# 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?*

* The purpose of this project is to create an online platform for DriverPass to provide comprehensive driver training to prepare students for driving tests. DriverPass wants to offer online practice exams and on-the-road training to address the gap in current training methods.
* The client, DriverPass, aims to build a system that facilitates online learning, practice exams, on-the-road training scheduling, and administrative functionalities.

### 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 aims to fix the issue of insufficient preparation for driving tests among students. The system is designed to offer online classes, practice tests, scheduling for driving lessons, and tracking user progress.
* The key components needed for this system include an online interface for users to access courses and practice tests, a scheduling module for driving lessons, an admin panel for managing user accounts, and a connection to the DMV for updated regulations.

### 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 provide online courses, practice tests, and scheduling for driving lessons to adequately prepare students for their driving tests.
* Measurable tasks include creating user profiles, implementing an online scheduling system, developing a secure payment gateway, providing access to updated DMV regulations, and enabling user progress tracking.

## 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 system needs to run in web-based environments with optimized speed. It should have minimal downtime and regular updates to incorporate new features or regulations.

#### 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 needs to run in web-based environments with optimized speed. It should have minimal downtime and regular updates to incorporate new features or regulations.

#### 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 identification should be distinct and accurate. The system should promptly notify admins of any errors or issues that may arise during user interaction.

#### 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 architecture should allow for easy modifications to user functionalities without the need for extensive coding. It should adapt seamlessly to platform updates. The IT admin requires full access for maintenance and user management.

#### 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 will log in with secure credentials. Data exchange between the client and server will be encrypted. In case of a "brute force" hacking attempt, the account will be locked temporarily. A password reset feature will be available for forgotten passwords.

### 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 provide online course access, practice tests, scheduling, and progress tracking for students preparing for driving tests.
* It shall enable user registration, scheduling driving lessons, selecting packages, viewing lesson history, and contacting support.

### 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 interface should be user-friendly, accessible via web browsers and mobile devices. Users should have easy navigation, quick access to courses, progress tracking, and a secure payment portal.

### 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?*

* Assumptions include assuming users have basic internet access, and the technology infrastructure can support the system's requirements.

### 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?*

* Limitations include potential resource constraints, budget limitations for implementing extensive features, and potential challenges in adapting to frequent DMV regulation updates.

### 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 calendar

Description automatically generated