# CS 255 Business Requirements Document Template

Roger Fisher 7/30/2025

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?*

* Client Name: DriverPass is a startup company that offers a more modern and effective training model on how individuals prepare for their DMV driving exams.
* Training Methods: The company offers progressive training packages, which consist of online learning classes, DMV test simulation exams, and in-car behind-the-wheel lessons.
* System Objective: The proposed system must facilitate the entire training process by enabling customers to establish accounts, purchase a training package, book classes, and monitor their progress on the online evaluation tools and testing resources.
* Roles and Access Management: The system should have multiple user roles, such as customers, secretaries, IT administration, and company owner, and be able to manage specific account related roles, schedules, reporting, and support.
* Business Purpose: The system will centralize and streamline the activities of driver training, making it more accessible, well organized, effective for the customers and less demanding when handling internal processes.
* Long-Term Vision: To have a scalable, user-friendly product that helps to increase pass rates, customer satisfaction, and translate DriverPass into a leader in contemporary driver education.

### 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 Statement: A lot of people fail their driving exams because of a lack of preparation, the inability to access realistic practice exams, and difficulty finding time to have in person classes. To resolve this problem, DriverPass intends to provide a platform that incorporates contemporary online learning technologies and traditional driver training.
* System Purpose: DriverPass needs a system with the combination of online and face-to-face training to produce a comprehensive and convenient training environment. This combination method allows students to learn at their own pace and still have the opportunity of a one-on-one lesson seat time.
* Customer Access: For students, they should be able to purchase a portfolio of three training packages each with varying lesson hours and access to DMV style mock exams. Convenient registration, online account management, progress monitoring via practice tests and completed lessons should all be offered within the system.
* Scheduling and Tracking: Customers should be able to schedule lessons online or by phone in addition to tracking which instructor, vehicle, and time slot has been assigned to each student. It must also record the user activity including who made, changed, or cancelled the booking to ensure accountability.
* Employee Tools: Secretaries must be able to make appointments and access customer records and IT staff need to be able to gain administrative access to reset passwords and update accounts as well as managing user access.
* Cloud and Reporting: This is the connection to the cloud so that information is available in real time on any mobile device. It must also enable employees to create reports, export data (such as Excel files), and operate offline when needed.
* DMV Integration: DriverPass would like the system to be regularly updated by the DMV whenever sets of new regulations or new test questions are published.

### 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?*

* User Account Management: The system should enable customers to create their personal accounts, change their log-in credentials, or retrieve a lost password and book, reschedule or cancel the appointment on their own, without further assistance. Such accounts must also act as windows through which the user may monitor training progress, lesson histories and test results.
* Employee Functionality: The secretarial personnel should be able to input customer details to book them in the training packages as well as manually do bookings by phone or in person meetings. The administrator should also have access to view lesson history and student progress when required to assist the administrator.
* Progress Tracking: The system must be able to monitor who is accessing the practice exams and report the critically important data including scores, time taken and the status (pass or fail). The instructor must also have the opportunity to add notes to each lesson and the history of the training session of each student.
* Activity Logging: To allow accountability, any activity involving the creation, addition or cancellation of any appointment should be recorded, including the user who acted on that operation.
* Secure Data Handling: The system should have secure entry and storage of sensitive data like personal information; credit card payments and student pick up location. There should be adherence to all best practices of data privacy and compliance.
* Package Management: The administrators must be in a position to add training packages that are made available through the system and withdraw and delete packages that have become obsolete and are no longer used. It should also involve flexibility in changing package contents or prices in response to the changing needs of the business.
* Cloud Deployment and Device Access: To maintain the ability to access it on most devices, the platform is to be deployed onto a cloud-based platform with a minimal downtime threshold. Whether they are remote or within the office, employees should have a stable interface.
* DMV Updates Integration: This is so that as the DMV policy changes, and test updates will be integrated in real time. New rules or questions sets need to be presented in training material and mock exams as it comes out.
* Future Expansions: When creating this platform, it should be optimized to grow and allow the DriverPass team to add new types of training, user types, or seamless connections with third party tools without having to redesign the entire system.

