# 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?*

* They want to create a cloud based system for DriverPass to improve driver training and preparation for DMV tests
* DriverPass should be able to offer online classes and practice tests through this program as well as allow consumers to reserve on the road training
* Consumers should be able to schedule modify and cancel driving lessons while providing DriverPass with the tools to manage appointments, track user activities, and access data from any device.
* Should be in compliance with DMV regulations which is accomplished through integration for real time updates and notifications
* The main reason for this program is to help DriverPass’s mission of reducing DMV test failure rates

### 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?*

* Web based interface for registering, scheduling, and accessing both online and in person training
* Reservation system to manage driving lessons, tracking drivers, cars, and times
* Data management for customer information, activity logs, and downloadable reports
* Integration with DMV systems for rule and policy updates
* Needs Role based access control for different users
* Needs to have a cloud hosted infrastructure for automatic backups and security

### 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?*

* Enable customers to create, modify, and cancel 2-hour driving lesson reservations online, viewing assigned driver, car, and time details.
* Provide access to online classes and practice tests with progress tracking (test name, time taken, score, status: not taken, in progress, failed, passed).
* Collect and store customer registration data (name, address, phone, state, credit card, pickup/drop-off locations) with role-based access for different types of users.
* Allow staff to track reservation assignments (driver, car, time) and log all reservation actions for auditing, with printable activity reports, and allow the owner to disable training packages.
* Integrate with DMV systems for real time updates and notifications on rules, policies, and sample questions to ensure compliance.

## 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 web based and cloud hosted, accessible via browsers on any device with a responsive design.
* Reservations and logins should process within 5 seconds; progress updates or DMV data pulls should complete in under 10 seconds to ensure a smooth user experience.
* Updates, including DMV rule changes, should be pulled automatically daily or on notification, with system maintenance updates deployed quarterly via the cloud provider.

#### 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 must be web-based and cloud-hosted, accessible via browsers on any device with a responsive design.
* Reservations and logins should process within 5 seconds; progress updates or DMV data pulls should complete in under 10 seconds to ensure a smooth user experience.
* Updates, including DMV rule changes, should be pulled automatically daily or on notification, with system maintenance updates deployed quarterly via the cloud provider.

#### 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 roles via unique login credentials and role-based access; e.g., IT can reset passwords, customers view only their progress.
* Input validation includes case-sensitive usernames/passwords; notify admin via email or dashboard alert on failed logins, reservation conflicts, or data mismatches.
* Precision in tracking: Log all actions with user ID, timestamp, and details; alert admin immediately if DMV integration fails or suspicious activity occurs.

#### 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 accounts can be added, removed, or modified via an admin interface without code changes; packages can be enabled/disabled by the owner without developer intervention.
* The system should adapt to browser/OS updates using standard web technologies (HTML5, CSS); the cloud platform handles scalability for increased users.
* IT admin requires full access to reset passwords, block users, view/maintain all records, and manage logs; future package customizations may need developer input but not in the initial release.

#### 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?*

* Login requires a username and password; customers register with personal/credit card details, validated on entry.
* Use HTTPS for secure connections; encrypt sensitive data (e.g., credit cards) in transit/storage via cloud tools; enforce role-based access controls.
* On brute force (e.g., 5 failed login attempts), lock the account for 30 minutes and notify the IT admin; repeat offenses trigger a permanent block.
* For forgotten passwords, provide an automated reset via email link with a security question or one-time code.

### 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 customers to create accounts and register with details (name, address, phone, credit card, pickup/drop-off locations).
* The system shall enable users to select from three packages (6/8/12 hours driving, with add-ons like in-person lessons/online access) and schedule sessions in 2-hour slots.
* The system shall permit online reservation, cancellation, and modification of driving lessons, matching users to available drivers, cars, and times.
* The system shall provide access to online classes, practice tests, and track progress (test name, time, score, status: not taken/in progress/failed/passed).
* The system shall display driver notes and lesson schedules in a table format (start/end times, comments).
* The system shall integrate with the DMV to receive updates and notifications on rules, policies, and sample questions.
* The system shall generate activity reports showing who made/canceled/modified reservations, with options to print or export.
* The system shall support secretary/phone-based scheduling and include a contact form for users.
* The system shall allow the IT admin to reset passwords, block access, and manage all accounts.
* The system shall log all actions for auditing purposes and notify on DMV changes.

### 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: A clean, intuitive web based UI with role specific dashboards; mobile responsive for on the go access.
* Users: Customers need to log in, view progress/tests, schedule lessons, and contact support; Owner needs reports and package management; IT needs account maintenance; Secretary needs appointment entry/views.
* Interactions: Browser based; forms for registration/scheduling, tables for progress/notes, buttons for actions (e.g., cancel, reset password); sketch like pages for test progress and lesson tables as described by Liam.
* Accessibility: Simple navigation, search for availability, notifications for updates; cloud hosted for 24/7 access.

### 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 online features like scheduling and accessing practice tests; lack of internet would limit functionality but was not detailed.
* The DMV provides a functional API for real time updates on rules and questions; integration assumes compatibility with this external system.
* Customers possess basic computer literacy and a valid email address for account registration and password resets, which was not explicitly confirmed.
* The initial fleet of 10 cars and drivers is sufficient to handle the expected user load; scalability beyond this was not addressed.
* Credit card processing assumes integration with a secure payment gateway, though specific provider details were not discussed.
* The cloud provider ensures 99.9% uptime and handles backups/security, aligning with the client’s desire to minimize technical management.

### 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 depends on internet connectivity; offline access is limited to downloaded reports with no modifications allowed.
* Custom package creation beyond the initial three (6/8/12 hours) requires developer intervention, not supported in the initial design.
* No real-time on-road tracking is included, limiting visibility during lessons.
* Resource constraints include budget assumptions covered by the client and a timeline ending May 10, with no buffer for delays.
* Technology limits include reliance on DMV API availability and potential scalability issues if user demand exceeds 10 cars/drivers without expansion.
* No voice activation or multilingual support is included, potentially reducing accessibility for some users due to scope or budget 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.*

