**5-3: Project One**

Joshua Brown

Department of Computer Science, Southern New Hampshire University

CS – 255: System Analysis and Design

Professor Maureen P. Kinkela, Ph.D.

July 30, 2023

# 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?*

* Our client is DriverPass, and it is their goal is to provide driver training in the form of online tests and on-the-road training for their customers.

### 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 creators of DriverPass noticed that there is a need for better training for young adult drivers. As young drivers begin the process of taking their driver’s tests for the purpose of getting their license, many of them end up failing. DriverPass’s solution is to create an online training course that uses practice tests to allow their customers to study for the written test, as well as provide their students with behind-the-wheel driver training so that they might pass the driving course portion of the test.

### 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?*

* Deliverables include:
  + A system that runs on a cloud-based platform.
    - The system can be accessed from any computer or mobile device.
  + Have user-based roles with different authorities (Administrative Roles, Security/IT-based roles, Secretarial-based roles, and Customer-based roles (regular user).
    - Roles are clearly defined and have separate privileges.
    - Admin Roles:
      * has full access to accounts and the ability to block access and reset accounts.
      * Ability to access and print system reports and other important documents.
    - Security/IT Roles:
      * Responsible for maintaining and monitoring system logs.
    - Secretarial Roles:
      * Answer emails.
      * Make general reservations (cancel and modify existing reservations).
    - User Roles:
      * Make reservations for their personal account (cancel and modify existing reservations for their own account).
      * Ability to take practice tests online.
      * Friendly user layout to house their interactive environment. Divided into sections:
        + Student info, online test progress, driver notes.
  + Security features:
    - Tracking of reservations :
      * Who made, modified, or canceled it.
    - Track when users make changes to system records.
    - Ability to print activity reports.
  + Packages offered to users for driving instruction:
    - Choices between packages 1-3 with differing details.
    - Ability to disable or suspend a package by the Admin.
    - (For a later release update): Ability to add, modify and delete packages by Admin.
  + Reservations:
    - Registration forms for students (including all student info.)
    - Track reservations made (students, driver, time, and car) assigned to the reservation.
    - Calendar Booking system to keep up with reservations and inventory of cars and drivers.

## 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 DriverPass system will require a web-based environment. Though the system will likely not be doing anything too strenuous such as hosting an online game, it will need to respond at a reasonable time to user inputs, amidst having a large amount of traffic from users to their online systems. Considering that the DMV normally doesn’t implement new updates but a few times a year, it would be safe to say that the DriverPass system need only be updated about once a month at the most.

#### 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?*

* Considering the desired goals of DriverPass, the platform that they should likely use would be a cloud-based platform. The owner of DriverPass was very adamant about being able to log in, access, and update the system from any device or mobile phone, and also did not want to deal with having to back up traditional servers and the security that comes along with it. Though DriverPass will definitely still need to have security features if they go with a cloud-based server, they will still find that this type of platform will meet their needs as a growing company.

#### 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 log in using a username and password approach, which should be case-sensitive to further increase levels. Users will be defined by different roles in the system each having different access and privileges in the system. Role types include:
  + Administrator
  + IT/Security
  + Reception/Teacher
  + Customer
* In the event of an issue with the system, or an attempt to access a high-privilege account the Administrator and security team should be notified immediately and the attempt should be added to the logs of the security report, indicating the time of the incident and the account in which there was an attempt on. A report of user log-in and changes to systems should be generated daily.

#### 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 account should have the ability to make changes to other users in the account. We should create an abstract class called user that will serve as the superclass for all user-type roles. From this class, we will create the subclass Administrator, which will have the privilege of changing user roles. When a user is first created they will automatically assume the role of a Customer User. A function should be added to the administrator role to change a user’s role on the front end without modifying the code. Functions should also be created to add users from scratch to a privileged role and to delete users entirely from the system using the Administrator account.
* Scheduled maintenance should be noted in advanced on the site and observed for platform updates, where the system should be made unavailable for most user roles except those deemed critical to updating the system. This should be done to prevent any errors, frustrations, or hacking attempts on the system during a time of possible vulnerability. Maintenance should be scheduled for times when there is known to be little user activity across the site.
* Access types needed for each role include:
  + Administrator & IT/Secuirty
    - Add/Delete/Modify user roles
    - Create/Reset Block user accounts
    - Add/Delete/Update Drivers
    - Add/Delete/Update Vehicles
      * Assign vehicle to a driver
    - Access and download all reports:
      * Report 1: Driver/Student Match records
      * Report 2: Teacher assigned vehicle records
      * Report 3: Changes to Records
      * Report 4: Security Logs
      * Report 5: Revenue Generated
      * Report 6: Package Numbers
    - Add/Delete/Modify all Reservations.
    - Add/Delete/Modify/Suspend Packages types
  + Reception/Teacher
    - Add/Delete/Modify all Reservations.
    - Access and download certain reports:
      * Report 1: Driver/Student Records
      * Report 2: Teacher assigned vehicle records
  + Customer
    - Add/Delete/Modify personal account reservations
    - Purchase package upgrade