## 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?*

* Platform Environment: The DriverPass system should be able to work in a web environment which can be accessed through all the mainstream browsers (Chrome, Firefox, Safari, Edge) and successfully be adapted to both desktop and mobile environments. It must be compatible with responsive design that would make it able to be used on multiple screen sizes and operating systems such as Windows, macOS, iOS, and Android.
* System Speed: Main pages like dashboards, booking tools, and training modules, among others, should be loaded within three seconds even under typical broadband conditions. Database queries accounting for 95 percent of transactions must be accomplished within two seconds in terms of fetching lesson histories or appointment information.
* Availability and Uptime: The system requires a 99% or beyond uptime standard per month to make it readily available to customers and employees. Schedules and scheduled application downtimes along with unscheduled downtime needs to be communicated to the administrative users early and in as limited frequency as possible around peak access periods.
* Scalability: It is required that the system be designed to scale horizontally to support increased traffic of users without loss of performance. Since DriverPass is expected to increase, the system should be capable of handling thousands of simultaneous users with the ability to be accessed by students, instructors and staff.
* Data Synchronization and Cloud Availability: The system should be able to reliably keep data in real time synchronized with cloud storage where people in various locations have access to them. Whatever a user carries out, including making a reservation or taking a mock test, should be reflected immediately on all the systems and devices.
* Update Frequency: At least once every quarter, the system must be updated with new functions, patches, and compatibility with DMV changes to tests. Any critical security updates or DMV content updates should be released or communicated within 48 hours followed up by deployment within 48 hours of release.

#### 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?*

* Supported Operating Systems: The DriverPass system should be compatible and work with some of the most modern operating systems like Windows, macOS, iOS and Android. The web-based app must work in all these settings with any popular browser like Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge. This is the only legacy browser support requirement other than basic rendering.
* Server Environment: A Linux server is better with the back-end server than a Windows platform because it is compatible, has better performance and can be scaled easily. The system ought to be capable of being deployed in a cloud environment, including AWS, Microsoft Azure, or GCP, so that it can horizontally scale and utilize a managed service.
* Database Needs: The core functionalities like user management, lesson schedules, progress tracking, payment records etc., will require a relational database like MYSQL or PostgreSQL. Such a database must be multiuser constrained and ensure consistency of ACID transactions.
* Middleware and APIs: The system needs to have connectivity with third-party APIs to include DMV data updates, email/SMS notifications, and payment gateways in the online system. RESTful API standards are to be adhered to in order to consider maintainability and interoperability.
* Front End Compatibility: The front end should be developed observing the principle of responsive designing to make it compatible with all machines (desktops, laptops, tablets, and smartphones). Touch input and simplified UI should be also taken into consideration when mobile users are set as priority.
* Development and Language: The system may not have strict requirements as far as the specific programming language required, but the system must be written using languages that are widely supported like JavaScript (front end), Java or Python (backend), and arranged using up to date structural frameworks. The tooling must support interacting with Git-based version control and CI/CD pipelines to support the continuous independence of the development and deployment.

#### 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?*

* User Role Identification: The system should not confuse the identity of various user types, such as customers, instructors, secretaries, administrators and the owner, on the basis of their log-in identification and the profile associated with the role. RBAC must be implemented in such a way to maintain that no user or workers should have access to the tools and data that are unfitting to their job description, scheduling features should be available to secretaries only, and progress reports to instructors only.
* Input Verification and Case Sensitivity: System inputs should also be verified and followed by strict regulations that are case-sensitive. String variables like the username, email, or password must be case sensitive to keep the site secure. In the other input fields (names, city names), user-friendly search and data retrieval should also be provided allowing case-insensitive matching. It also has to impose input constraints on the formats and lengths and required fields (phone numbers should be numeric value, pickup locations should follow address format).
* Error Detection and Alerts: The system must automatically determine anomalies in case of conflicting appointments, duplicate registrations, in-complete data submissions or unauthorized accesses. Such errors ought to activate a real-time alert system that informs administrators on their dashboard and alternative through email address or SMS, as appropriate.
* Audit and Logging: Every process that alters the state of the system (including account creation, lesson booking, password resets and access revocation) should be logged with a timestamp and user ID that executed it. This will make it traceable and will also allow determining the possibility of misuse or technical errors.
* Threshold Alerts: The system must alert the administrators when certain limits are reached or surpassed. On this example, should availability of appointments be less than a certain number of slots daily, or should there be a number of unsuccessful logins that may indicate a brute force hacking attempt, then the system has to create an alert message to the administration dashboard so that an administrator can look into it.

