# CS 255 Business Requirements Document Template

## 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?*

* Build a system for DriverPass, a company that provides driver training and practice tests to customers who want to pass their driving tests at the DMV.
* The system must handle online classes, practice tests, and on-the-road training for customers.
* The system should be able to access data from anywhere, online as well as offline.
* The client also wants the system to be flexible so that they can customize the packages in the future.

### 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 problem that DriverPass wants to fix is the lack of proper driver training, which leads to many people failing their driving tests at the DMV.
* DriverPass wants to offer a solution to this problem by providing better driving training.
* The solution they want to offer is a system that handles online classes, practice tests, and on-the-road training for customers.
* The system should be able to access data from anywhere, online as well as offline.
* The client also wants the system to be flexible so that they can customize the packages in the future.

### 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 handle online classes, practice tests, and on-the-road training for customers.
* The system should access data from anywhere, online as well as offline.
* The system should be flexible so that the client can customize the packages in the future.
* The system should track user activity, such as reservations, modifications, and cancellations.
* The system should allow users to make appointments, cancel, and modify appointments online.
* The system should identify the driver the customer is scheduled to go out with, since there are many drivers and many cars.
* The system should allow the user to make reservations for driving lessons.
* The system should allow the user to pick one of three packages: six hours in a car with a trainer, eight hours in a car with a trainer and an in-person lesson, or twelve hours in a car with a trainer, an in-person lesson, and access to an online class with all the content and material.
* The system should allow the user to schedule appointments over the internet.
* The system should allow the user to automatically reset their password if they forget it.

## 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?*

* **Responsiveness:** The system must respond quickly, with little lag to user interactions and have

no delays or issues loading screens.

* **Load Capacity:** The system should handle a high number of users without compromising performance or functions.
* **Uptime:** The expectations are that the system has 99.9% uptime, minimizing any issues with

access and experience.

* **Optimization:** To ensure cost control, the system should optimize the use of computing

resources.

#### 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?*

* **Cross-Browser Compatibility:** The system should run flawlessly on different browsers to ensure

consistent behavior increasing user satisfaction and providing an experience.

* **Mobile**: The system should adapt to various mobile platforms, iOS and Android, and various

screen sizes and resolutions providing optimal viewing experiences across devices and

platforms.

* **Database Technology:** The database must support efficient handling of data with scalability

options to support growth without a degradation in performance.

* Cloud Service: The system should be compatible with top cloud providers for scalability, performance and security reasons.

#### 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?*

* **Data Validation:** The system should implement strict data validation measures to ensure accuracy of inputs and avoid any data being corrupted.
* **Error Handling:** The system should have strong error handling to manage inaccuracies in processing and provide messages to users when error occurs.
* **Time Syncing:** The system should maintain accurate time synchronization accurate precision related to time related data or operations.
* **Real Time Data**: The system should reflect changes in state in real time to maintain data consistency across components. Crucial in terms of user payments and registration.

#### 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?*

* **Modular Architecture:** The system should be designed modularity which will allow change with ease in terms of feature additions, changes to existing features and any on-the-fly changes as conditions require.
* **Software Update Compatibility: The** system should be compatible with any updates to ensure long-term use and security considerations.
* **Flexibility:** Each user has unique needs and preferences, and the system should accommodate these user preferences.
* Integrations: Integration capabilities are critical, and the system should be able to handle external services to extended functionality.

#### 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?*

* **Login Requirements:** User login should require credentials, multi factor authentication should be used for greater security. Passwords should be secured using techniques such as hashing.
* **Secure Data Exchange:** The connection and data exchange between client and server should be done using proven protocols such as HTTPS.
* **Additional Password Protection: Multiple** failed login attempts should trigger warnings and notifications and lock the user account.
* **Password Recovery:** The system should provide a secure and easy password recovery option to reset and recover the user password.

### 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 users via their credentials when logging in
* The system shall allow users to register, schedule, update and cancel appointments.
* The system shall provide users with online classes and practice tests.
* The system shall on the road training scheduling.
* The system shall allow administrators to add, change or remove training packages.
* The system shall track and report activities and any user progress.

### 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 user interface must be intuitive, responsive and easy to navigate. The contouring interface will provide easy access to crucial user functions.
* **Users:** Users include Admin, IT Admin, Students, and Drivers, which all have different needs and require different access levels.
* **Interaction:**  User will interact with the interface primarily through browsers on a variety of platforms and devices, such as desktop and mobile.

### 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 have basic understanding of web navigation and access to a stable internet connection.
* It is assumed that users have access to devices that are able to use the latest web technologies and latest, modern browsers.
* Adequate security measures are assumed to protect user data and ensure we act with integrity with the user data.

### 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?*

* Reliance on user’s technological literacy and their access to the internet and compatible devices.
* Tight budgets may limit advanced features from being developed and may constrain scope.
* Time constraints may limit testing and not allow further iterations of the product resulting in refinements being limited.
* Initial version may be limited intentionally in order to receive feedback from user’s and develop those features they want.

### 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.*

