# CS 255 Business Requirements Document Template

## System Components and Design

### Purpose

This system will address the void in the driver learning market. Currently, there are no options in the market to study and practice for a driving test. The system should also manage traditional on-the-road training. In addition, the system will enable online scheduling of appointments.

### System Background

The solution to this void in the market will be a web application that enables students (and/or customer representatives on their behalf) to schedule online driver training as well as in-person driving lessons. The students should be able to track their profile information. The system administrators should be able to export the system data and analyze it over time.

Administration, secretary and user accounts, and what they will be used for.

### Objectives and Goals

The system should be able to

* Enable the system administrators to access the data within the system anywhere online, as well as export data as needed for analysis in Excel.
* Enable an administrator to create, and thereafter, potentially disable different packages of the services offered.
* Offer two ways of scheduling appointments: over the internet or over the phone where a representative can type in the necessary information.
* Security
  + Different levels of access shall be available, including an overarching system administrator role.
  + Capability for users to reset their password.
* Provide alerts when DMV regulations are updated.
* Provide a home page with student profile information, and the status of the various tests and lessons they’ve taken.

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

* The application shall be hosted in the cloud via Amazon Web Services (AWS).
* AWS will make infrastructure updates automatically every Saturday at 2am. This configured day/time can be changed as needed.
* Standard online operations should take no longer than 350 milliseconds (Rogers, 2021). This performance figure can be configured and edited as necessary. AWS will automatically scale resources up or down to meet this performance criterion.

#### Platform Constraints

* The system shall use the AWS DynamoDB service to store all data necessary.
* Because it is a cloud-based solution, the application will run on any platform that connect to the internet utilizing a recent version of one of the following browsers:
  + Google Chrome
  + Mozilla Firefox
  + Microsoft Edge

#### Accuracy and Precision

* Each user shall be identified with a numeric UserID within a table within AWS DynamoDB.
* IDs and Passwords entered as part of registration shall be case-sensitive, allowing any characters found on a standard US keyboard, including symbols.
* The AWS CloudWatch service shall be utilized to monitor and alert the IT Administrator of any system problems.

#### Adaptability

* The IT administrator will have root access to the AWS platform. This will enable them to make changes to the application code as well as the data behind it.
* The IT administrator will be able to add/remove/modify users directly in the User table within DynamoDB, in case this is necessary.
* Platform updates will be automated, and any application issues resulting from such updates will be detected by the AWS CloudWatch service, and such notifications will be sent to the IT Administrator.
* The system shall be connected via API to the DMV. When the DMV makes updates to their rules, policies, or sample questions, the IT administrator shall be emailed automatically. When that occurs, management will determine any necessary steps (if any) required to alter the training and online practice tests.
* The AWS Backup service will be integrated to perform automated backups of the data in DynamoDB.

#### Security

* Users shall log in with a combination of a registered id and password.
* Upon registering, users will answer three security questions. If users later forget their password, they can reset the password upon correctly answering the three security questions.
* The connection shall be secured via https.
* The application shall lock all user accounts in the event that it detects a “brute force” hacking attempt. At this point only and administrative user may log in, and unlock user accounts, once the hacking attempt has been mitigated.

### Functional Requirements

* Registration
  + Required Customer Information
    - Addition of a customer shall require inputting the following fields:
      * First name
      * Last Name
      * Address
      * Phone number
      * State
      * Email
      * Credit card number
      * Expiration date
      * Security code
      * Package Selection
      * PickUp and DropOff Location
  + Phone
    - A secretary user shall be able to log in to the application, and add/remove/modify users, upon receiving a phone call from them.
  + Online
    - A customer shall be able to register (as described in the Security section, above), and input the registration information themselves.
    - The customer shall also be able to modify their information, or remove themselves from the system.
  + Packages
    - The various packages that shall initially be available are:
      * Package One: Six hours in a car with a trainer.
      * Package Two: Eight hours in a car with a trainer and in-person lesson, explaining the DMV rules and policies.
      * Package Three: Twelve hours in a car with a trainer, an in-person lesson, and access to the online class, including practice tests.
* On-the-road Training
  + Registered users with available training hours remaining shall be able to schedule a two-hour block based on a calendar that shall display available times and available instructors.
  + Once the user schedules a two-hour block, their available training hours shall be decremented by 2, unless they cancel that session before it occurs.
  + Secretary users shall be able to schedule these sessions on behalf of registered customers.
  + The scheduling calendar shall take into account the limitation of a maximum of 10 instructors, and should synchronize with those instructors’ own schedules, incorporating days off, etc.
* In-person Lesson
  + Registered users who have selected package two or three shall be able to schedule an in-person lesson based on a calendar that shall display available lesson slots.
* Online Class and Practice Tests
  + Registered users who have selected package three shall be able to initiate the online class as many times as they like. They shall also be able to initiate an online practice test as many times as they like.
* Reporting
  + Secretary users and administrative users shall be able to export data from the system into a csv file. Three types of exports shall be available:
    - Transaction Details: this shall include information on which users initiated which additions/changes/deletions of customer registration data, as well as changes to any scheduling data, including both driver training and in-person lessons.
    - Customer Details: this shall include an export of the data of all customers registered in the system.
    - Scheduling Details: this shall include details behind all driver training sessions as well as in-person lesson scheduling information per registered customer.
  + These users should be able to filter these exports based on a date range, as well as include active and/or inactive users, and then export the data.

