# 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?*

* DriverPass wants to provide students with online practice exams and on-the-road training.
* The system should allow customers to register, purchase packages, schedule lessons, and track progress.
* The client wants cloud-based access from any device.

### 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?*

* Problem: 65% of students fail DMV tests due to poor preparation.
* DriverPass wants to fill this gap with a comprehensive training system.
* Needs: scheduling, reporting, DMV compliance updates, secure payments, and role-based access.

### 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 24/7 access to online tests and study materials.
* Enable lesson scheduling with drivers and cars.
* Track student progress and instructor notes.
* Ensure security, reliability, and DMV compliance.

## Requirements

## Nonfunctional Requirements

* **Performance**: Web-based, responsive (<2s load time), monthly updates.
* **Platform Constraints**: Runs on Windows/Linux servers, uses relational DB, integrates with payment gateways.
* **Accuracy/Precision**: Unique logins, input validation, activity tracking.
* **Adaptability**: Admins can add/remove users, disable packages, system adapts to browser/OS updates.
* **Security**: SSL/TLS encryption, role-based access, account lockout after failed attempts, password reset via email.

#### 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 run as a **web-based application** accessible through modern browsers (Chrome, Edge, Safari, Firefox).

 It should also be **mobile-friendly**, allowing access from smartphones and tablets without requiring a separate app at launch.

 The system should respond to user actions (such as login, scheduling, or test submission) in **under 2 seconds** under normal load.

 The system must support **concurrent users** (students, instructors, administrators) without performance degradation.

 Updates and maintenance should occur on a **monthly basis**, with emergency patches applied as needed for security or compliance.

 Scheduled maintenance should be performed during **off-peak hours** to minimize disruption.

 The system should maintain **99% uptime** to ensure reliability for students preparing for their DMV exams.

#### 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 run on **Windows and Linux servers** to provide flexibility in hosting environments.
* It must be **cloud-based**, ensuring scalability, reliability, and reduced need for local IT maintenance.
* The backend requires a **relational database management system (RDBMS)** such as **MySQL, PostgreSQL, or SQL Server** to store user accounts, reservations, packages, and test results.
* The system must support integration with **third-party payment gateways** (e.g., PayPal, Stripe) for secure transactions.
* The application should be accessible via **modern web browsers** (Chrome, Edge, Safari, Firefox) without requiring additional software installation.
* The design should allow for **future integration with DMV systems** to receive updates on rules, policies, and test content.

#### 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?*

• Each user will be distinguished by a unique username or email address to prevent duplicate accounts.

• Passwords and login credentials will be case-sensitive to ensure security and accuracy.

• Input validation will be enforced for all fields (e.g., names, addresses, credit card numbers, scheduling times) to reduce errors.

• The system will log all user actions (e.g., reservations, cancellations, modifications) with timestamps and user IDs for accountability.

• The system will notify administrators when:

* Multiple failed login attempts occur (possible brute force attack).
* Scheduling conflicts arise (e.g., double-booked car or instructor).
* Payment processing errors or declined transactions occur.
* DMV updates fail to sync or cannot be applied.

Reports will be generated to track who made changes, when they were made, and what was modified, ensuring traceability.

#### 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?*

 Administrators must be able to **add, remove, and modify user accounts** (students, instructors, secretaries, and admins) through the system’s interface without requiring code changes.

 The system should allow administrators to **enable or disable training packages** without developer intervention.

 The system must be designed to **adapt to browser and operating system updates** with minimal reconfiguration, ensuring continued compatibility with modern platforms.

 The architecture should support **scalability**, allowing new features or modules (e.g., additional training packages or reporting tools) to be added in future releases.

 IT administrators require **role-based access control**, including:

* Full access to manage accounts and reset passwords.
* The ability to block or revoke access for terminated employees.
* Permissions to configure system settings, monitor logs, and apply updates.

 The system should support **cloud-based deployment**, reducing the need for manual maintenance and ensuring automatic updates for security patches 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 be required to log in with a **unique username or email address and a secure password**.

 All data exchanges between the client and the server will be protected using **SSL/TLS encryption** to ensure confidentiality and integrity.

 Passwords will be stored in the database using **secure hashing and salting techniques** to prevent exposure in case of a breach.

 Accounts will be **locked after five consecutive failed login attempts** to prevent brute force hacking attempts. Locked accounts can only be reset by the IT administrator or through a secure password reset process.

 If a user forgets their password, the system will provide a **self-service password reset option**:

* The user will request a reset.
* The system will send a **verification link or code** to the registered email address.
* Once verified, the user can create a new password that meets security requirements (e.g., minimum length, complexity).

 Administrators will receive **alerts of suspicious login activity**, such as repeated failed attempts or logins from unusual locations.

