# CS 255 Business Requirements Document Template

Complete this template by replacing the bracketed text with the relevant information.

This template lays out all the different sections that you need to complete for Project One. Each section has guiding questions to prompt your thinking. These questions are meant to guide your initial responses to each area. You are encouraged to go beyond these questions using what you have learned in your readings. You will need to continually reference the interview transcript as you work to make sure that you are addressing your client’s needs. There is no required length for the final document. Instead, the goal is to complete each section based on your client’s needs.

**Tip:** You should respond in a bulleted list for each section. This will make your thoughts easier to reference when you move into the design phase for Project Two. One starter bullet has been provided for you in each section, but you will need to add more.

## 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 purpose of this system design is to create a website to help people get their drivers licenses by providing driving training to individuals that need it. The Client, DriverPass, wants their website to: Hosts Classes, provide practice tests, and coordinate / schedule in person driving lessons. They want to offer three different packages to meet student needs: package one, two and three that provide an increasing amount of time behind the wheel with a driver. The third package also includes online tests, and lessons. They eventually want to be able to customize these plans to tailor to each specific student need but that will be put on the back burner for now.

### 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?*

* Ultimately, they want to help people pass their driving tests.
* DriverPass’s system needs:

1. Some type of user database with different rights that can store information.
2. Admin rights (Ian and Liam)
3. Scheduler rights (Secretary)
4. What about driver / instructor rights? I think this should be presented because drivers are going to have notes of their own and users should have the ability to review them for their own self-improvement. It already is implied that they will have their own type of right based on the image they provided. This may need to be clarified.
5. User rights (Those that use the service)
6. Can record and time stamp changes made.

Questions:

* Registering should happen over the phone. Is this really the only way they want people to be able to register for their courses? I think this needs to be clarified.

1. Some type of Master Question File, a way to present questions in a meaningful way, record test results and associate them with users and can flag out of date information if it gets updated by the DMV.

Questions:

* Having the ability to flag stuff if its updated by the DMV may need a little bit more clarification. This doesn’t exactly seem straight forward, especially if its expected to happen in an automated manner.

1. Some type of database containing online courses and a way to present course content to site users.
2. See comment above for having the ability to flag out of date info.
3. Some type of Database of Cars and Drivers plus a way to schedule driving sessions with drivers, students and cars.
4. Amount of time someone can schedule in a car with a driver is based on what package they purchased.
5. The scheduling portion of this needs to stamp who made reservations and when.
6. This needs to be able to interact with user profiles to determine how much time they’ve got remaining on their driving package.
7. A way to provide additional training to those that need it.
   1. Implementing the suggested tiered system allows customers to get the extra training they need, if they need it.

### 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 goals for this system are to improve students driving skills and abilities to pass their driving test. To meet this goal, this system will step thru the following objectives:

1. Provide minimum required training through the offered packages.
2. Facilitate the scheduling of driving sessions
3. Provide professional feedback to students through helping identify and correct problem areas.
4. Provide additional training through more hours behind the wheel with an instructor, if needed via second package.
5. Provide access to resources to further assist students if additional hours aren’t the only thing needed via third package.

## 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?*

* It appears that DriverPass needs to be able to function in a general web browser. From a business perspective, it would make sense to offer a web application that functions on major web browsers such as Chrome, Edge, and Safari first while expanding horizons to additional browsers later. Doing this makes the system available to the majority of users immediately.
* There was no specific requirements as far as system performance. With that being said, system performance requirements should fall under the category of: standard budget desktop computer, whether PC or Mac, or standard Android or Apple iPhone.
* The system should be updated as required to meet DMV policy.

#### 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 platform that this specific website should be expected to run on include either Windows, MacOS, iOS, or Android. As for Rationale for this, see above justification for web browsers. These platforms utilize these web browsers which is what DriverPass will run on. Support for Unix or Linux and their web browsers should come later.
* DriverPass will require a small database for its customers and its online courseware. While the database likely will not need to be large, having a small database to maintain its customer records on, in addition to hosting its courseware, will be necessary.
* DriverPass may require an API to interact with local DMV to update courseware and testing materials as required. While an API may not be necessary to start out with, it will automate the process of updating courseware and testable materials.

#### 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?*

* Different users should be distinguished by different usernames with their own unique passwords. Different types of users, whether a student, scheduler, or administrator, should be distinguished at the time of registration
* Usernames and passwords should be case sensitive.
* System should notify administrators of problems during normal business hours unless malfunction is dealing with exploitation of sensitive customer information such as payment methods. Then administrators should be notified immediately, regardless of time.

