# 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 project involves creating a training system for DriverPass, owned by Liam, to improve students' success rates on DMV driving tests. The system will offer online classes, practice tests, and on-the-road training, accessible from any device. It will manage reservations, user roles, and security, track changes and activities, and enable data access both online and offline. The system will also include scheduling, user management, reporting, and DMV compliance updates.

### 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 wants for the system to offer online classes, practice tests, and on-the-road training to help students pass their DMV driving tests more successfully. The problem they want to address is the high failure rate of driving tests. The system needs components such as online and offline data access, user role and security management, reservation and scheduling features, user activity tracking, reporting capabilities, and DMV compliance updates. Additionally, it should handle user management, enable customers to make and modify appointments, and provide flexible training packages.

### 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 should enable students to register, schedule, and modify driving lessons online, access online classes and practice tests, and provide on-the-road training options. It should support user role management, secure data access, and activity tracking. Measurable tasks for the system design include developing a user registration and login module, implementing a scheduling system, creating online course and test interfaces, designing a driver and vehicle management system, ensuring security and role-based access control, integrating reporting features, and setting up a notification system for DMV updates. Additionally, the system should facilitate data synchronization, user feedback, and performance tracking.

## 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?*

* This system needs to run in a web-based environment, accessible via both desktop and mobile devices. It should operate efficiently with minimal loading times to ensure a seamless user experience, ideally responding within a few seconds for most interactions. The system should be updated regularly to maintain security, fix bugs, and incorporate new features, with minor updates deployed as needed and major updates scheduled at least quarterly. Additionally, it should stay synchronized with DMV updates to ensure compliance with current driving test requirements.

#### 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, running on major operating systems like Windows, macOS, and Unix/Linux to ensure broad accessibility. The back end will require a robust database, such as MySQL, PostgreSQL, or MongoDB, to manage user data, schedules, and other information. Additionally, it will need a web server (e.g., Apache, Nginx) and potentially a cloud service provider (e.g., AWS, Azure) for hosting and scalability. Other tools might include a framework for web development (e.g., Django, Ruby on Rails) and security measures to protect user data and maintain system integrity.

#### 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?*

* The system will distinguish between different users by assigning unique identifiers, such as user IDs, and implementing role-based access control (RBAC) to manage permissions and access levels. User inputs, such as usernames and passwords, will be case-sensitive to enhance security. The system should inform the admin of problems immediately when critical issues arise, such as failed login attempts, data breaches, system errors, or any unauthorized access attempts. Additionally, the admin should receive regular reports highlighting any anomalies or potential issues in the system's operation.

#### 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?*

* Yes, the system will include a user management interface that allows administrators to add, remove, or modify users without changing the code. This interface will provide tools for updating user information, roles, and permissions as needed. The system will be designed to adapt to platform updates by using standard web technologies and frameworks that are regularly maintained and updated. It will rely on a modular architecture, where components can be updated independently. Automated testing and continuous integration (CI) pipelines will ensure compatibility with new platform updates. The IT admin will need full access to the system, including the ability to manage user accounts, reset passwords, configure security settings, monitor system performance, and troubleshoot issues. Additionally, the IT admin should have access to audit logs, system health reports, and the ability to implement and manage system updates and backups.

#### 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?*

* Users will need a username and password to login. To secure the connection and data exchange between the client and the server, the system should use https with ssl/tls encryption. This ensures that all data transmitted is encrypted and protected from interception. In the event of a brute force hacking attempt, the system should implement security measures such as account lockout after a certain number of failed login attempts, captcha verification, and alerts to the admin. Additionally, monitoring and rate-limiting login attempts could be implemented. If a user forgets their password, the system should include a password recovery mechanism. This could be most easily implemented by asking for the email address associated with the account and sending a password recovery email there.

### 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 register for an account online.
* The system shall enable users to schedule, modify, and cancel driving lessons.
* The system shall provide access to online classes and practice tests.
* The system shall manage user roles and permissions to ensure appropriate access levels.
* The system shall track and log all user activities, such as reservations, cancellations, and modifications.
* The system shall generate and download reports in formats such as Excel.
* The system shall enable administrators to add, remove, and modify user accounts through a user management interface.
* The system shall synchronize data with DMV updates to ensure compliance with current regulations.
* The system shall provide a secure connection using HTTPS with SSL/TLS encryption.
* The system shall lock user accounts after a predefined number of failed login attempts to prevent brute force attacks.
* The system shall send email notifications to users for password recovery.
* The system shall enable users to reset their passwords by verifying their identity through security questions.
* The system shall allow customers to choose from various driving lesson packages.
* The system shall assign drivers and vehicles to scheduled lessons and track their availability.
* The system shall enable administrators to disable specific training packages as needed.
* The system shall support both online and offline access to data for authorized users.
* The system shall display user progress on online tests, including test names, times taken, scores, and statuses.
* The system shall maintain driver notes, including lesson times, start and end hours, 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.)?*

* The interface needs to be user-friendly, intuitive, and accessible on both desktop and mobile browsers. Different users of the interface include students, administrators, IT staff, and secretaries. Students need to register, log in, schedule and modify driving lessons, access online classes, and view progress on practice tests. Administrators require capabilities to manage user accounts, monitor system activities, generate reports, and handle reservations and scheduling. IT staff need full system access for maintenance, security, and troubleshooting. Secretaries should be able to assist with scheduling, manage customer information, and communicate with students. The interface should support seamless navigation and interaction across various devices, ensuring a consistent and efficient user experience.

### 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?*

* Some aspects not specifically covered include the detailed layout and design of the user interface, the specific technologies and frameworks for development, the database schema, and the exact methods for integrating DMV updates. Additionally, the design omits considerations for multilingual support and accessibility features for users with disabilities. Assumptions made in the design about the users or the technology they have include the expectation that users will have access to modern web browsers and reliable internet connectivity. It also assumes users possess basic digital literacy to navigate and use the online platform. Additionally, it is assumed that the technology stack chosen will meet scalability and security requirements and that the IT staff will have the necessary skills for system management and maintenance.

### 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 design has limitations such as dependency on internet connectivity for most functionalities, potential complexities in integrating real-time DMV updates, and challenges in ensuring high-level security and data protection. There might also be limitations in scalability if user growth exceeds initial projections, and in flexibility, as some customizations might require developer intervention. As for resources, time, budget, and technology, limitations include the availability of skilled developers, potential constraints in the project timeline that might impact thorough testing and quality assurance, and budget restrictions that could limit the scope of features and functionalities. Technology limitations might arise from the chosen platforms and tools, impacting system performance and compatibility with future updates. Additionally, there may be challenges in providing comprehensive support for all user devices and ensuring robust disaster recovery mechanisms within budget and time constraints.

### 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