# 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 specific purpose of this project is to create a system online where customers can login and create, manage and request services related to DMV driving tests and practice. The client is called DriverPass which wants to fill a void with training new drivers for their driving test.
* DriverPass would like their system to do the following: Handle reservations, create accounts for both the user but also driver teachers and the administrators of the site.
* The drivers who are taking students around need to be able to track users that are matched up with drivers the time and their car’s.
* They would like different tiers.
  + Package One: Which includes 6 hours in a car with a trainer
  + Package Two: Which includes 8 hours in a car with a trainer and in-person lessons
  + Package Three: Which is 12 hours in a car with a trainer, in-person lessons where they explain rules and regulations and access to all online classes and materials with practice tests.
* A fluid registration system. For example, the user creates an account and DriverPass gets a Phone Call and then they get the customer information in this fashion. This information is First Name, Last Name, Address, Phone number, CC information. It should also include pickup and drop off locations.
* This system would need to be a cloud system that is accessible from any device anywhere. This could mean apps, and access to administrator content on the fly.
* Another feature that needs to be included is driver notes, which allow comments for the driver to post and for the trainee to post.
* They also would like API integration into the DMV so that all information is current and updated for that county or state.

### 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?*

The background of the proposed system is provided to:

* The system needs to be able to offer online classes covering DMV rules and regulations to new users who have never taken the driver test before or who have failed but need to take it again.
* It also needs to provide Practice tests simulating the actual DMV test environment.
* Finally, it also needs to provide scheduling for the on-the-road training sessions with certified instructors.
* The system must perform a wide range of data access functions. The data needs to be able to be accessed online from any device or computer. It must also be able to download reports and information for offline use.
* It must also implement role-based access control for different employee roles. And full access for IT officers to manage and reset accounts.
* Packages also need to be included in this. There are 3 tiers of packages.
  + First is a 6-hour package that includes on-the-road driving instructions.
  + Second is 8 hours of on the road driving instruction, an in-person lesson on DMV rules and policies.
  + Third is 12 hours of on the road driving instruction, in person lessons, and access to all online content including classes and practice testing.
* The system also needs to collect detailed customer information during registrations. This includes CC information, user contact information, Address and phone numbers. It should also be able to schedule and manage driving appointments to include, time, date, car, where pick up and drop off locations are.

### 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 do the following when it is completed:

* The system should be able to handle reservation management. It should be able to support reservation management process, allowing customers to book, cancel, and modify driving lessons online. Also, should be able to handle reservations made via phone or in person, with functionalities to track driver assignments, lesson timings and vehicle allocations.
* The next thing it should be able to do is offer different training packages. It ideally should offer 3 training packages that are flexible enough to enable or disable specific training packages based on business needs.
* The system should be able to handle customer information. This system must collect and store detailed customer data during registration, this includes contact and payment details. It should also facility scheduling and managing driving appointments effectively.
* The system should also feature a cloud-based, user-friendly interface accessible to all user roles. This includes tracking and reporting tools that allow users to monitor activities and interact with the system efficiently.
* They system should have the following measurable tasks:
  + Implementation of role-based access functionality. This can include configuring user roles and permissions within the system.
  + Activity tracking and reporting is next. This can include mechanisms for logging changes to records and developing reporting tools for generating activity reports.
  + Designing and implementation of tracking packages is next. Configuring the system to manage three predefined training packages and provide the capability to enable or disable these packages as needed.
  + The system should develop customer information management systems. Implementation of forms and processes for collecting and storing customer data and creating scheduling tools to manage driving appointments.
  + Create a reservation management system. This system should integrate with phone and in-person reservation systems. This should also include tracking for drivers, lesson times and vehicles. The development of online reservation booking, cancellation and modification should be included in this.
  + Ensuring compliance with DMV requirements. Developing integrations for receiving updates from DMV systems and implementing notification systems for alerting users to change.

