Nash Ellis

CS-255

Prof. Smallish

01/12/25

# 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 design a secure and user-friendly system for DriverPass to help students prepare for driving tests more effectively. DriverPass’s goal is to provide comprehensive online practice exams, on-the-road training sessions, and lessons on DMV rules and policies. The system will streamline scheduling and ensure compliance with DMV requirements. By creating a scalable and cloud-based platform, DriverPass aims to fill a gap in the market for effective driver training solutions.

### 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: Many individuals fail their DMV driving tests due to inadequate preparation and the lack of high-quality training resources.

-Solution: DriverPass proposes a system that provides:

* Online practice exams and lessons tailored to DMV requirements.
* On-the-road training sessions with professional instructors.
* Scheduling and reporting tools to track customer progress and lesson completion.

-System Components:

* Cloud-based platform for training materials and data management.
* Role-based access control (RBAC) for different users (owner, IT admin, secretary, customer, and trainer).
* Integration with DMV systems to receive updates on driving laws and requirements. Also allows updated rules and regulations of the DMV to be obtained.
* Logging and tracking tools to monitor activity and ensure accountability.

### 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 DriverPass system will:

1. Provide physical on-the-road training much like Defensive Driving schools.

* Allow customers to schedule driving lessons.
* Offer simulations of different conditions or terrain (snowy roads, wet roads, or any other conditions to combat a certain weakness).
* Offer flexible packages to different levels of customer.

1. Provide training resources with updated laws and requirements regarding the customer’s country, state, or providence.

* Allows customers to learn the driving laws of their current location or the laws of other countries or states in the event of traveling or moving.
* Provide online lessons, practice tests, and feedback to customers.

1. Provide scheduling many scheduling options

* Allow users to choose how they want to schedule appointments and/or view their history and progress (text, call, online, in-person, email, etc.)

1. Implement Role-Based Security

* Defining the roles of users and limiting the access and privileges they have regarding the application and servers is crucial to remaining secure.

1. Use Cloud-Based Infrastructure:

* A cloud-based infrastructure is less upfront costs, easier to scale, and requires little effort of maintenance from DrivePass.

1. Ensure Real-Time Updates:

* Immediately update any changes made to reservations, customer packages, user data, or training material.

Refined Measurable Tasks for System Goals:

Training Resources

* Track the number of users accessing online lessons and practice tests.
* Measure practice test success rates (scores and pass rates).

Scheduling and Reservations

* Ensure scheduling or modifying a lesson takes less than 1 minute.
* Monitor the percentage of conflict-free appointment bookings.

Role-Based Security

* Track and block unauthorized access attempts.
* Measure the number of password resets and role updates performed by admins.

Real-Time Updates

* Confirm changes to reservations or training materials and sync across devices very quickly.

User Engagement and Satisfaction

* Monitor user feedback ratings for training sessions and lessons.
* Track the completion rates for online lessons and training packages.

System Uptime and Scalability

* Ensure system uptime is as close to 100% as possible.
* Support seamless operation with simultaneous users.

## 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 will be web-based, and accessible via browser on mobile devices and computers.
* Load times and pull requests should be executed within a few seconds.
* The system should be updated at least monthly, but the more updates the better.

#### 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 will run on Windows, macOS, iOS, and Android.
* It will be accessible and compatible with most browsers (Chrome, Safari, and Edge).
* The backend will require a secure database, specifically MySQL.
* The server(s) will be hosted on Azure or AWS, since it has a lower initial costs than hosting them in-house and are relaible.

#### 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?

* Users will be uniquely identified using email and password.
* Case-sensitive authentication will be enforced.
* The system will send real-time alerts to the admin for failed login attempts and account activity.

#### 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?

* Admins must be able to add, remove, or modify users without code changes.
* The system should support API integration with third-party platforms.
* IT admins will have full access and permissions to the database and system logs.

#### 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 must log in using multi-factor authentication (MFA).
* Data will be encrypted using AES-256.
* After 3 failed login attempts, an email will be sent to the account owner, and the account will be temporarily locked.
* A reset password feature will be available for users if they have access to the account's email or phone number.

### 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 register and create accounts.
* The system shall validate user credentials during login.
* The system shall enable users to schedule, cancel, and modify driving lessons.
* The system shall support role-based security for different user access levels.
* The system shall allow instructors to log lesson feedback.
* The system shall track user progress on online lessons and practice tests.
* The system shall integrate with the DMV database for up-to-date regulations.
* The system shall generate reports on user activity and performance.

### 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-

* Customers: Access training materials, schedule lessons, track progress, manage settings, account info, and payment plans.
* Customer Support: Manage bookings, reset customer passwords.
* IT Admins: Monitor security, update user permissions.

User Interaction-

* The system should be accessible via web browsers and mobile devices.
* UI should be easy-to-use and mobiole-friendly.

### 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?

* Customers will have internet access to use the platform.
* Users will enter correct contact and payment 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?

* System performance may be affected by network connectivity.
* Manual intervention may be required for resolving complex booking conflicts.
* Customization of training packages will be limited initially.
* Budget constraints may restrict initial features or resources.

### 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.