 Role-based access control will ensure that users only have access to the features and data appropriate to their role (student, instructor, secretary, administrator).

### 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 students to register for an account and provide required personal information.
* The system shall allow students to purchase training packages securely online.
* The system shall allow administrators to enable or disable training packages.
* The system shall allow students to schedule, reschedule, and cancel driving lessons.
* The system shall ensure that each scheduled lesson is linked to a specific student, instructor, car, date, and time.
* The system shall prevent scheduling conflicts by checking instructor and car availability.
* The system shall allow instructors to view their assigned schedules and update their availability.
* The system shall allow instructors to add notes and comments after each lesson.
* The system shall provide students with access to online practice tests.
* The system shall automatically grade practice tests and display results (score, time taken, status).
* The system shall track student progress, including completed lessons, test results, and status updates.
* The system shall generate reports for administrators showing activity logs, reservations, cancellations, and modifications.
* The system shall log all user actions (create, modify, cancel) with timestamps and user IDs.
* The system shall integrate with DMV updates to receive new rules, policies, and test questions.
* The system shall send email confirmations for registrations, payments, and scheduled lessons.
* The system shall allow administrators and IT staff to reset user passwords and block accounts when necessary.
* The system shall provide a self-service password reset option for users who forget their credentials.
* The system shall process payments securely through a third-party payment gateway.
* The system shall allow administrators to export reports and data into formats such as Excel for offline use.

### 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.)?*

 General Needs

* The interface must be intuitive, user-friendly, and accessible to users with varying levels of technical skill.
* It should be browser-based and optimized for both desktop and mobile devices.
* The design should be responsive, adapting to different screen sizes such as laptops, tablets, and smartphones.
* Navigation should be clear and consistent, with menus and dashboards tailored to each user role.

 Students

* Register for an account and log in securely.
* Purchase training packages and view available options.
* Access online practice exams and study materials.
* Schedule, reschedule, and cancel driving lessons.
* View progress reports, test scores, and instructor feedback.
* Receive email confirmations and notifications through the interface.

 Instructors

* Log in securely with role-based access.
* View their assigned schedules and availability.
* Update availability by marking times as open or unavailable.
* Access student lesson details and add notes after each session.
* Communicate updates or feedback to students through the system.

 Secretary

* Input customer information for students who register by phone or in person.
* Schedule, reschedule, or cancel lessons on behalf of students.
* Access student records and package details to assist with customer service.
* Generate simple reports or summaries for management.

 Administrators and IT Staff

* Manage user accounts by adding, removing, or modifying students, instructors, and secretaries.
* Reset passwords and unlock accounts when needed.
* Enable or disable training packages.
* Monitor system activity logs and generate detailed reports.
* Configure system settings and ensure compliance with DMV updates.

 Interaction Methods

* Primary access will be through a web browser such as Chrome, Edge, Safari, or Firefox.
* The interface must be mobile-friendly, allowing students and instructors to use smartphones or tablets.
* Email notifications and confirmations will be integrated with the system to support communication outside the interface.

### 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?*

 Students will have access to a stable internet connection and a modern web browser.

 Students will have basic computer literacy to navigate the system without extensive training.

 Instructors and secretaries will be trained on how to use the system before it goes live.

 Payment processing will be handled by a secure, third-party provider (e.g., PayPal, Stripe).

 The DMV will provide timely updates to rules, policies, and test content that can be integrated into the system.

 DriverPass will provide the necessary resources (cars, instructors, and staff) to support the scheduling system.

 The system will initially launch in English, with the assumption that most users can read and understand it.

 Users will provide accurate and truthful information when registering and scheduling lessons.

 The hosting environment (cloud servers) will be reliable and scalable to handle expected traffic.

 Customer support will be available to assist users with login issues, scheduling conflicts, or payment errors.

### 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?*

 Budget constraints may limit the inclusion of advanced features such as a dedicated mobile app at launch; the initial release will focus on a browser-based solution.

 Time constraints may restrict the number of practice test variations and lesson package options available in the first version of the system.

 The system’s performance will depend on server capacity and internet speed, which may affect users in areas with poor connectivity.

 Limited IT staff at DriverPass may delay troubleshooting, updates, or system maintenance.

 Integration with DMV systems may be limited by the availability and timeliness of DMV-provided updates.

 The system will initially support only English, which may limit accessibility for non-English-speaking users.

 Customization of training packages (e.g., creating personalized lesson plans) will not be available in the initial release and may be considered for future updates.

 The system will rely on third-party services (such as payment gateways and cloud hosting), which introduces dependency risks if those services experience downtime or changes.

### 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 project

AI-generated content may be incorrect.*