# 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 this project is to develop a comprehensive system for DriverPass, a company aimed at improving driver training to reduce the high failure rate (over 65%) of students taking driving tests at the DMV.
* The client wants the system to enable online classes, practice exams, on-the-road training reservations, user management with role-based access, secure data handling, tracking of activities, and integration with DMV updates to ensure content remains current.

### 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 seeks to fill a market gap by providing tools for better preparation, including online learning and practical driving sessions, as current methods like studying past tests are insufficient.
* The problem to be fixed is the lack of effective training resources, leading to high failure rates; the system will offer online access to data (with offline viewing but online-only modifications to prevent redundancy), reservation management for lessons, package selections, and progress tracking.
* Key components include user authentication and roles (customers, secretary, drivers, IT admin, owner), an online learning platform for classes and tests, a reservation system for 2-hour driving sessions with car and driver assignment, reporting tools for activity logs and downloads (e.g., to Excel), security features, DMV notification integration, and cloud-based hosting for minimal maintenance.

### 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?*

* Provide customers with access to online classes, practice tests, and progress tracking (e.g., test name, time taken, score, status not taken, in progress, failed, passed).
* Enable secure reservations, cancellations, and modifications for driving lessons, including package selection (Package One: 6 hours; Package Two: 8 hours + in-person lesson; Package Three: 12 hours + in-person lesson + online access).
* Support different user roles with specific functionalities, such as full access for IT, appointment management for secretary, and oversight for owner.
* Ensure system security, change tracking (who made reservations/changes), and notifications for DMV updates on rules, policies, or questions.
* Achieve a web-based, cloud-hosted system that allows data access from any device, with measurable tasks like generating activity reports, handling password resets, and disabling packages as needed.

## 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 operate in a web-based environment, accessible via browsers on computers and mobile devices.
* It should load pages and process reservations quickly (e.g., under 5 seconds for standard operations) to support multiple simultaneous users.
* The system should be updated regularly to incorporate DMV changes, with automated notifications ensuring timely content refreshes.
* Handle peak loads during reservation times without degradation and support offline data viewing but require online connectivity for updates.

#### 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 compatible with major platforms like Windows, macOS, iOS, and Android via web browsers.
* Back-end requires a relational database to store user data, reservations, test results, and activity logs.
* Hosted on a cloud platform to manage backups, scalability, and security automatically.
* No on-premises servers, as the client wants minimal technical involvement.

#### 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 are distinguished by unique usernames, passwords, and roles; customer data includes precise fields like credit card details and locations.
* Inputs like passwords are case-sensitive, while names and addresses may be standardized for consistency.
* The system should notify admins immediately of issues like reservation conflicts, failed logins, or data inconsistencies via email or in-app alerts
* All changes must be logged with user ID, timestamp, and details for accurate auditing.

#### 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?*

* Admins can add, remove, or modify user accounts and disable packages without code changes, via admin interfaces.
* The system should automatically adapt to browser or OS updates through cloud hosting and standard web technologies.
* IT admin requires full access to all accounts for maintenance, password resets, and blocking access.
* Support future expansions like new packages, but initial design focuses on flexibility in user and content management.

#### 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 must log in with username and password; customer registration requires personal and payment details stored encrypted.
* Secure connections using HTTPS for all data exchanges between client and server.
* Implement brute-force protection by locking accounts after 5 failed attempts, with admin unlock required.
* Provide self-service password reset via email verification for forgotten passwords.
* Role-based access: e.g., customers can't access others' data, IT has full rights track all actions to prevent unauthorized changes.

### 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 during login and support role-based access for customers, secretary, drivers, IT admin, and owner.
* The system shall allow customers to register with details including first name, last name, address, phone number, state, credit card number/expiration/security code, pickup location, and drop-off location.
* The system shall provide access to online classes and practice tests, tracking progress with details like test name, time taken, score, and status.
* The system shall enable customers or secretary to create, cancel, or modify reservations for 2-hour driving lessons, specifying day, time, driver, and car from available options.
* The system shall offer three selectable packages for training, with options to disable them administratively.
* The system shall receive and notify users of DMV updates on rules, policies, or sample questions.
* The system shall display driver notes in a table format including lesson time, start/end hour, and 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 must be web-based, responsive for mobile and desktop browsers, with cloud hosting for accessibility.
* Different users include Customers (register, schedule lessons, take tests, view progress/notes), Secretary (manage appointments via phone/office/online), Drivers (enter comments on lessons), IT Admin (manage accounts, resets, blocks), Owner (view reports, oversee all).
* Customers interact via login portals, dashboards for test progress, reservation forms, and package selection.
* Secretary uses admin-like tools for scheduling, IT has a management console, Owner has reporting dashboards.
* Interactions include forms for input, tables for display, buttons for actions like cancel/modify, a contact page and notification 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 have reliable internet access for core functions like reservations and updates, offline is limited to viewing downloaded data.
* Credit card processing complies with standards like PCI DSS, possibly integrated with a third-party service.
* DMV provides a mechanism for update notifications.
* All users are trained in basic web usage.
* The initial focus is on the three packages.
* Project timeline assumes no major client changes post-approval and availability of team members as scheduled.

### 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?*

* Modifications and updates must be made online to avoid data duplication or conflicts.
* Customizing or adding new packages beyond disabling requires developer intervention.
* No native mobile app, reliance on web responsiveness may limit some mobile experiences.
* Integration with DMV depends on their cooperation and may face delays if APIs change.
* Resource constraints include team availability (e.g., holidays, vacations), and budget for cloud hosting.
* Security features protect against common threats but may not cover advanced attacks without additional tools.

### 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.*

[Insert chart]