## 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 environment that the system needs to run in is as follows:

* The system must operate as a web-based application to ensure accessibility from various devices, including computers and mobile devices. The system should also be compatible with major web browsers such as Google Chrome, Mozilla Firefox, Safari etc.
* It should also be based on a cloud-based infrastructure. This leverages cloud-based hosting solutions to support scalability and flexibility. Which allows the system to handle varying loads efficiently and ensures that the applications remain accessible and responsive.
* The system should consider the following when dealing with speed and throughput:
  + The system must provide quick response time to enhance user experience. Ideal response times for user interactions, such as page loads, form submissions, and search queries should be within 2 seconds. For more complex operations such as generating detailed reports or processing large datasets, response times should not exceed 5-10 seconds.
  + Another consideration is system throughput. The system should be capable of handling multiple concurrent users without performance degradation. It is essential to estimate peak usage scenarios and ensure that the systems infrastructure can support the expected load.
  + The system must be designed to be scaled horizontally and vertically to accommodate increasing numbers of users and data volume. Cloud-based solutions often provide built-in scalability features that facilitate this growth.
  + Finally, the system should have low latency. The system should aim for low latency by opti8mizing data transfer protocols and reducing the size of transmitted date. Using CDN could help to decrease this speed. Efficient database design and indexing as essential to minimize query latency. Things such as query optimization, caching, and indexing to ensure that database operations are performed swiftly.

#### 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 constraints could be a few different things and those solutions could be as follows:*

* Supported platforms are very important to this overall. Supporting the most popular platforms should be the first consideration:
  + Windows: The system should be compatible with Windows 10 and 11 at a minimum and ensures that users can access the system using it.
  + macOS is also essential to support users on Apple devices and support OS such as MacOS Ventura and newer.
  + Linux, even though Linux is not as popular it is still used widely. It should support Linux operating systems such as Ubuntu and CentOS.
* The system should also support all modern browsers such as Chrome, Firefox and Safari. Because of this it is important to ensure that the system works flawlessly on each platform. Mobile browsers are also important in this because of how widely used mobile devices are in our lives.
* The system should use a few different backend systems which include the following:
  + Relational databases are important here because of their robustness, scalability and support for complex queries and transactions. Databases such as MySQL or PostgreSQL can be used in this instance because of their ease of use and robustness and scalability.
  + We can simplify the application server for this to a few different server-side technologies.
    - First being PHP for its use in web applications, providing extensive support for various frameworks and libraries.
    - Node.js could also be used for its asynchronous processing and it’s suitable for handling real-time data and large-scale application.
    - Finally, Python, which provides different frameworks such as Django or Flask, supports rapid development and is known for readability and use.
  + The system will use cloud infrastructure:
    - The biggest challenge will be determining which cloud provider to use that can provide the system with the best uptime, scalability, resources and any measurable metrics that are needed for our platform.
  + The system will also require backup and recovery:
    - The system should implement some type of automatic back up that regularly backs up critical data. This helps to protect data in case of hardware failures, data corruption or accidental deletions.
    - A disaster recovery plan should also be in place to ensure continuity of service in the event of major outages or failure. This plan should include procedures for restoring services and data from backups.

#### 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 accuracy and Precision are as follows:

* User identification. The system should clearly differentiate between various user roles, including administrations, IT officers, secretaries and customers. This differentiation ensures that each user has appropriate access and capabilities based on their roles.
* Each user must have a unique identifier. Such as username, email address to avoid confusion and ensure accurate tracking of actions and changes. This should be designed in a way to handle input in a case-insensitive manner for usernames, ensuring that login attempts are not affected by case variations.
* The system should use input validation and error handling. The system must validate all user inputs to ensure that they meet predefined formats and constraints. For instance: valid phone number in the phone number field, properly formatted email addresses in the login forms.
* Error detection must handle errors gracefully, providing users with clear feedback and instructions for correcting invalid inputs.

