# 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 design and develop a system for DriverPass.
* DriverPass aims to improve driver training and help students prepare for their DMV driving tests
* The system should allow students to:
  + Take online practice tests.
  + Schedule on-the-road training lessons.
  + Manage their reservations.
* The system should enable administrators and employees to:
  + Manage student accounts.
  + Schedule lessons.
  + Track reservations efficiently.

### 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 has identified a significant gap in driver training, as many individuals struggle to pass their DMV driving tests.
* Their solution includes:
  + Offering comprehensive training through online courses, practice tests, and on-the-road driving lessons.
  + Providing students with an online platform to register, schedule lessons, and track progress.
  + Allowing administrators to manage accounts, appointments, and security settings.
  + Ensuring compliance with DMV updates by integrating notifications of policy changes.

### 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 be able to
  + Enable students to create accounts, schedule driving lessons, and access online course materials.
  + Allow students to take practice tests and track their progress, displaying test results and statuses.
  + Provide a scheduling system for on-the-road lessons, ensuring coordination between students, instructors, and available vehicles.
  + Ensure that administrators and employees have role-based access to manage student accounts, schedule lessons, and track appointments.
  + Allow the IT administrator to manage user accounts, reset passwords, and restrict access when necessary.
  + Track all system activities and changes to appointments, ensuring accountability and transparency.
  + Integrate with DMV regulations to receive notifications about updates to driving tests and policies.
  + Run on a web-based cloud platform for accessibility and ease of maintenance.
  + Provide a secure login system for students, instructors, and administrators to protect user data.

## 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 DriverPass system must be a web-based platform that operates smoothly across modern internet browsers and is optimized for both desktop and mobile environments. Key systems actions, like logging in, accessing tests, scheduling lessons, and viewing progress, should load withing 2 seconds under normal usage conditions. The system should be available 24/7, with 99.9% uptimes to support flexible scheduling and access from various times zones. Updates to course materials, user interface elements, and backend features should be rolled out on a quarterly basis, or more frequently if security or performance patches are required.

#### 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 DriverPass System must be platform independent and fully accessible through modern web browsers (e.g., Chrome, Firefox, Safari. Edge) on both Windows and macOS operating systems, as well as mobile platforms such as iOS or Android. Th system will be web-based, eliminating the need for users to download or install software. On the back end, the system will require a relational database management system (RDBMS) to support storage and retrieval of user data, test results, schedules, and payment records. It must also integrate with third-party APIs for secure payment processing (e.g., Stripe or Shopify) and may use cloud-based hosting for scalability and reliability.

#### 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 must distinguish between different users through unique login credentials, including a valid email address and a secure, case-sensitive password. Role-based access will be implemented to differentiate students, instructors, and administrators, ensuring that each user type has access only to relevant features and data. Input fields such as login, payment details, and practice test answers must be validated in real-time to reduce errors. All test scoring must be calculated precisely and store with timestamps for audit purposes. The system should automatically notify administrators of issues such as failed payment attempts, scheduling conflicts, test scoring anomalies, or user access errors. These alerts should be flagged in the admin dashboard and optionally forwarded via email or internal messaging.

#### 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?*

* The system must support user management and administrative interface that allows authorized IT admins to add, remove or modify user accounts without modifying the codebase. User roles and permissions should be adjustable through configuration settings or a web-based control panel. The platform must be designed to adapt to operating system and browser updates by using responsive design principles and keeping all dependencies (e.g., frameworks, libraries) current. Regular maintenance updates should ensure compatibility with evolving web standards. IT administrators will require full backend access to user management, reporting tools, audit logs, and system configuration settings. They should also have the ability to initiate data backups, manage integrations, and control system access permissions for troubleshooting and security.

#### 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 using a valid email address and a secure, case-sensitive password. Passwords should be encrypted using modern hashing algorithms (e.g., bcrypt or SHA-256), and login sessions should be authenticated via HTTPS to ensure all data exchanged between the client and server is secure and encrypted. To protect against brute force attacks, the system should implement rate limiting and temporarily lock the account after five failed login attempts, requiring either a cooldown period or administrator intervention to unlock. Users must also complete a CAPTCHA challenger after multiple failures to further prevent automated attacks. If a user forgets their password, they should be able to reset it through a secure “forgot password” flow, which verifies their identity through email authentication before allow password changes. All password reset tokens should be time-limited and single-user to prevent misuse.

### 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 should validate user credentials when logging in.
* The system shall allow students to create and manager user accounts, including personal, billing, and contact information.
* The system shall allow students to schedule driving lessons with available instructors.
* The system shall provide access to online practice exams and automatically grade them.
* The system shall track student progress through lessons and practice test performance.
* The system shall process credit and debit card payments via a third-party vendor.
* The system shall send automated notifications to students confirming transactions. Schedule updates, and test records.
* The system shall allow administrators to view and manager user accounts, driving lesson schedules, and test records.
* The system shall allow instructors to update availability and view their scheduled driving lessons.
* The system shall log all transactions and test attempts for record keeping and analysis.

### 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 DriverPass system will feature a web-based interface that is fully responsive and accessible via modern web browsers on both desktop and mobile devices. The interface must be user-friendly, intuitive, and accessible to users with varying levels of technical experience. All users will interact with the interface through a secure browser session, and all actions will be protected by login authentication and encryption.

There are three primary user typers for the interface:

* **Students**: Students need to be able to create and manage their accounts, take practice exams, schedule on-the-road training sessions, view progress reports, and make secure payments. The interface should include dashboards for tracking performance and upcoming sessions, as well as notifications and account settings
* **Instructors:** Instructors require access to view and manage their schedules, update lesson availability, and view student progress for upcoming sessions. Their interface should be streamlined to focus on calendar management and communication with students or admins.
* **Administrators:** Admins will use the interface to oversee all system operations. This includes managing users, updating course materials, configuring scheduling rules, viewing system logs, processing support requests, and performing basic troubleshooting. Their interface should provide access to analytics and management tools while maintaining a secure and role-restricted view.

### 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 users will have reliable internet access and access to a modern web browser on either desktop or mobile device.
* It is assumed that DriverPass will use an existing third-party payment processing service that supports secure and compliant transactions.
* It is assumed that instructors and administrators have basic digital literacy and will require minimal training to user the interface.
* The design assumes that the platform will be hosted on a cloud-based environment with built-in scalability and backup services.
* It is assumed that users will be responsible for maintaining their account information, including contact details and scheduling preferences.

### 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 will rely on external payment vendors, which may introduce limitations in terms of customization, transaction fees, or downtime outside of DriverPass’s control.
* The current design assumes a standard level of internet connectivity, which may not accommodate users in rural or low-bandwidth areas.
* There may be budget constraints that limit the ability to implement advanced features such as AI-driven performance feedback or multilingual support in the initial release.
* Time and staffing limitations may impact how quickly new features (like expanded scheduling tools or advanced analytics) can be developed and tested.
* The system will initially be developed as a web-based platform, which means that offline access to materials or scheduling tools will not be available in the first version.

### 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.*