### User Interface

* Header General Design
  + The header shall display at the top of each page within the application.
  + The DriverPass Logo shall appear centered within the header.
  + Login, Profile, Scheduling, Contact Us and Register links shall appear, right-justified, in the header.
* Registration
  + Upon clicking the registration link, the user shall be taken to a registration page.
  + The registration page shall enable the user to input the customer details for a new registrant.
* Profile
  + Upon clicking the Profile link, the profile page shall display allowing a user to update their registration details.
  + There shall be a search button (available only to secretary users and administrators) to search for a user, select a user from search results, and edit their details or delete the user.
* Reporting
  + Upon clicking the reporting link,
* Dashboard
  + Upon logging in successfully, the user shall be taken to the Dashboard page
  + For secretary and administrator users, the dashboard shall consist of various reporting options, discussed previously.
  + For customer users, the following panes shall display:
    - Online Test Progress (only displays for users who purchased Package 3)
      * This pane shall display a list of tests available, whether each was taken or not or is in progress, and whether the student passed or failed the test.
    - Driver Notes
      * This pane shall display all notes entered by the instructor for each of the training sessions taken by the customer, as well as the lesson time, start hour, and end hour.
    - Special Needs
      * This pane shall display miscellaneous notes input on behalf of the customer.
    - Images
      * A photo of the driver that provided the instruction shall display.
      * A photo of the student who is logged in shall display.
* Contact Us
  + Upon clicking the Contact Us link, the customer user shall be taken to a page enabling them to email DriverPass, as well as displaying the company’s phone number.
  + Secretary or Administrator users shall view a page enabling them to search for the phone number and email of a particular customer.
* Scheduling
  + Upon clicking the Scheduling link in the header, the customer user shall be taken to a calendar, enabling them to schedule/edit/cancel either an on-the-road training lesson, or an in-person lesson.
  + Secretary or Administrator users who click on this link would first encounter a search widget, enabling to first search and select a customer, before viewing the calendar for that user.

### Assumptions

* Data within the application should be accessible and editable anytime from anywhere that is internet accessible.
* Data can be downloaded to a user’s device when they are online, and the data can be analyzed offline (eg., using Excel), but any edits to the data cannot be processed unless the user is online.
* Data shall not be editable unless a user is accessing the system online.
* Internet browsers more than 5 years old shall not be supported.

### Limitations

* Initially, the three packages available to customers can only be configured by a system change. Eventually, these packages should be configurable by an application administrator without the need for a code change.
* After the initial release in June, DriverPass will need to negotiate with our company the timeline and costs for any enhancements to the system.

### Gantt Chart

Table

Description automatically generated

**References**

Rogers, R. (2021, June 29). How UCL migrated its Moodle virtual learning environment to the cloud in 10 weeks. Amazon. Retrieved September 12, 2021, from <https://aws.amazon.com/blogs/publicsector/how-ucl-migrated-moodle-virtual-learning-environment-cloud-10-weeks/>.

**Analysis of Potential Use of Process versus Object Model**

Process Modeling Approach

If we were to undertake a process modelling approach to our DriverPass application, I would structure it in the following way. The first process would be user registration, followed by the user logging in and viewing the dashboard page. Next the user could choose among three administrative subprocesses: update their profile, schedule an activity, or contact Driver Pass. The profile subprocess would result in the user editing or deleting their profile. The contact Driver Pass subprocess would result in the customer contacting DriverPass via email or phone. The schedule activity would lead to three other optional subprocesses: schedule a driving lesson, schedule an in-person lesson. Directly from the dashboard, the user could also take the path of completing an online lesson or taking an online practice test.

Object Modeling Approach

My preferred approach to Object Modeling is a UML diagram. I would create a User superclass from which the Customer, Secretary and Administrator subclasses would inherit. These would represent the various users applicable to the system. I would create a Schedule class from which the ScheduleDriver and ScheduleInPersonLesson subclasses would inherit. These would contain the data and functions related to scheduling these activities. Next, I would create a PracticeTest class and a Reports class, to represent the taking practice tests and the generation of reports.

Process Modeling Approach Compared to Object Modeling Approach

Choosing between these two models depends on what is most important to us at this point in the project: clarifying requirements with DriverPass management, or getting a head start on the technical design of the application. In the former case, the Process Model would be more beneficial. It is more business and action-focused, so it is more comprehensible to a more business focused audience. If we felt that business requirements were solidly in place, I would recommend the creation of the Object Model. This could then be used by the developers to start coding the application, by creating the various classes with attributes and functions, that would be needed in their code.