#### 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 should be Adaptable in the following ways:

* User Management. There should be flexibility in user change.
  + Role management should be supported and should include the addition, removal and modification of user roles and permissions without requiring code changes. This can be achieved through user friendly interface that allows administrators to adjust roles and permissions as needed.
  + The system should have some type of account management. Administrators should be able to manage user accounts effectively, including creating new accounts, deactivating accounts, and updating user information. This functionality should be accessible through a secure admin panel.
* Platform Changes.
  + The system should be designed to adapt to updates in the underlying platform or technology stack. This includes ensuring that the application remains compatible with new versions of OS, browsers, and database systems.
  + Implementation of a testing framework to validate the system’s compatibility with platform updates before deploying changes. This helps to prevent disruptions and ensure that the system continues to function as expected.
* IT Admin Access
  + Administrative control. The admins require access to configuration settings, user management tools, and system logs. This access should be restricted to authorized personnel only and protected by robust authentication mechanisms.
  + Ensure that IT admins have the necessary permissions to perform their tasks, such as resetting passwords, modifying user roles, and addressing system issues, while preventing unauthorized access to sensitive data.

#### 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?*

The system shall provide Security as follows:

* User login:
  + Users must authenticate themselves using secure credentials. This typically is a combination of username/email and password. The system should enforce strong password policies, including minimum length, complexity and periodic changes.
  + Two-Factor authentication should be used as an additional layer of security, requiring users to provide a second form of verification (e.g., a code sent to the mobile device).
* Securing data transmission
  + Utilize encryption protocols such as HTTPS(SSL/TLS) to secure data exchanges between clients and servers. This ensures that data transmitted over the internet is protected from interception and tampering.
  + Encrypting sensitive data, such as credit card information and personal identifiers, both at rest and in transit. Use industry-standard encryption algorithms to safeguard data against unauthorized access.
* Protecting against security threats:
  + Implement mechanisms to detect and respond to brute force attacks by locking accounts after a specified number of failed login attempts. This helps prevent unauthorized access through repeated password guessing.
  + Set up monitoring tools to detect suspicious activities and generate alerts for potential security threats. This enables timely intervention and mitigation of attacks.
* Password Recovery:
  + Providing a secure password recovery process that includes verifying the user’s identity through email or SMS before allowing password resets. Ensure that the process involves secure authentication methods to prevent unauthorized password changes.

### 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 users to create and manage their accounts, including registering new accounts, updating personal information, and managing account settings.
* The system shall provide role-based access control to differentiate between admins, IT officers, secretaries, and customers, ensuring that each role has appropriate permissions and access levels.
* The system shall allow customers to schedule driving lessons online by selecting available times and instructors, and make reservations for different driving packages (e.g., six hours, eight hours, twelve hours.).
* The system shall enable customers to modify or cancel their driving lesson reservations online and through phone or in-0person requests, with appropriate notifications sent to both the customer and the office.
* The system shall generate and provide downloadable reports on reservations, lesson schedules, and user activity which can be accessed and manipulated offline.
* The system shall enable users to log in using their unique credentials and provide mechanisms for password recovery and resetting in case of forgotten passwords.
* The system shall manage the allocation of driving instructors and vehicles, ensuring that each lesson is assigned to a specific instructor and vehicle based on availability.
* The system shall track and record all changes to reservations, including who made the reservations, who modified or canceled it, and when these actions occurred.
* The system shall integrate with the DMV to receive updates on new rules, policies, and sample questions, and notify the administrators of any changes to ensure that training content remains current.
* The system shall support secure, encrypted communication between the client and the server to protect data during transmissions.
* The system shall allow administrators to manage user accounts and roles, including creating new users, modifying existing accounts, and deactivating accounts as needed.
* The system shall offer automated notifications and alerts for critical events, such as failed login attempts or potential security issues, and provide administrators with tools to address these events.
* The system shall maintain an audit trail of all the suer actions and system changes, allowing trace and identification of any issues
* The system shall support flexible package management. Allowing the changing, disabling and adding of training modules.

