# 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

* The client, Driverpass, is looking to make a platform for individuals to proactive their driving skills prior to taking their test at the DMV
* They would like their information to be accessible for online and offline access.
* They would like the system to be as stable as possible so they can focus on the business
* DriverPass wants the application to be web-based and preferably utilizing cloud platforms.

### System Background

* DriverPass wants to fix the high percentage of failing outcomes on the state driving exams.
* They want to provide online practice tests and courses as well as in person drives, customers can schedule.
* The system must have a user interface accessible from the web where users can schedule their drives
* The system must also have a business interface so the secretary can manually schedule appointments for clients

### Objectives and Goals

* Provides 3 packages, package one: six hours with trainer. Package two: eight hours with trainer and an in-person lesson on dmv policies. Package three: twelve hours with trainer, an in-person dmv policy course and finally, practice tests.
* Must allow for role based security authorization. roles: Big Boss (Liam), IT Officer (Ian), Secretary and users. Ian, the IT Officer must have access to the entire system.
* Must keep records on user drive history. Tracking user pairing with driver, time, and vehicle in use.
* Customers must be able to call the business to make an appointment using their first name, last name, address, phone number, state, credit card information and pick up and drop-off locations. Similarly the customer must also be able to accomplish all of these tasks by themselves through the online portal.
* Measurable tasks include: Building class diagrams, Building interface, and Linking the database to the interface

## Requirements

### Nonfunctional Requirements

#### 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 should be web-based
  + Preferably with a cloud platform for hosting
* The system should operate as fast as possible
* Security updates - ASAP
* Feature updates - Rollouts

Rational: Our client has requested that the system is web based. Additionally, they have also requested, if reasonable, to make the system cloud based. We should aim to make the speed of the system as fast and efficient as possible; This way we can create a positive customer experience. Theoretically, our security updates should be pushed as soon as possible, in order to patch security related bugs. Features should be rolled out periodically on rollouts, ideally anywhere from quarterly to semi-annually.

#### 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?*

* Can use windows-server or Unix
  + Unix is preferred
* Will require a database
  + Mysql, MongoDb, etc.
* Extra Tools
  + Spring Boot - Web Framework
  + Apache NGINX - Web Server
  + Jira - Project Management
  + DataDog - Monitoring Service

Rationale: The system can theoretically use windows-server or an operating system from the Unix family. I recommend Unix since it offers better functionality and support from the open source community for development. Unix is in most cases the go-to OS for its lightweight and secure nature. Our system will require a database in order to store user information, drive history and any other data we will be storing. Additionally, we will also need to utilize backend-tools like spring boot for website configuration, Apache or NGINX for our web server and potentially a project management tool like Jira.

#### 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?*

* Usernames, emails are ways to differentiate our users
* Login info (Username, email, password) should be case insensitive
  + Reduces potential for system errors due to complexity
  + Increases the convenience for users
* inform the admin as an error is detected in
  + User login
  + Scheduling an appointment
  + Database insertion, deletion, etc.
* System should have live monitoring
  + Notify admin through pager or SMS depending on severity.

Rationale: We will be able to distinguish our users by implementing Usernames and Email addresses. Further, we need to make the username, email and password case-insensitive. This way we have a less chance for collisions in accounts. Additionally, case-insensitivity makes a more pleasing user experience. Our system should monitor the user's ability to log in, schedule an appointment and system processing and database errors. In the event that the system meets these problems, the admin should be notified through a pager service on their phone and SMS. We can use a monitoring service like DataDog and a pager to send notification to the system admins phone. Additionally, if problems are critical then the system should also send messages to the sys-admin’s phone.

#### 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?*

* in order to make changes to the user we can implement a user management interface
  + Allows admins to perform these actions through a GUI or CLI
* adapting to platform updates
  + implement best software development practices
  + Version control systems
  + automated testing
  + CI/CD pipelines
* It admin access
  + User management
  + System management

Rationale: The system needs to allow system administrators to add, remove, and modify users without changing code. We can utilize either a GUI or CLI to allow our admins to communicate with the system to accomplish these operations. Preferably a GUI to decrease the learning curve for employees using our system. The system will adapt to platform updates through using a version control system. This way, we can commit changes to the system and resolve any errors that arise before pushing the version to production. Our IT admins will need access to managing users (Add/remove/modify) and system management. Forms of system management such as monitoring and logging. This way, IT admins can perform their job of modifying and maintaining the system.