#### 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?*

* Flexibility in User Management: The system ought to be dynamic in its ability to receive changes in user accounts via some sort of administrator level access so that IT staff can add or remove users or change generated permissions but not necessarily touch the underlying codebase. The role assigning to be done via a configuration panel which is flexible in terms of giving control access as per the business requirement.
* Compatibility with Platform Update: The system design should consist of a modular architecture and industry standard frameworks because it will enable future platform changes. It should have minimal effect on upgrading operating system, browser, or devices and therefore front-end and back-end components should be loosely coupled.
* Role-Based Access Control: RBAC should achieve configuration where administrators can change the role permission or produce additional types of user roles as the organization structure changes. That way, adaptability is ensured in the long term without having to redevelop the software or involve engineers.
* Management Roles and Privileges: The role of IT administrator must have high privileges with a secure admin console. These privileges should enable them to control user access, user password reset, system settings (including the triggering of notifications, data retention policies), among other things, activate or deactivate features as needed by the business.
* New Features Scalability: The system should be constructed in such a way that it may be extended in the future so that the platform can accommodate future needs. Any new module or set of integrations (new DMV APIs, greater training, or reporting dashboards, etc.) must be able to be installed in the form of a plug-in or through an API and not result in downtime or a rebuild of the system.

#### 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?*

* Authentication requirements to log in: The user will have to log in with a unique and case-sensitive username or email address and a password that complies with the complexity requirements (such as a minimum of 8 characters, a number, an uppercase, and a symbol). The administrators and staff could require multi-factor authentication to make the account more secure.
* Encrypted Data: The whole client-server communication must be through HTTPS with TLS 1.2 or later. AES-256 encryption standards should be used to encrypt sensitive data (which include passwords, credit card information, personal information) both during transmission and at rest. The session tokens ought to be kept in a safe place and time out after a specific idle interval.
* Brute Force Protection: To counter brute force attacks, the system should be capable of identifying and preventing such attacks by locking an account once five consecutive failed attempts are made. The user will have to wait 15 minutes and be automatically reactivated or call support to help a user manually after locking. The dashboard should alert the administrators when several attempts at brute force are taking place on several accounts within a specified period of time.
* Forgotten Password Recovery: Users that have forgotten their password are supposed to undergo a secure process of resetting the password. This covers the confirmation of their identity using a registered email address or phone number and a time stamped reset link. The links to reset the password should expire in 15-30 minutes and should count only once.
* Oversight by the Administration: Each attempted log in as well as the unsuccessful authentication and password changes should be recorded with a time stamp, user ID, IP address and type of device. This is a log which should be made available to IT administrators in audit and security checks. Abnormal behavior like frequent logins to unfamiliar places should be exercised by verification or restricted access.

### 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 ensure verification of user login such as case sensitive usernames/emails as well as encrypted passwords.
* The system shall enable the users to create, update, and delete their accounts where sensitive changes are verifiable.
* The system shall enable the customer to buy one of three packages of training which entails varying services.
* The system shall enable customers to book, reschedule or cancel driving lessons online or phone.
* The system shall monitor the lesson history of each student, the practice test results, and the level of progress (pass or fail)
* The system shall be used to allocate students to the available instructors, vehicles, and time slots according to their availability.
* The system shall record any activity of the appointment work such as those who created it, those who edited it and those who cancelled it.
* The system shall enable secretaries to have access to a special interface to add new data about customers and appointments.
* The system shall enable teachers to append notes to the lessons of the students and to show progress reports.
* This system shall have the ability to offer password reset, role modification and access revocation to the administrative users.
* The system shall be connected to DMV APIs to get updates in real time for rules, policies, and test questions.
* The system shall be able to export the reports easily go offline like excel.
* The system shall provide alerts in the event of an activity in the account, an appointment, or any update in the DMV.
* The system shall keep records of all login attempts, password changes and any operations or critical activities.

