# 0CS 255 Business Requirements Document

## System Components and Design

### Purpose

* DriverPass wants to improve the availability of driver training for prospective drivers

### System Background

* DriverPass sees flaws within the training available from the DMV.
* The company aims to provide the training missing from the system in order to improve the likelihood of success in the final driving test from the DMV
* If the software is deployed properly, customers will have the ability to complete their driving tests and comply with all up to date rules of the DMV

### Objectives and Goals

* The customer must be able to access the data on their home device and view on MS Excel
* All users must have the ability to log in with different authorization levels
* Customers can make reservations
* Record all transactions
* Use the interface supplied during interview
* Customer can contact company and vice versa
* Connect to cloud

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

* Web-based, using cloud to host server
  + Actual choice will vary and will discuss with client over price, etc.
* Refreshes the home page every 30 minutes
  + While the student will be excited to learn the results of their test, a new appointment will be further than an hour away
  + The student already knows their own home information
* Tests should only accept inputs after at least five seconds of inactivity other than the mouse or scroll bar
  + This allows the driver to absorb the question

#### Platform Constraints

* File system is based on Windows OS due to its popularity and security
* A database will be required, using a CSV file for each driver
* Use and knowledge of cloud architecture is necessary to support ongoing maintenance to the larger system

#### Accuracy and Precision

* If any users change authorization level or other problem occurs, the admin must be automatically informed by email and text message of the error
* On the home page, there will be a login screen complete with ‘forgot password’ link
  + Both usernames and passwords are case-sensitive
    - equality is easier than ignoring case and more secure
* Each user will have separate authorization level with IT and big boss having the highest level.
  + Both can change entire system if needed
  + Secretary has level 2
    - Secretary can create, modify or delete appointments for all users
    - Can create password reset link for all users except IT and big boss
  + user has level 1
    - can create, modify or delete own appointments only
    - can create password reset link for self only

#### Adaptability

* When requested, the user can be created and added to the list of users with basic authorization at the behest of any level of authorization.
  + All will request the basic login credentials and password reset questions
* When requested, all users can modify or delete their own profile
  + Will require both a confirmation of their password and an onscreen confirmation of their choice
  + All users can alter or delete users with a lower authorization level than them without the lower users’ authorization, but will require their own password and onscreen confirmation
    - As previously stated, both the big boss

#### Security

* Each user logs in with a case-sensitive username and password
  + Will lock account with three failed attempts in half an hour
    - Can unlock with authorization from user with greater authorization or equal in the case of the big boss and IT
* Each user can reset their password with a ‘forgot password’ link close to the login area that redirects to a random security question set when the user was created
* In order to maximize security, a one-time code can be generated to be input and sent to the users using SMS or an existing authenticator

### Functional Requirements

* The system shall validate user credentials when logging in
  + Multiple authentication levels present in system
* The system shall allow users to download their personal reports
* The system shall allow users to reset password
* The system shall record and create reservations at the users’ behest
  + Each appointment shall be two hours long
* The system shall collect each user’s information when creating their profile including first name, last name, address, phone number, state, credit card number, expiration date, security number and pickup location for each appointment
* The system shall display three options when creating user profile:
  + Six hours with a driver
  + Eight hours with a driver and updated DMV rules
  + Twelve hours with a driver, DMV rules and an online class complete with a practice test
    - IT manager or big boss can close option when needed
* The system shall record modifications or cancellations of appointments
* The system shall link with the DMV to remain current

### User Interface

* The user interface displays the menu described by the interviewees
* On mobile, the separate modules shall be vertically aligned starting with the driver’s information
* On browser, the modules will be displayed as presented below
* The notes shall be presented in a table easily referenced
* Users include students, IT, and the secretary
* Students will see the table below
* Everyone else will see a search table first in order to find the student they will reference
* Once they get to that page, they will need the option to view and modify information in each section
* A computer screen capture

  Description automatically generated with medium confidence

### Assumptions

* The big boss does not have much technological knowledge, so we’ll have to create documentation and how-to’s to ensure that there are not any problems down the line
* DriverPass seems like a small business

### Limitations

* Due to its nature as a small business, the budget will be naturally constrained
* The temporal budget seems to be tight as well, doesn’t allow for much slack
  + We may have to work weekends in order to see it completed on time.
  + We must work quickly at first to create slack in the time budget

### Gantt Chart

Chart, timeline

Description automatically generated