#### 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?*

* Updating user information should be done without modifying code. This will avoid additional labor needed to update basic user information.
* Platform updates should not be a factor with DriverPass as it is a website and will function on most standard web browsers.
* The Administrator will need to have user account access to perform account or password resets and to block privileges if needed. They should also have access to time stamped information to assist in account troubleshooting. This will ensure that system integrity is maintained and also need to know information is

#### 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?*

* Users are required to log in using their case sensitive username and password
* If users forget their username and password, they should be able to reset it using some form of information they’ve provided like their email or phone number. Otherwise System administrators should be able to reset it.
* Securing the connection between the client and server can be done thru limiting who has access to it at any given time. The registration process is an example of limiting access: Only someone with specific system rights can sign people up for the service. Otherwise, once a user has a user name and password that meets the criteria, they should have access to their information.
* If a hacking attempt is made and an account is compromised, as soon as its recognized as a malicious activity the account should be either disabled, suspended, isolated somehow so that corrective measures can be applied. Admins, and users should be notified of nefarious activity.
* Other security needs should be delt with by the cloud hosting service, per Liam’s request.

### 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 make data reports using a file type that is readable by major spreadsheet applications. This is specifically requested by Liam.
* The system shall identify who has made changes within it and time stamp changes made for the purpose of accountability. This is specifically requested by Liam.
* The system shall facilitate the coordination and scheduling of driving appointments based on availability provided by students and drivers. This is implied throughout the interview.
* The system shall be able to discern which packages customers have registered for, and limit availability if required. This is implied throughout the interview.
* The system shall facilitate the registration of new users, associating information provided by the user with their account. This is implied throughout the interview.
* The system shall accommodate an API that interfaces with the DMV to update courseware and testing material, as required and if requested. This is implied through the interview, however implementation of this functional requirement is dependent on what client deems necessary.

### 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: Display student information that is stored on the server in a similar fashion to the layout provided by the DriverPass team; allow modification of information stored on the server pertaining to students, so long as who is modifying that information has user rights to do so;
* From the information gathered, there will likely be four primary users of this system: Students, drivers, administrators and schedulers.
* Students should be able to: Have limited administrative functions such as resetting passwords and change certain account information provided during registration, like pickup location for example; See driver notes that were given during previous driving sessions; See testing progress and their provided statuses if they’ve paid for the package; List out any special needs they may have; Post a picture of themselves; schedule an appointment to drive.
* Drivers Should be able to: list their availability; list a professional photo of themselves; input or edit their own notes taken during student driving sessions; view previous driving writeups from other drivers;
* Administrators should be able to: Perform administrative functions such as reset passwords, change account information, disable, suspend or isolate any user account; register new students, drivers and schedulers; View name and time stamped changes; Perform IT duties such as interact with the server that is hosting the webservice; update data related to driving courseware being taught if DMV policies change; Have the same interaction with the interface as students; Have the same interaction with the interface as Drivers, possibly ;
* Schedulers should be able to: Register new students, set up a driving session using driver’s and student’s listed availability; Have similar interaction with the interface as students, except for resetting passwords.

### 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?*

* *All u*sers will have internet access to use the service or make changes to the data within the databases. This was illuded to during the interview, specifically when Liam was pressed on updating the database while offline and having that would lead to data redundancy.
* *All* platform maintenance, to include server maintenance, will be accomplished by the hosting company. This was illuded to during the interview, specifically when asked about the interface. Ian did not want to deal with security, information backup or maintenance.
* DMV will provide specific updated information that will be easily recognized and transferred over to existing coursework by an API. This is a *big* assumption that, if proven false, will lead to a large resource drain. This was illuded to via the interview when Jennifer asked about compliance with DMV policies. Liam sounded fairly convinced that there would be such an API.

### 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?*

* Being an online only environment is a significant limitation, especially for the elderly. Elderly are historically resistant to technology and the implementation of it where there normally was no implementation prior. There is nothing saying that DriverPass specifically was for initial issuance of drivers licenses.
* Registering for DriverPass only through phone call could be a large limitation as younger-generation individuals are used to accomplishing things through a technological intermediary ( via an online application).
* There was no budget or time requirement specifically outlined by Liam.

### 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.*

[Insert chart]