#### 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?*

* In order for any user to log in, they must have the valid username and password associated with the account and the username must be checked upon creating an account to make sure that there is not already an account with the same username. Considering the databases will store sensitive information, two-factor authentication should be implemented to further increase the level of security associated with the system. Usernames, passwords, and other sensitive data (such as credit card numbers, and physical addresses) should be encrypted when stored. In the event of a breach, this will prevent hackers from doing anything with the data. TLS connections should be used between client and server to ensure privacy and secure data exchange.
* In the event that a password for any account regardless of role or privilege is entered incorrectly for a username, then it should show on the Security Logs, and that particular instance should count as a Failed Attempt to log in. After 3 consecutive Failed Attempts (3 unsuccessful attempts not interrupted by a successful attempt), then the account should be locked and prevented from any further attempt to access. The user will then have to contact DriverPass IT to reset their password in order to access their account, in which they will have to successfully answer a series of security questions to determine if the person who is attempting to access the account is indeed the owner of the account. Doing this will prevent hackers from attempting to brute force their way into accounts. Although they could very well try, they will not be able to brute force their way within 3 attempts. In the event, that a regular user forgets their password, there should be a “Forgot Your Password” link, that walks the user through the steps of recovering their 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 will allow users to log in to their personal accounts by inputting their username (email) and password.
* The system shall validate whether the entered username and password match the credential for the account.
* If the credentials match, the system will allow the user access to the account. If not, the system will deny access to the account, and count the attempt to access the account as a “Failed Attempt”. In the event of 3 consecutive Failed Attempts to access an account, the system will lock the account and prevent any further attempt to access it until this action is reset by a user with proper clearance.
* The system shall determine the type of user that is accessing the account based on the instance of the user that is using the account, which shall be predefined by a user with proper clearance.
* Upon accessing a user account, the system shall show the main page associated with the account. The main page’s contents shall depend on the type of role of the user.
* If the user is a customer, then the system shall show the package that the customer is enrolled in along with their current test progress and driver notes from their teacher if any.
* If the customer has yet to enroll in a package, then the system shall show the user the 3 package types along with a full description of each package to allow the user the opportunity to select a package that best fits their needs.
* Once a customer has enrolled in a package, the system shall allow the customer to make a reservation by inputting their information and requesting a date to schedule behind-the-wheel instruction with a teacher driver.
* The system shall allow customers the ability to Add/Modify/Delete their own personal reservations associated with their account.
* The system shall allow customer users to access and interact with practice tests to the point of being able to record their answers and grade their responses to answers when finished.
* The system shall allow users of the role of Reception/Teacher to Add/Modify/Delete all reservations.
* The system shall allow Reception/Teacher users to access and download records that match Teacher Drivers to Students as well as records that match Vehicles to Teacher Drivers.
* The system shall allow Admin/IT users to access and download all records and reports.
* The system shall allow Admin/IT users to Add/Delete/Modify user roles and user accounts. Add/Delete/Modify Teacher Driver and vehicles. And register vehicles to Teacher Drivers.
* The system shall allow Admin/IT users the ability to block and reset user accounts.
* The system shall allow Admin/IT users the ability to Add/Delete/Modify all reservations.
* The system shall allow Admin/IT users the ability to Add/Delete/Modify/Pause/Suspend package types.

### 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.)?*

* Customer User: the interface needs to allow users to:
  + Click a button or link to register for behind-the-wheel driving.
    - Fill out an online form in fields for the user to type in information (such as first name, last name, phone number, pickup, and drop-off location).
    - Click a button to submit the form upon completion.
  + The main page layout should show the user all scheduled reservations if any.
    - A link to modify or delete each existing reservation.
  + Click a button or link to take a practice test:
    - The ability to interact with the practice test to answer questions, and the ability to click a button to prompt the test to move to the next question or back to the previous question.
    - Click a button to submit the test for grading.
  + The main page layout should show the user the package that the user is currently enrolled in.
    - Have a link for the user to purchase an upgrade for a higher package.
* Reception/Teacher: the interface needs to allow users to:
  + The layout should have a search feature that comes with a drop-down bar to allow them to search for:
    - A reservation by student name, teacher/driver name, or confirmation number.
      * The ability to select a record by clicking on the existing record.
      * A link to Modify/Delete the record.
    - Vehicles assigned to drivers.
      * The ability to select a record by clicking on the existing record.
  + A button on or link to create a new driving registration for a customer.
    - Fill out an online form in fields for the user to type in information (such as first name, last name, phone number, pickup, and drop-off location).
    - Click a button to submit the form upon completion.
* Admin/IT: the interface needs to allow all of the privileges of the Reception/Teacher users as well as:
  + The layout should have a link to reports, so that when clicked it loads a new page that has a drop-down bar that allows them to choose from of the available reports.
  + Along with the search bar for the Reception/Teacher user, there should be the option to search for:
    - A user account by name or account number.
      * The ability to select along with a link to reset, modify, and delete the account.
    - A Driver/Teacher account by name.
      * The ability to select along with a link to modify, block, delete or assign a vehicle to the account.
    - A package type:
      * The ability to select along with a link to modify, block or pause a package.
  + The layout should include an area to Add a new item, that has a drop-down bar to select a list of database to add items to:
    - Add a new Teacher:
      * Fills out a form to add a new Teacher to the list of teachers.
    - Add a new Vehicle:
      * Fills out a form to add a new Vehicle to the list of Vehicles.
    - Add a new Reservation:
      * Creates a new Behind-the-wheel driving reservation for a customer.
    - Add a new Package:
      * Fills out a form to add a new package to the list of packages.

### 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 user has a computer or mobile device to access the system.
* The user has a stable internet connection to access the web-based services.
* The cloud-based services provided will adequately handle the size of the system.

### 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?*

* It is not possible to access and or download information from the system without an internet connection.
* The system will be down when performing system updates.
* The system will require a total of 15-16 weeks to build.

### 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.*

