# 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 develop a system tailored to the needs of the client, DriverPass. DriverPass specializes in offering driver tests and practice to individuals aiming to acquire their driver’s license from the DMV. The client seeks to expand their business capabilities by implementing a system that enables online reservations, scheduling, and payment functionalities for users. Additionally, the system should provide DriverPass employees with access to user data and schedules, as well as facilitate the scheduling of drivers to users. In essence, the client desires a comprehensive system that enhances their current business model and streamlines operational processes.

### 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 gap in the market where individuals preparing for their driver skills test at the DMV often lack adequate preparation and consequently fail the test. They seek to remedy this by providing better practice tests and driver training to increase the success rate of test-takers.
* The system needs to be able to:
  + Provide users with the ability to store their information securely in accounts.
  + Enable users to schedule driver tests or driving practice sessions conveniently.
  + Allow users to track their progress in preparation for the driver skills test.
  + Facilitate online payments for the services offered by DriverPass.
  + Enable DriverPass to monitor its employees and the vehicles used for training efficiently.
  + Ensure seamless accessibility to the system from any internet-enabled device.
* The components needed are:
  + User Account management: A system for securely storing user information and managing accounts.
  + Scheduling Functionality: Capability for users to schedule driver tests or practice sessions.
  + Progress Tracking: Feature allowing users to monitor their progress in preparation for the test.
  + Payment Gateway: Integration to facilitate online payments for the service provided by DriverPass.
  + Employee and vehicle management: Tools for DriverPass to oversee its staff and the vehicles utilized for training.
  + Cloud-Based Infrastructure: Infrastructure to ensure accessibility from any device with an internet connection.

### 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 Database Management:
  + Develop a system to securely store store user data in a cloud-based database. Users can input personal information into their secure accounts accessible through login.
* Intuitive User Interface:
  + Obtain customer approval for the overall design before proceeding with building the user interface. Design a user-friendly interface that provides client-users with access to their account details, including personal information, progress tracking, notes from their driver, scheduled services, and their photo. Additionally, create an admin interface displaying options for scheduling, car selection, processed user payments, and other relevant client-user information.
* Automatic Car/Schedule Matching:
  + Implement a function to organize DriverPass employees and available cars. Develop a system that automatically matches available employees with cars and schedules the to train drivers based on user-requested days. Allow users to modify automatic selections if necessary.
* Employee Roles:
  + Establish parameters distinguishing between employee accounts and general user accounts. Define specific parameters for each type of employee using the system and allocate corresponding permissions.
* Automatic DMV Updates:
  + Create a function to regularly check for DMV updates through their website or database. Set up a notification service to alert administrators of any DMV updates promptly.
* Cloud Accessibility:
  + Design the system to operate in the cloud for universal accessibility. Develop functions to compile specific datasets into downloadable CSV spreadsheet files and upload modified data to the cloud, ensuring the most recent modifications are reflected.
* User Profile Interactions:
  + Enable users to access their schedule of purchased services, make changes, cancel, or pay for services. Offer three different service packages for purchase. Implement a dashboard-style interface to display user progress. Incorporate a service that connects to an online map repository for setting pick-up and drop-off locations when a user purchases a service.
* System Testing:
  + Conduct static and dynamic tests over a period of 10 days to ensure the system functions correctly.
* Product Delivery:
  + Complete system testing with the test team before delivering the finished product.

## 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?*

* This system needs to operate in both web-based and application environments. Specifically, it requires a web app compatible with PC web browsers like Firefox or Chrome, and a smartphone app for mobile access.
* In terms of speed, the system should be capable of supporting video streaming efficiently, a critical component of DriverPass’s business model. While the quality of the stream depends on user bandwidth, the system should handle multiple users streaming driver’s education videos simultaneously without significant lag.
* The system should be updated at least as frequently as the web browsers and mobile operating systems it supports. Whenever updates are released, such as those from Google Chrome or smartphone OS updates, the DriverPass web app should be tested on the new versions.

#### 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 platforms compatible with web browsers, which typically include Windows, Unix, and others supporting HTML-based applications. For the mobile application, support is needed for Android and iOS platforms at least. While niche mobile operating systems exist, it’s recommended to focus marketing efforts on web application usage for such users until a dedicated application is developed, subject to market research findings.
* To support the application, the backend requires tools such as a database for storing user information for clients, administrators, and drivers. Depending on developer preferences, a backend development tool like ASP.NET core or Node.js may also be necessary. SQL is essential for the server to handle database queries effectively.

#### 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 should allow for user management tasks such as adding, removing, or modifying user accounts without necessitating changes to the underlying code. This can be achieved through an administrative interface that provides functionalities for managing user accounts directly from the system’s dashboard or control panel.
* To ensure compatibility with platform updates, the system needs to be designed with flexibility in mind. This can involve using standardized technologies and frameworks that are less likely to be affected by platform updates. Additionally, regular testing and updates should be conducted to ensure the system remains compatible with the latest platform versions.
* The IT administrator will require elevated access privileges to perform various tasks related to system maintenance, monitoring, and troubleshooting. This includes permissions to manage user accounts, access system logs, configure server settings, and install updates or patches as necessary. Access levels may vary depending on specific responsibilities, but generally, the IT admin should have full access to system resources to effectively manage and maintain the system.