### 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 system User interface shall be designed as follows:

* The interface must be intuitive and easy to navigate, ensuring that users can efficiently perform their tasks without confusion.
* The design should adhere to accessibility standards, making the system usable by individuals with disabilities.
* The interface must be responsive, adapting to various screen sizes and devices including desktops, tablets and smartphones. This ensures consistent experience across devices.
* There needs to be a logo at the top. Then on the left side an online test progress box that displays test results and progress for the online courses.
  + Next is notes that are integrated with notes that the driver and the secretary and the user create and post so that everyone can read them.
  + Next is user information. Personal information that is displayed in an easy-to-read box for the user.
  + Then any special needs that the user has
  + A driver photo and a user photo.
* Some key functional elements to consider:
  + Navigation must be clear and accessible.
  + Forms and inputs for things such as booking lessons, entering personal information and managing user accounts must be user friendly and validate input to prevent errors.
  + The interface should provide timely notifications for important events such as reservation confirmation, cancellation or system updates.
* Different uses sand their needs:
  + Administrators:
    - Admins need a comprehensive dashboard to manage user accounts, oversee system operations, and generate reports. They should be able to view and modify all aspects of the system.
    - They also should be able to run detailed reports
  + IT Officers:
    - They need tools for managing system configurations, performing maintenance tasks, and handling security issues. This includes the ability to update system settings, manage backups, and monitor system health.
    - They should be able to assist users with account-related issues, such as resetting passwords or managing user roles.
  + Secretaries:
    - Secretaries require an interface for scheduling and managing driving lessons, including the ability to view available time slots, assign instructors and handle appointment changes. They need tools to interact with customers, including options for modifying or canceling appointments based on phone calls or in person requests.
  + Customers:
    - Customers should have access to an online portal where they can book, modify, or cancel driving lessons, view their progress and manage their personal information. They need user friendly interface as described above to track their information.
  + Web based access:
    - The system will primarily be accessed through web browsers on desktops and laptops. The interface must be compatible with popular browsers (e.g., Chrome, Safari, Firefox). The interface should use responsive design to ensure that tit adapts seamlessly to different screen sizes, providing an optimal viewing experience.
  + Mobile Access:
    - The system should be optimized for mobile devices, including smartphones and tablets. This involves designing a mobile-friendly version of the interface that maintains usability and functionality. It should also support touch screen interactions including swipes, taps, and gestures.
  + Offline Access:
    - Although the system is designed for online access, there should be provisions for offline access to certain functionalities such as view previously downloaded reports or information. Anything done offline should be synchronized online when available.

### 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?*

The system specific assumptions are:

* User training and support:
  + It is assumed that users, including administrators, IT officers, secretaries and customers will have a basic understanding of how to use web-based interfaces and mobile applications The design does not explicitly address user training or support materials which could be crucial to ensuring that users are able to effectively utilize the system.
* System performance under load.
  + The design assumes that the system will handle expected user loads efficiently. Specific performance metrics, such as max number of concurrent users o9r peak usage time were not detailed.
* Compliance and Legal requirements:
  + It is assumed that the system will comply with relevant legal and regulatory requirements, such as data protection laws and industry standards.
* Assumptions about users and Technology:
  + It’s assumed that users will have access to modern web browsers and mobile devices that support the system’s features.
  + It’s assumed that the users will have a stable internet connection to access the system’s online features. Offline access was mentioned but may not cover all scenarios where users have intermittent or unreliable connectivity.
  + It’s also assumed that a standard user experience is based on common web and mobile interface conventions.
  + It’s assumed that users expect a high level of security and privacy for their personal and financial information.

### 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 has the following limitations:

* User Customization:
  + The current design doesn’t fully account for extensive user customization options.
* Offline Functionality:
  + The system does allow offline work, it’s limited to viewing previously downloaded information. It doesn’t support full offline functionality such as the ability to schedule or modify appointments without an internet connection.
* Integrations with external systems:
  + The system design assumes integrations with the DMV for updates on rules and policies however, it does not specify the technical details or requirements for this integration. Challenges related to API compatibility, data synchronization and real-time updates may arise.
* Time constraints:
  + The project timeline, as outlined in the schedule is constrained by fixed deadlines for each phase of development, including design, implementation and testing. This leaves limited time for addressing unforeseen issues or incorporating additional features.
* Budget constraints:
  + Budget limitations could affect the scope of the project, including the depth of testing, the inclusion of advanced features, or the ability to hire additional expertise for specialized tasks.
* Technological Resources:
  + The design assumes the availability of current web technologies and development tools. If the development team encounters limitations with existing technology or faces challenges with integrating new tools, it could impact the systems functionality or performance.

### 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

Description automatically generated