appropriate for company computers and backend management. However, JVM could be implemented to enable more protection to the company computers.

#### 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?*

* Implementing an automatic email service to notify admins of potential problems with the system is important. Users will have case-sensitive usernames and passwords for their login, as well as verified credentials (such as security questions). Top-level authorization for admins must come from backend computers with kernel-mode access and several layers of security, such as changing passwords and facial recognition. Lower-level admins such as IT consultants and office managers will require the same amount of verification but will not enable kernel-level access unless it is authorized by higher-ups.

#### 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?*

* User information will be stored as a set of objects containing the various types of information that the user object will implement. This will enable different levels of security clearance to changing user info. For example, the user should easily be able to book appointments, but shouldn’t be able to alter the state of cars and training drivers, while an office manager shouldn’t be able to alter the user’s appointment schedule.

#### 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 must enter the correct login information (username, password, and security question solution). Before this happens, the client must establish an SSL connection with the server in order to ensure the data is encrypted. If there is a brute-force attempt, the user’s sensitive information will be locked and inaccessible until top-level authorization resets this. Also, the rest of the user’s data objects will be backed up in case a security breach or brute force attempt is detected. If the user forgets their password, they can simply call the DriverPass office or else complete a verification process over the Internet.

### 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 when logging in.
* The system shall provide an interface to allow the user to track their progress, view their plan details, schedule driving tests and on-the-road-lessons, and access their online courses.
* The system shall provide an interface to allow office managers to communicate with and track the status of drivers, cars, and appointments.
* The system shall provide an interface to allow IT staff to inspect the underlying code of the system.
* The system shall provide an interface to allow administrators to escalate authorization for IT staff and office managers.
* The system shall provide an interface for staff to view data stored in the cloud, request higher authorization, and interfaces to update information.
* The system shall provide tools for administrators to have full control over the system.
* The system shall provide an interface for users to create accounts and view account 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.)?*

* The superuser administrator can view and modify all code and data objects in the system.
* The general administrator can view and modify some code and all data objects in the system.
* The IT administrator can view all code and data objects in the system.
* The office manager can view some code and most data objects in the system and modify some data objects.
* The user can view and modify some data objects in the system.
* The user can access the interface from a variety of mobile applications and browsers.
* All roles with higher authorization must access the interface from the company computers running the backend code on the Linux OS, and in some cases on the JVM.

### 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?*

* The users must have access to a phone as well as internet in order to register.
* The company must implement proprietary security software and hardware.
* The company must query data from the DMV and convert it to a format consistent with the system’s data architecture.

### 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?*

* The system must be cloud-connected.
* The system must have sufficient memory to maintain a stable and secure implementation.
* The software development must be concise enough to save time, but robust enough to save money and profit in the long run.

### 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 calendar

Description automatically generated