#### 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?*

* User log in
  + Provide username or email address and a password
  + EXTRA: 2FA/MFA
* Secure connection / Data exchange between client and server
  + HTTPS - Encrypts data in transit
* If there is a “Brute Force” Attempt
  + Account lockout - after login attempts
  + CAPTCHA - Automated attacks
* User Forgets password
  + Sending a password reset link to user email

Rationale: In order for the user to login, they must provide their username or email and password. This will allow us to validate our user and to protect their information. Additionally, if we would like to take it another step further, we can integrate 2-factor authentication to improve our security. For us to secure our connection to the system, we need to implement HTTPS so our client server conversations are encrypted. In the event that someone tries to brute force an account, they should be stopped by an account lockout after a certain amount of login attempts. We should also consider a form of CAPTCHA to aid us in preventing automated attacks. If a user forgets their password, we will have a link for the user to enter their email where we can send them a link to reset their password. We must ensure that this function is most particularly secure; If we choose to use 2FA, we can implement this into changing passwords.

### 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 users to create an account with a unique username and password
* The system shall allow users to schedule driving lessons online
* The system shall store driver notes, including comments and lesson times
* The system shall provide different user roles, administrator, instructor and student
* The system shall allow the admins to add, remove, and modify user accounts and roles
* The system shall notify the admins of any system problems

### 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.)?*

* Needs
  + user-friendly
  + Intuitive
  + Accessible from various devices
* Different users
  + Administrators
    - Manage user accounts, roles and permissions
    - View and manage driving lesson schedules
  + Instructors
    - Need to view their schedule, including driving lesson times and locations
    - Enter driver notes
  + Students
    - create an account, schedule driving lessons, view their schedule, and track progress.
    - Take online tests and study material

Rationale: The user interface needs to be user-friendly and intuitive so it doesn’t require much skill to navigate. It must be accessible to any number of devices, meaning we must ensure all devices have a usable interface. More importantly, the client needs our interface to display: online test progress, driver notes, user information, special needs, driver and user photos. The different users of the interface are administrators, instructors and students. Administrators must be able to manage user accounts and alter roles and permission. They must also be able to manage driving lesson schedules. Instructors should be able to view their schedule that includes driving lesson times and locations. Drivers should also be able to enter driver notes for their student. Students should be able to create an account, schedule lessons, view their schedule, and track progress. They should also be able to take online practice tests as well as study course material.

### 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?*

* Internet connection
* Users have devices
* laws and regulations regarding data privacy and security

Rationale: The most apparent assumptions we are making is that our users have access to the internet and devices in which to access our service. We are also assuming that our system will comply with all laws and regulations before researching. In regard to the proposal of MFA/2FA, we are assuming that the user has access and knowledge of authenticator applications.

### 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 Design limitation
  + Scalability
  + Data security
* Resource limitations
  + Skilled developers/designer
  + Time constraints
    - May limit our ability to accomplish certain features
  + Budget Constraints
    - Can restrict our access to further advances technologies
  + Technology constraints
    - Dependencies

Rationale: Some limitations in the system design are scalability and data security. Scalability is always a limitation to software applications, there is always extra work that is involved when implementing newer, bigger features; nothing is always plug and play. Data security is never definite, there is always the possibility for vulnerabilities, if there is a way an attacker will find it. Some resource limitations: expertise, time, budget and technology. Firstly, it is not easy to find equally skilled developers, which can lead to some errors in code production. With time being limited, we may not be able to put as much detail as desired into certain aspects of the system. Our budget may restrict our ability to purchase new hardware or services like a cloud provider or various SaaS applications. Lastly, technology may limit our feature production through our dependencies. We may find ourselves limited by the functionalities of our dependencies.

### 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.*

