# CS 255 Business Requirements Document

## 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?*

* The client is DriverPass, a company who has found a void in the driver testing market.
* Students often only study from test questions, which can result in poor test taking and lacking knowledge to be well educated drivers. DriverPass aims to improve this problem.
* There is little out there as far as resources for helping drivers’ study and learn important driving information. It is the hope that by offering this to the customers, they can pass their tests and help improve the overall passed tests.
* The system will need to allow the users to take classes, practice tests, and provide on the road training. Doing this, it will help learning drivers find success.

### 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 should be able to help students learn their material and give them the proper tools for success when taking their driving test.
* The system will need to include:
  + Customer Account system
  + Business contact information
  + Purchase packages for access to resources
  + Access and role security for each employee
  + Download reports for those with proper access
  + Schedule driving lessons

### 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?*

* The system should contain a customer’s personal profile account:
  + The system will need to allow users to make accounts and take in their data.
  + The user should be able to register for a package, each one with different access to the sites features.
  + The user will need to be able to set up for on the road training, where they can set a specific time, date, and pick-up/drop-off locations. The system should save this information and give identify the driver who will be giving them their class.
* The system should enable the customer to access learning materials:
  + The system DriverPass wants should give access to drivers in training to help improve their success in learning and passing their final tests.
  + The user should be able to take classes and practice tests.
* The system will need to save scheduled classes and each of their driver’s information.
* The system should give be able to restrict and set access.
* The administrator should be able to download reports and have access from multiple platform types.

## 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 run on a web-based application, where the customer can access the sight from a computer or their smartphone.
* The system will need to be able to work fast enough to relay videos from the sight to the users computer, as well as update information about appointments promptly to avoid double scheduling.
* The system would need to be updated as new functionality is required and the business exhibits growth.

#### 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?*

* The system should run on Windows, as a web-based application, Windows has the largest support for servers, whether run in house or outsourced.
* The backend would need a database to hold user information, appointments, login access, and application history.

#### 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?*

* To distinguish between different users, there will be user accounts that will require a specific password to grant access. Input should be case sensitive, and the password can be used to form a specific ID number for a user to be identified as. This ID would never change from the user, even though the password may change.
* Different users will have different access, of which will require multi-authentication for administration privileges.
* When access to an account is done from a computer the account site doesn’t recognize, or the location is different than the typical user’s location, or when a password is failed too often, the account would need to be reverified, and the administrator would need to be informed if this type of access is attempted to privileged accounts.

#### 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?*

* An IT admin would need access to the back end of the system, in order to help troubleshoot issues, help unlock employee users from their computers, and full access to make updates as required.
* The system will need to have modularity, implementing a REST API will allow the system to make changes to the users without changing other portions of code and enable for adaptability to updates.

#### 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?*

* To log into an account, there should be a case-sensitive password that has a length requirement, special symbol, and numerical number. This will decrease the likelihood of brute force hacking. Double authentication, by sending a pin to the user through their email or phone when accessing it through a different computer or location, can help prevent further access, if the password has been guessed.
* The data would need to be encrypted to secure the transfer of information.
* If an account has been hacked, all account activities would need to be suspended, and the user would need to be notified. Security of the system would need to be evaluated and changed, along with password change updates for users.
* If the user forgets their password, they will need to provide detail of their identity, such as sending an email to the registered account, or having set up questions during account creation that would need to be answered correctly to have access and then immediately reset the 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 store important user account information.
* The system shall validate user credentials when logging in and lock an account if they can not be validated after a certain amount of attempts.
* The system shall watch for suspicious activity and report it to the administrator.
* The system shall be able to create driving practice lessons in person and relay this information to the driving instructor employee.
* The system shall allow users to update and change their account information, to add information, to change their paid subscription, and to delete their account.
* The system shall require scheduled updates to assist the system with changes to its required functionality. This means that users will need to be notified of system down time.
* The system shall be able to have different layers of access, where the administrator could access important system information, and the IT admin would need access to help users with their account and make changes to the system as needed.

### 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 well organized and easy to find information for the user.
* Menu options should be clear and explicit to where they lead and what functions they perform.
* Assistance for using the interface should be easily accessible on the main load in page of the system.
* The user should be able to access lessons, schedule lessons if they paid for that service, access their account, change their account information.
* Lessons should be organized and easy to search through.
* Practice tests should be easy to navigate and input answers.
* The user should be allowed to interact with the interface with their browser. Future updates could allow for mobile app that would allow for easy access to lessons.

### 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 am assuming that users have modern computers and smartphones that would allow them to access 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?*

* The time frame for each type of functionality that is required is a limitation, as too much time cannot be focused on solely one aspect.
* Budget will limit how many people can work on the project, the tools used, and the technology used.
* Limitations the system would have, is adding and removing modules in future releases, such as changing or adding to the list of packages a customer can buy.

### Gantt Chart

