# CS 255 Business Requirements Document

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 design a comprehensive system for DriverPass, a company aiming to provide effective driver training solutions.
* The client, DriverPass (represented by Liam and Ian), wants the system to support both online and in-person training for driving tests.
* The system must allow users to:
  + Take practice tests online
  + Enroll in on-the-road driving sessions
  + Reserve lessons
  + Access their training progress
  + Communicate with DriverPass staff
* The client wants secure, role-based access for administrators, IT, and customers.
* The system must work online and allow some downloadable content (e.g., reports).

### 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 the system to help students better prepare for their DMV driving exams through:
  + Online practice tests
  + On-the-road training sessions
  + DMV rule instruction modules
* The core problem DriverPass aims to solve is the high failure rate (65%+) among students who rely solely on memorizing previous test questions.
* The system must provide:
  + A flexible and scalable training package structure (3 levels of packages with potential for future customization)
  + A scheduling and reservation system for driving lessons
  + An admin and IT interface for user and account management
  + Customer-facing web pages for account access, booking, and training tracking
  + Tracking and reporting tools for user activity and system logs
  + Integration with DMV systems for real-time updates on rules and policies

### 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 allow students to:
  + Register for training packages online or via phone
  + Schedule, reschedule, or cancel driving lessons
  + Access practice tests and track their progress
* The system should allow DriverPass staff to:
  + Create and manage training packages
  + Assign students to instructors, cars, and time slots
  + Generate and print reports (activity logs, test scores, scheduling)
  + Track user actions (who scheduled, modified, or canceled lessons)
* The system should allow IT admins to:
  + Manage user roles and permissions
  + Reset passwords and deactivate accounts
  + Ensure system security and data integrity
* The system should:
  + Be accessible via web and mobile devices
  + Sync with DMV updates and notify users of changes
  + Store and export data (e.g., to Excel)
  + Provide a scalable backend for future feature expansion

## 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 run as a cloud-based web application accessible via desktop and mobile browsers.
* The system should load all standard pages within 3 seconds under normal conditions.
* Online practice tests and scheduling pages should respond in under 2 seconds.
* Updates should be rolled out on a monthly basis or sooner if critical DMV changes are received.
* The system must support simultaneous access for at least 500 users without degradation.

#### 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 run on a cloud platform (e.g., AWS, Azure, or Google Cloud) with high availability.
* The backend should support a relational database (e.g., MySQL or PostgreSQL).
* The system must be compatible with all major browsers (Chrome, Firefox, Safari, Edge).
* It should be mobile-responsive to support Android and iOS browser access.
* Development tools may include HTML/CSS/JS (frontend), Node.js or Java (backend), and REST APIs.

#### 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 distinguished by unique usernames (email-based login) and role-based permissions.
* Inputs such as usernames and passwords will be case-sensitive for security.
* System will log all user actions (e.g., reservations, modifications, deletions).
* Admins will be notified of failed login attempts, data integrity issues, and system errors via dashboard alerts and email notifications.

#### 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, disable, or edit training packages without needing code changes.
* The system will allow modification of user roles and assignments through the admin interface.
* Platform will be hosted in a cloud environment that automatically adjusts to traffic and supports OS/browser updates.
* IT admins have full control over system settings, user access, and package availability.

#### 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 authenticate using email and password; two-factor authentication is optional for admins.
* Password recovery/reset is automated through email verification.
* All data exchanged will be encrypted using HTTPS and TLS protocols.
* Role-based access control (RBAC) will be implemented for system features.
* If multiple failed login attempts are detected (e.g., brute-force attack), the account will be locked and admins notified.

### 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.”*

* Here are the functional requirements for the DriverPass system:
  + The system shall allow users to create, update, and delete accounts.
  + The system shall authenticate users with a secure login process.
  + The system shall allow customers to view, schedule, reschedule, and cancel driving lessons online.
  + The system shall track lesson bookings, cancellations, and modifications, with user audit logs.
  + The system shall allow secretaries to manually enter customer appointments via phone or walk-in.
  + The system shall store and display student data, including name, address, phone number, and preferred pickup/drop-off location.
  + The system shall assign students to instructors, vehicles, and time slots for each session.
  + The system shall allow users to register for one of three training packages.
  + The system shall allow administrators to enable/disable training packages without code modification.
  + The system shall allow instructors to enter comments and session feedback for each student.
  + The system shall track online test progress, showing test name, time taken, score, and status (not taken, in progress, failed, passed).
  + The system shall generate downloadable reports for user activity, scheduling, and test results.
  + The system shall allow automatic password resets via email verification.
  + The system shall notify users and staff of DMV updates to rules, policies, or sample questions.
  + The system shall allow customers to contact support through a contact form.

### 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, cloud-hosted, and mobile-responsive for both customers and staff.
* **User Types and Their Needs:**
  + **Customers (students):**
    - Register and log in to their accounts.
    - View available training packages and book appointments.
    - Access and track their online test progress.
    - View lesson schedules and instructor feedback.
    - Modify or cancel appointments.
    - Reset passwords via email.
    - Contact DriverPass support.
* **Secretary:**
  + Search for customer records.
  + Book, cancel, or reschedule lessons on behalf of customers.
  + View and edit customer profiles.
  + Print reports as needed.
* **Instructors (Drivers):**
  + View assigned students and lesson schedules.
  + Enter feedback and session notes for each student.
  + Access training package information.
* **Admin/IT Officer:**
  + Manage all user accounts and roles.
  + Reset passwords and block/deactivate users.
  + Enable/disable training packages.
  + Monitor system usage, audit logs, and receive alerts.
  + Update system configurations and security settings.
* **Interface Features:**
  + Clean navigation with role-specific dashboards.
  + Visual representation of test progress (e.g., status indicators: Not Taken, In Progress, Passed, Failed).
  + Forms for inputting personal details, appointment details, and package selection.
  + Contact form for customer support inquiries.
  + Export options (e.g., Excel downloads for reports).

### 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?*

* It is assumed that all users (customers, staff, admin) will have reliable internet access when using the online system.
* It is assumed that the system will only be used within the United States, complying with DMV standards relevant to U.S. regulations.
* It is assumed that users have basic computer literacy and access to a smartphone, tablet, or computer.
* It is assumed that the company will continue to maintain and update training packages and content regularly.
* It is assumed that payment handling will be conducted securely through a third-party payment processor (e.g., Stripe or PayPal).
* It is assumed that the DMV will have a digital method (API or portal) for providing rules, updates, or test content.
* It is assumed that DriverPass has access to cloud hosting services and basic IT infrastructure to support deployment 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 cannot support offline data editing or synchronization due to risks of data redundancy and conflicts.
* Customization of training packages (adding new modules or altering logic) will require developer intervention in future updates.
* Budget and time constraints may limit advanced features like live chat support, AI-based recommendations, or advanced analytics in the initial release.
* System scalability is initially designed for a small-to-medium user base; performance tuning may be needed if the company rapidly expands.
* Integration with DMV systems may be limited by the availability or compatibility of DMV APIs.
* The interface design may need refinement over time based on user feedback, but initial design is based on the client’s rough sketch.
* The project timeline is fixed and assumes no major change requests or scope creep from the client after initial approval.

### 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 graph showing a bar graph

AI-generated content may be incorrect.*