# 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?*

* Our client, DriverPass, wants to develop an application that provide better driver training to pass the DMV’s driving test
* App must allow for user reservations and have those reservations accessible by staff and modifiable by users

### 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?*

* Provide online classes and practice testing for driving students to pass DMV driving test
* Provide on-the-road training for customers that will include practice tests and online classes
* System to be web-based comprised of multiple web pages, manage customer reservations, maintain user roles, and provide security for customer data as well as app integrity

### 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?*

* Download reports
* Create homework from reports
* Online and offline accessibility w/o creating redundancy
* Customer account creation
* IT, employee, and user role assignment
* Cloud-based managed by third party
* Reservation, cancellation, and modification tracking of user appointments
* Account activity report generation for employees
* System requires password-based security that is able to be controlled by IT and customer for resetting
* System receives input from customer for day and time of reservation through app
* customer must have access to create, modify, or cancel reservation
* App must track schedule to track vacancies for reservations
* Secretary can input reservation into system for customer
* System must link/assign a driver and a car to a customer reservation to avoid over booking
* IT officer has full access to all aspects of app
* System has 10 cars and 10 drivers available
* System has 3 packages to choose from:

“Package One: Six hours in a car w/ a driver”

“Package Two: Eight hours in a car with a trainer and an in-person lesson where we explain the DMV rules and policies”

“Package Three: Twelve hours in a car with a trainer, an in-person lesson where we explain the DMV rules and policies—plus access to our online class with all the content and material. The online class also includes practice tests.”

* Driving sessions are 2 hours long. Packages are broken up over multiple sessions
* \*\*System requires flexibility to be changed by the owner: add, change, and remove packages (To be implemented in future release. See item below)\*\*
* Current system requires ability to disable existing packages
* System allows for the following customer information to be input for customer account creation:

First name

Last name

Address

Phone number

State

Credit card number, expiration date, and security code

* Reservations should include pick-up and drop-off locations
* System requires being connected to DMV for rules, policies, or sample questions updates
* System should be alerted when an update to the rules, policies, or sample questions is received
* Employee/owner access to customer account page appearance to be based on customer supplied specifications showing online test progress, customer information, driver notes, special needs, driver photo, and student(customer) photo
* System should show tests the customer took, what’s in progress, and completed. Test name, time taken, score, and status should be displayed with status declaring “not taken”, “in progress”, “failed”, or “passed”
* A table that documents driver notes, lesson times, and start and end times is required for the owner to see
* System to have different pages: input form for customer information input, client contact page, and customer contact page

## 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?*

* Web and phone application-based
* RESTful API for fast and secure app performance
* The system should be updated at will by the client when they have a change they require and whenever there is a change in the DMV’s rules, policies, or sample questions
* Needs load bar for users to see that system is working if performance of site is slower than expected

#### 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?*

* Using proper RESTful API should allow for all platforms to interact with the site and app without issue
* Since DriverPass does not want to deal with security or back-ups, the system should be run from a cloud-based platform. My recommendation is DigitalOcean. DigitalOcean is directed toward small to medium sized businesses and offers secure block storage for fast data access and managed databases. It utilizes containerization ensuring consistency when ran on multiple platforms and offers many tiers of transparent pricing plans.

#### 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?*

* There needs to be a user database established to distinguish users and user roles, such as customer, employee, and admin.
* A user’s login must not be case sensitive, but the user’s password must be
* All user input should be converted to lowercase where case sensitivity is not deemed important to ensure smooth workings for the practice written tests.
* Errors should be logged to a server for analyzing as required
* An alert should be sent to the admin as soon as a problem is logged for corrective actions to be taken

#### 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?*

* There should not be a requirement to change code for modifying the user database. Admin and IT employees should have access to add, remove, or modify users and user roles in real-time
* The system should check for updates to the DMV’s policies and test questions either based on the DMV’s revision schedule, if there is one, or daily if there isn’t one, and should update any information in the system within 24 hours but during non-office/working hours to minimize system down time for customers and maximize data accuracy
* Platform updates will be handled by the cloud provider
* IT will need access to force system updates manually, to disable packages as required, reset user passwords and roles, and add notifications to the site/app

#### 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 user is required to have registered to gain access to the system creating a login ID and a password.
* Two-factor authentication will be a requirement for ensuring authorized access attempts and reset password requests
* If there are four consecutively failed login attempts, the account should be locked for five minutes, and user will be required to reset password
* Users should be able to click a “forgot password” link to reset password manually. The link should send an email to the address attached to the username with a link provided to change their password.
* If user contacts DriverPass directly, IT should be able to force a password reset for user
* Secure communication between customer devices and cloud servers by encrypting using TLS1.3 and using RESTful API with RBAC and rate limiting.
* If a brute force attack is perceived, the account should be locked, the user and admin will need to be notified, and the user will need to change 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 shall validate user credentials when logging in
* The system shall create reports in a format that works well with Excel
* The system shall be accessible online and offline w/o creating redundancy
* The system shall create user accounts with well defined roles
* The system shall create reservations, cancellations, and modifications to appointments
* The system shall log account activities
* The system shall manage user passwords
* The system shall allow users to modify their account information
* The system shall process customer payments
* The system shall maintain a schedule based on appointments
* The system shall assign drivers to customers w/o overbooking
* The system shall allow for customer package selection
* The system shall allow for packages to be disabled by admin
* The system shall update when the DMV updates their rules, policies, or test questions
* The system shall notify administration and maintain a log if errors occur
* The system shall allow the user to reset their password as required
* The system shall show customer test results and progress
* The system shall maintain driver notes and lesson times

### 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 UI needs to be logically and simplistically laid out for any level of user’s computer experience.
* UI needs links to allow users to register and reset their password
* The UI must be simple to navigate between pages using large tiles (Tiles are not required, but an example). The tiles should link to different pages pertinent to the user’s role.
* The customer role should have tiles for user profile management, scheduling appointments, contacting DriverPass, practice tests and test study materials, progress through the current selected package, and purchasing packages.
* The driver’s role should have show any drivers’ notes and their current scheduled students information, test progress, any special needs, and their student’s photos.
* The administrator’s role should have access to reset customers’ passwords, lock accounts, and change roles of other users. The owner’s role should allow for reports to be generated and have a view of the entire system.
* Users will interact with the UI primarily with mobile devices through a browser followed by a desktop environment through a browser. Interaction may be expanded to a mobile app in the future.

### 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?*

* I assume the system will be accessed in the future through a mobile app.
* I assume the system will sync seamlessly if used offline by the owner.
* We assume the DMV updates will not require manual changes to the system.
* We assume the DMV will cooperate with DriverPass to send notifications about test changes
* I assume users will have regular access to the internet

### 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?*

* Not all users will have the same performance experience based on their access to fast internet connections
* For adding or removing modules, a developer or analyst will be required
* Without a developer on staff, real-time troubleshooting will not be available

### 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.*