#### 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?*

* Users can be managed within the system without requiring changes to the underlying code. This means administrators can easily add, remove, or modify user accounts through the system’s interface, utilizing database storage and manipulation mechanisms.
* To ensure the system adapts to platform updates, it should be designed with flexibility in mind. Utilizing standardized technologies and frameworks that are less susceptible to platform changes is key. Additionally, regular testing and updates are crucial to maintain compatibility with evolving platform versions.
* The IT administrator requires elevated access privileges to effectively maintain the system. This includes permissions to manage user accounts, access system logs, configure server settings, and install updates or patches. While they can view certain user information and update specific details, they won’t have access to sensitive information such as previous payment details or newly submitted data once it is entered into 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?*

* To log in, users must provide a unique user ID along with a strong password, meeting specific criteria such as minimum character length, including uppercase and lowercase letters, numbers, and symbols.
* The connection and data exchange between the client and the server can be secured by implementing HTTPS over TLS protocol. This ensures that data transmitted between the user’s device and the server is encrypted, preventing unauthorized access or interception of sensitive information.
* In the event of a brute force hacking attempt, where an attacker repeatedly tries to guess a user’s password, the account should be locked after a certain number of failed login attempts. This prevents further unauthorized access and protects the account from potential compromise. The user would then need to contact IT support for a password reset to regain access to their account.
* If a user forgets their password, they are typically allowed multiple attempts to log in. However, after a certain number of unsuccessful attempts, their account may be temporarily locked for security reasons. To regain access, the user can initiate a password reset process through a “forgot password” link provided on the login page.

### 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 support two-factor authentication for all users.
* The system shall keep track of the user’s progress in testing.
* The system shall store user information in private accounts, including driver notes, special needs, driver photo, address information, and payment details.
* The system shall allow users to schedule driver tests and driving practice sessions through one of three pre-loaded packages.
* The system shall enable users to make payments online.
* The system shall track employees of the business.
* The system shall monitor the scheduling of cars assigned to employees.
* The system shall track user data and progress continuously.

### 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.)?*

* Interface Needs:
  + The interface needs to be user-friendly, simple, and mobile-friendly. It should support touch or click-based interactions, with keyboard support for inputting information. Additionally, it should provide easy navigation and accessibility across different devices, including phones, tablets, and PCs.
* Different Users for the Interface:
  + The interface caters to three main user groups: customers, employees, and administrators.
* Tasks for Each User Through the Interface:
  + Customer Users:
    - They should be able to log in using credentials, check account status and progress, review account balance, schedule driver tests and practice sessions, update profile and account information, and manage billing details
  + Employee users:
    - They need access to their schedules and the schedules of assigned cars. They should be able to view and update their employee profiles, including availability status.
  + Admin users:
    - They require extensive access to user and employee information, including the ability to reset passwords and unlock accounts. Admins should be able to view and manage employee schedules, make adjustments, add or remove users and employees from the system, receive updates from the DMV, and schedule appointments for users’ driver tests and practice sessions.
* User Interaction with the Interface:
  + Users will interact with the interface primarily through mobile devices such as phones and tablets, as well as through web browsers on PCs. The interface will be designed to support touch or click-based interactions, with keyboard support for inputting information. It will be optimized for simplicity and mobile-friendliness to ensure a seamless user experience across different devices and platforms.

### 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 programming language used to develop the application was not specified.
* The final visual design and color scheme of the website were not determined, nor were the settings to accommodate different user preferences or needs.
* The hosting location for the driver-test videos was not finalized, with considerations regarding bandwidth usage, storage space, and potential costs left unaddressed.
* Assumptions:
  + Users are assumed to have access to modern devices such as smartphones or PCs with updated web browsers.
  + It is assumed that users are familiar with using web applications and can navigate them effectively.
  + The design assumes that users have internet access to stream driver-test videos and perform other online tasks.
  + The technology infrastructure required to support the application, including server capacity and bandwidth, is assumed to be in place to accommodate user traffic and data storage needs.

### 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?*

* Resource Limitations:
  + The size of the development team may be limited, affecting the speed and capacity to implement all desired features with the given timeframe. This could result in prioritization of features or reduced functionality.
* Time Constraints:
  + There may be limitations on the project timeline, potentially leading to challenges in delivering a fully comprehensive system within the allocated time. This could result in the need to streamline development processes or cut back on certain features.
* Budget Constraints:
  + The project budget has not been established, which could potentially lead to limitations in terms of available resources for development, implementation, and maintenance. Overstepping the budget could necessitate a renegotiation of project requirements or additional funding.
* Technological Limitations:
  + Depending on the chosen technology stack and infrastructure, there may be limitations in terms of scalability, performance, or compatibilities of the selected database host, particularly in terms of bandwidth and storage capacity.

### 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 close-up of a calendar

Description automatically generated