CS–255 Business Requirements Document

Curtis Felsher

10-25-2022: Completed the Systems Components and Design portion

11-22-22: Completed remaining portions of document

# 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 client, DriverPass, would like a new system to be built that will help train students for their driving tests. The client noticed a problem in which people were failing their driving tests due to the lack of courses to help pass the exams. The system that the client would like to be developed will use training lessons and packages that are booked through reservations. The system should allow for purchases of diverse driving lesson packages to suit all needs. The system should allow for data edits to be online only as well as employee access to data offline with changes being tracked.

### 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 company would like to fix the problem of people failing their driving tests as there is a lack of driving training courses. The system should have the ability to store information of reservations and client details. The system should also allow for user passwords to be reset, run through a web interface via the cloud, and connect with the DMV to update any material that needs updating. This will require a server to store data, a data server with encryption that can store user profiles, a structure that allows the application to run over the web, and integration with the local DMV to update any course material. The system will also require an interface for employees to input and store client info/reservations. The system should also be malleable in the sense that the interface can be adapted to add, remove, or change packages.

### 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 allow customers to reset passwords, access data offline, edit data while online, allow employees to make reservations for customers, allow customers the ability to accomplish the previous requirement themselves through the web interface, make use of different service packages, be able to track any edits made, and assign user roles using a system of least-privilege for employees.

## 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 would perform best as a web-based application in the cloud
* The system should be lightweight and fast with a minimal delay between interactions
* The system should be updated any time the DMV updates their information as well as any functionality added to the system

#### 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’s frontend should operate as a browser-based application on any browser
* The system’s backend should function through the cloud with the necessary databases to store user information, application data, and any logs generated throughout the day
* The platform should utilize Linux as the OS due to the minimal hardware requirements and security of Linux

#### 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 have user accounts that utilize a username and password system. The passwords should utilize case sensitivity
* Users should also have the option to add two factor authentication for an added layer of security
* If a user has 4 or more incorrect password inputs, users should get an email allowing them to reset their password
* Admin manual intervention may be required if the user loses access to their email to reset their password and would need to provide information that the user is the owner of the account

#### 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 administrator of the application should be able to alter any data within the application such as user account creation/deletion without code alteration
* All user changes should be able to be made by the user if it is simple information such as contact information and the administrator should be able to update user accounts manually through the backend without changes in the code
* The system should be able to be updated if a major browser change occurs by implementing OOP within the code. The code must be modifiable to add any additional features without breaking other functionality within the system
* The system should be able to remove/add packages as they are introduced or discontinued

#### 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?*

* A username and password will be required for the user to login as well as a verified email address and optional two-factor authentication
* A secure connection should be aided by firewalls to recognize malicious activity and lock out this type of behavior
* After four or more failed login attempts (the number can be changed based on client feedback), the user should be locked out and must reset their password via their verified email address. This will prevent brute force guesses of passwords by locking the account and preventing user account theft
* Admin accounts should be differentiated via the use of least-privilege for access roles

### 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 login with their created account
* The system shall validate the user via a suer created password and optional 2fa
* The system shall lock a user out of their account after four or more password failed attempts
* The system shall show online test progress
* The system shall display user information such as their name and address
* The system shall display driver notes
* The system shall show courses for the user to complete
* The system shall show different package options for the user
* The system should allow for data input for its necessary functions such as booking a reservation or changing account information if needed
* The system shall perform quickly and with minimal downtime if any is required
* The system shall display any changes made by the DMV to their regulations
* The system shall allow for backend downloads of reports for administrators

### 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 operate over a browser or an app runnable on any device such as a PC or phone with an internet connection
* The UI should adapt based off screen size of the device being used to access the application
* The different users of the interface should be the customers themselves, the product developer, and a company administrator
* Customers should be able to view information displayed by the UI in an easy and elegant way with continuity throughout the interface. Devs and administrators should be able to view all user accounts and system information to be able to make changes

### 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 the user has a device capable of displaying the application
* It is assumed that the user has an internet connection to access the application
* It is assumed that users know how to create their account as well as access their personal email account
* It is assumed that the budget for the project covers all aspects such as development and deployment of hardware like a Linux based system to host the cloud server

### 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 client has specified a window of time of around 16 weeks. If any aspect of the development is grossly miscalculated, this could cause delays in product development
* The client has not specified a budget so that may be a limiting factor
* Limitations could also occur if the code is implemented in a way that does not allow for easy updating in the case of a major browser change

### 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.*

Chart, waterfall chart

Description automatically generated