# CS 255 Business Requirements Document Template

## System Components and Design

### Purpose

* DriverPass, the client, wants to provide driving training for customers to improve the pass-rates of driving tests and the success of customers.
* The system will provide online classes and practice tests, as well as optional on-the-road training.
* It will be accessible across multiple operating systems, and data can be downloaded into reports for the user to track their progress.

### 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 system must work on mobile devices as well as desktops, so it must be accessible across multiple operating systems. As it is being built, the client should have access to disable certain functions of the application if they decide they do not wish to use it.
* As far as security is concerned, hierarchy needs to be established and client-side managers must have the option to block employee access after termination.
* The app must track reservations, cancels, and modifications that their customers have created, and a report should be generated for the client when these actions occur.
  + Reservations are 2 hours long with 3 options: 6 hours, 8 hours, and 12 hours options. Each reservation is spread over 2-hour sessions depending on which option they choose. This requires the customer to input the date and time for reservation and should be posted through their account.
* An option needs to be made to call the office to schedule as well. The customer’s first and last name, address, phone number, state, credit card information, and pickup/drop-off location is required to make a reservation as well.
* Multi-factor authentication needs to be set up in case the customer forgets their password and the app must allow them to update their password when necessary.
* System needs to be cloud based to minimize technical management on the client’s side, so they can primarily focus on their business.
* A function needs to be developed to update DriverPass of any DMV changes so that their material is up to date and standard.

### Objectives and Goals

* This system needs to work effectively and efficiently to provide DriverPass’s customers with correct driving training.
  + Account creation
    - Personal details such as first and last name, birthdate, and contact information with optional profile picture.
    - Password update/reset option every 90 days here.
  + Appointment calendar
    - Should allow account holders to schedule reservations on the calendar and provide a notification the day of.
    - Selection of pickup/drop-off locations will be required.
    - Choice of reservation package should be given here.
  + Results
    - Display notes from instructor.
    - Display test results (final scores and specific missed problems)
    - Display reports and provide downloadable options.
  + DMV module
    - Display current events and up to date regulations.
  + Database for vehicles
    - Tracks reservations.
    - Tracks which vehicles are used.
      * Mileage, amount of time driven, maintenance
  + Allow for application updates.
  + Allow tiered employee access.
* To make this as efficient as possible for developers, here are the project checkpoints that need to be met by their respective dates:
  + February 18: UMLs created.
  + March 7: Research for User Interface Design.
  + March 9: Activity Diagrams created for each respective UML.
  + March 9: Class Diagrams are built.
  + March 11: Customer Approval.
  + March 24: Interface is built.
  + April 3: All data is linked to the interface.
  + April 27: Business Logic is finished.
  + May 7: System Test is finished.
  + May 9: System is delivered to DriverPass.
  + May 10: Concluding sign-off meeting.

## Requirements

### Nonfunctional Requirements

#### 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 operate on mobile devices so students can quickly access the application.
* It also needs to be cloud based so that staff (such as receptionists) can access it from a desktop.
* The system should connect with the office and schedule appointments within 5 seconds of submission from the user. Then, it should automatically update the user’s in-app calendar with the scheduled reservation.
* In general, the system should execute within 5 seconds from user input (calendar selection, reports, reservations, etc.)
* To start, updates should occur once every 2 weeks depending on the number of bugs, then once a month once all major bugs have been fixed.

#### Platform Constraints

* System will run on Windows, Google Chrome, AndroidOS, and iOS.
* A database of account holders, vehicles, and reservation packages will be needed to support the system.
  + For information storage and pull.

#### 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?*

* After downloading the app, the landing page will consist of two button options: “Employee” and “Guest”.
  + The landing page will only be displayed once when the app is downloaded. Once the user taps their choice, they will go to a registration page to enter in their information or credentials.
* When the application is accessed on a desktop, a login for employees will be separated from a login for clients.
* Usernames and Passwords will be case sensitive.
  + Passwords must have:
    - At least one uppercase
    - At least one number
    - At least one special character
    - At least longer than 8 characters
* Any problems should be reported to administration immediately.

#### 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?*

* Changes to the user can occur without changing code.
  + An “Edit” module needs to be implemented so that information about the user can be changed instead of erasing the code for that user entirely.
  + All edits (additions, modifications, and removal) should automatically update the database once submitted.
  + The system should be built to adapt seamlessly to any platform updates necessary. This includes storage.
  + IT administrators will need access to the database and the cloud to check any issues necessary. It’s possible they will need access to the server as well.

#### 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?*

* Multi-factor authentication will be used for the user sign-on. A combination of 2 of the 3 bullets below shall work.
  + Username/password combination
    - If password is forgotten, a code can be texted or emailed to the user.
    - 3 Security Questions can be answered.
    - OPTIONAL: A backup contact (I.e. a parent or guardian) can be given login credentials.
  + Face Identification
  + SMS/Email a 6-digit code.
* All data exchange should be encrypted.

### 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 the difference between employee and student login upon first login.
* The system shall validate login information.
* The system shall validate updated password information.
* The system shall allow account creation with contact information and profile picture.
* The system shall be accessible anywhere with wi-fi or cellphone data, or desktop connection.
* The system shall allow scheduling of reservations, selection of packages, and cancellations.
* The system shall monitor which vehicles are out via the database.
* The system shall display reports, notes from instructor, and grades.
* The system shall connect to the DMV for updated, accurate information.
* The system shall display all lesson details (length, curriculum).

### 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 needs to be stable enough to work on mobile OS and desktop browsers.
* The student needs to access the following:
  + Home page
  + Profile page
  + Reservation calendar
    - Includes package information upon scheduling.
  + Notes and reports
  + Help page.
* The employee needs to access the following:
  + Home page
  + Driver profile page
  + Reservation calendar
    - To approve, deny or schedule appointments (phoned in)
  + Notes and reports
    - To add notes and update grades
* The admin needs to access the following:
  + All database information
  + All package information (to update and edit)
  + Security access
  + All application pages.

### 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?*

* Internet providers will be in full service without outages.
* Cell-phone data carriers will be in full service.
* All reservations will show and follow through.
* All lesson plans will be completed.
* All users will have appropriate knowledge of all the technology needed to be used.

### 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?*

* Reservations cannot be made due to the number of drivers available.
* All technological requirements must be met.
* DriverPass budget and time

### 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 computer screen

Description automatically generated*