### 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.)?*

* User Access Points: The DriverPass platform will have a web browser interface that will integrate with the latest browsers like chrome, Firefox, safari, and edge. The interface should be dynamic and applicable with full functionality of both the desktop, tablets and the mobile devices. Mobile users require the optimization of touch and clean navigation and scalable elements.
* Customer Interface Needs: Customers need the facility to create accounts, select the training packages, and make and track bookings, ways to schedule driving lessons, read the lessons and practice exam results and update information about the customer. It must have a clean and user-friendly interface and dashboards indicating progress in an easy-to-understand manner and supplying access to the means of scheduling at the same glance.
* Secretary Interface Needs: The secretaries should have access to records about the customers, be able to manually enter the bookings received either by phone or in a face-to-face manner and manage the schedules of the instructors. They will also create reports and avail communication facilities where confirmations and reminders can be sent on the appointments.
* Instructor Interface Needs: The instructors will need to see their teaching schedule, have access to details of the students and be able to add notes to each lesson. There should be a sleek calendar-like view of future appointments and one-click student profiles. They should also be capable of putting their availability into the scheduling system.
* Admins and IT Interface Needs: Administrators must have a secure interface where they can perform system configuration, user role and permission, password reset, account deactivation, and security log monitoring. Offers DMV updates, system health status, and alerts should also be available on the interface.
* User Interaction Requirements: All the user roles must use the system via user-friendly dashboards customized according to their duties. Forms, buttons and navigational paths should be uncomplicated and should not require learning. Error processing and confirmation messages as well as visual consistency requirements should be provided to help the users and avoid missteps.

### 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?*

* User Technology Readiness: All the end-users, including customers, secretaries, instructors, and administrators, are expected to have access to the stable internet connections and modern devices (smartphones, tablets, or desktop/laptop computers) that could be used to access updated versions of the major browsers, such as Chrome, Firefox, Safari, or Edge.
* Basic Digital Literacy: It is also hoped that users possess basic digital literacy in the shape of familiarity with navigating a site, email usage, and filling out web forms. Customers are not going to require training, but the secretarial and administrative personnel may be offered one.
* Third-Party API Availability: The assumption made is the continued availability of external services which include DMV data APIs, email and SMS service providers and payment gateways which will continue to be available, functioning and accessible on secure endpoints.
* Content Consistency from the DMV: The system presumes that the DMV will be publishing any changes in rules and questions on a test with consistency that allows either an automated update or a semi-automated one via an API or manual upload.
* Scalability Demand Projection: The assumption surrounding such Demand Projection is that according to the growth of the user base of the DriverPass, the customer base will not peak out of the blue and exceed the projected platform capacity. The demand for scaling systems would be estimated by quarterly growth indicators.
* Deployment and Hosting: It is assumed here that DriverPass will host the platform by leveraging the cloud-based infrastructure (AWS, Azure, or GCP) with the support of continuous deployment, scale, and monitoring.
* Training Package Stability: The first three training packages are projected to be the main offering at the very least on the first version of the system. These packages may require future customizing, though this does not have to be done in the initial launch.
* Legal Responsibilities: It is based on the assumption that the set legal standards regarding the storage and transmission of information about the user, including both financial and personal data, will be addressed through the cooperation of the development team with the legal team of DriverPass and should not be the sole responsibility of the development team.

### 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?*

* Initial Budget Constraints: The initial development will occur with minimal startup investment, and this will potentially limit the scope of initial functionality, performance tuning and third-party integration. As the scalable architecture will be constructed, it can be delayed till the launch to have the full automation or more sophisticated features such as the AI-enabled scheduling or on-demand 24/7 support bot.
* Feature Scope at Launch: As such, because of time and resource constraints, only the three core training packages will initially be implemented in the first system rollout. Complex features like the ability to customize the packages offered, tracking the instructor performance or even customer reviews are part of the potential future additions.
* Third Party Dependency Risks: It will also have third-party dependency risks because the platform will use DMV APIs, payment processing APIs, and communication APIs (email/SMS), all of which may experience downtimes, policy updates, or deprecated endpoints. These dependencies are one of the possible limitations that cannot be directly controlled by the development team.
* Geographical Limits: The very first version of the platform considers its implementation in only one or a few regions. It might not initially be in favor of being localized (multi-language interface, regional regulatory compliance) until the turnover or other sources of finance could ensure such growth.
* Real Time Support and Offline Access: Though the system function is cloud-based and designed to be operable at any time 365 days a year, it is not guaranteed of human support in real-time when the business is offline. There will also be limited offline access to the system and the users will need a stable internet connection in order to leverage the core functionalities.
* Testing and Compliance Timeframes: Between regulatory compliance and rigorous testing, iterative work may be necessary past the original deadline based on how quickly the documentation, legal reviews, and certification may take place.

### 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.*

