# CS 255 Business Requirements Document Template

Brandon Ellis

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?*

* Collaborate with client, DriverPass, to develop a comprehensive system for driver training.
* Provide a solution that integrates online classes, practice tests, on-the-road training, and efficient appointment scheduling.

### 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?*

* Identifies a prevalent issue: High failure rates in driving tests at the DMV.
* Aims to improve driver training methods to address the shortcomings in traditional approaches.
* Downloadable reports for offline access.
* User tracking to monitor changes made within the system.
* Reservation management system for clients.
* Comprehensive profiles for drivers, cars, and students.
* Notification feature to alert DriverPass about DMV requirement updates.
* Cloud-based web 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?*

* Gather information about what the driver training system needs to do.
* Draw diagrams to show how different parts of the system will work.
* Make detailed diagrams for each part of the system's activities.
* Explore and plan how the system will look and feel for users
* Create a picture of the system's structure using a class diagram.
* Build a user-friendly interface for progress tracking and lesson management.
* Connect the database to the interface

## 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 platform with offline access
* Fast enough to handle rea-time data processing
* It should be updated regularly to keep up with DMV rules and regulations

#### 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?*

* It should be accessible via major web browsers like Chrome, Firefox, Safari, etc.
* It should utilize a relational database to mange user profiles, appointments, etc
* Hosting on a cloud based service like AWS is preferable

#### 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?*

* Users will be identified based on their roles and permissions assigned to those roles
* Usernames should be case-insensitive to prevent login issues
* Passwords should always be case sensitive for security
* Admin should be notified for critical errors in the system, unauthorized access, system downtime, and compliance issues

#### 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?*

* Users should be able to update their account information
* Admins should be able to update user accounts if the User is having a problem
* Updates should either be seamless or announced downtime for maintenance
* The IT admin should have unlimited access to 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?*

* The user should have a username and password to login. Also consider incorporating MFA for extra security
* We can encrypt the data to make sure the data exchange is secure
* The account should be locked if there is a brute-force attempt
* If the user forgets their password we should have a forgot password option where either the user can have a new password emailed to them or answer a security question to reset the password

### 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 allow users to update their profile information, including password, contact information, and payment methods independently.
* The system shall enable administrators or IT staff to update user profiles on behalf of customers when necessary.
* The system shall encrypt all data exchanged between the client and the server
* The system shall lock user accounts to prevent brute force attacks.
* The system shall provide a way for users to reset their forgotten passwords securely
* The system shall allow customers to make, cancel, and modify reservations for driving lessons through an online interface.
* The system shall track and log all user activities concerning data modifications and access to provide an audit trail.
* The system shall send notifications to administrators when potential security breaches or system failures occur.
* The system shall integrate updates and notifications from the DMV regarding changes in rules and regulations.
* The system shall allow administrators to add, remove, or modify the details of the driving lesson packages offered.
* The system shall provide a user-friendly interface that displays test progress, lesson schedules, and driver comments.

### 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.)?*

* Interface Needs
  + User-Friendly Design: The interface should be intuitive and easy to navigate for all users, minimizing learning curves and enhancing user satisfaction.
  + Responsiveness: The design should be responsive to different devices, including desktops, tablets, and smartphones, ensuring consistent usability.
  + Accessibility: Compliance with accessibility standards to ensure users with disabilities can effectively use the system.
  + Security: Secure interfaces to protect user data, especially during login, payment transactions, and personal data updates.
* User Needs
  + Customer
    - Make Reservations: Book, cancel, and modify appointments for driving lessons.
    - View and Update Profile: Access and update personal information, payment methods, and password.
    - Track Progress: View the status of lessons, test scores, and package details.
  + Administrator
    - Manage User Accounts: Add, remove, and modify user accounts and roles.
    - Access Reports: Generate and review activity reports, track user modifications, and manage reservations.
    - Update System Settings: Adjust driving packages, update lesson schedules, and integrate DMV updates.
  + IT Staff
    - System Maintenance: Perform backend updates, manage security settings, and troubleshoot issues.
    - User Support: Assist in resetting passwords and unlocking user accounts after failed login attempts.
  + Secretary
    - Handle Appointments: Schedule, reschedule, and cancel appointments via phone or in-person interactions.
    - Customer Service: Assist customers with inquiries and update customer information as needed.
* Interface Interaction
  + Web-based Application: The primary interaction will be through a web-based application accessible via browsers on multiple devices (desktops, laptops, tablets, smartphones). This allows for flexibility and accessibility from any location.
  + Mobile Optimization: The interface should be optimized for mobile devices, offering a seamless experience for users who prefer to interact via their smartphones or tablets.

### 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?*

* Things Not Mentioned
  + Support for multiple languages
  + Allow for user feedback or to report system bugs
  + User training to familiarize themselves with the system
* Assumptions
* Users have devices able to access the system
* Users have reliable internet access
* Users have a basic level of technological proficiency to use the system

### 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?*

* Scalability: Rapid growth in the user base could cause a bottleneck on the system
* Third Party Integration: Reliability on DMV and payment integration in the system could cause unforeseen downtime or issues
* Budget Constraints: A limited budget may reduce the quality of technology and limit the hiring of skilled professionals such as developers and security experts, potentially compromising system quality and security
* Time Constraints: A tight timeline may necessitate prioritizing certain features, leading to a phased rollout instead of a complete launch, potentially affecting user satisfaction and competitive positioning

### 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.*

