# 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 the project is to create an online system that can be accessed via internet connection that will provide student drivers study material, online courses, one on one training with a driving instructor and many other offerings that will improve the passing rate of driving students.
* The client is DriverPass, a company that wants to increase the pass rate of student drivers by providing them an online system where they can study, schedule training, and keep up to date with DMV rules and regulations.
* Driverpass would like all the above criteria to be accessed from anywhere with a secure internet connection, preferably run over the cloud.

### 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?*

* DriverPass wants to create a system that will help driving students pass their tests.
* The system DriverPass wants to implement has many different components:

-Online Test Progress

-Driver Notes

-Special Needs

-Driver Photo

-Student Photo

-Driver Information

-Reservation System

-Password Reset System

-Location System

### 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?*

* DriverPass wants to be able to access data from anywhere using a computer or mobile device
* They also want to be able to download reports that are accessible to work on from home using Excel and other software.
* They want to implement a reservation system for users to reserve a time and date for driving lessons.
* They need to be able to track who the user is reserving a lesson with and for how long.
* They need to give users options to be able to reserve 1 of 3 driving packages and block access to any driving package that is unavailable.
* The system needs to be able to automatically reset a users password if the password is lost.
* The system needs to be able to take user information and location for picking up users for driving practice
* System needs to be connected to DMV so that they have a notification when policies are updated.

## 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?*

* Specifically, Ian proposed that DriverPass would like their system to be “run off the web, preferably over the cloud.”
* Since DriverPass would like their system to be accessible anywhere and wants it to run off the cloud, I propose that they investigate using a cloud-based system like AWS, or Google Cloud.
* The system should be capable of loading pages and handling user interactions with a response time of less than 5 seconds. Having a strong, dedicated internet connection could possibly shorten the time to less than or equal to 2 seconds. Considering that they would like the system to be accessed from anywhere via the cloud, less than or equal to 5 seconds in response time should be adequate for system use.
* How often the system should be updated depends on the different functions of the system. Consider security for example, updating security systems should be done regularly especially if there is an issue. Contrast this to DMV rules and regulations. There may not be many frequent changes to DMV rules and regulations so updating this part of the system should be done either bi weekly, monthly, and as needed.

#### 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 be able to run on windows, macOS, iOS, Google Cloud, and AWS. This should be sufficient for most if not everyone to be able to access the system on their mobile devices, PCs or even local library computers.
* The system does need a back end database in order to store information such as user information, course materials, DMV rules and regulations, progress tracking, training schedules etc.

#### 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, roles will be assigned during account creation for the system. The user can be prompted if they are a student, instructor, or administrator.
* Email credentials for logging in are not case sensitive, however when creating a password, it should be case sensitive.
* The system should inform administrators of issues when it affects the performance of the system, issues regarding users, and any issues revolving around security. These issues need to be addressed as soon as possible to ensure the system runs efficiently and that users are happy with.

#### 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?*

* The system should be able to make modifications to users without changing the code because the Users shouldn’t be part of the systems source code. All User information should be handled on a separate database that can be modified by admin without the need to change any of the systems code.
* The design of the systems should be dynamic and adaptable. Having versions of the application that the system runs on would help ensure that updates won’t crash the system, by testing the update prior to implementation. Or if the new update creates issues in the system, you can revert to an older version to ensure that the system is still accessible while the team fixes bugs in the new patch.

#### 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?*

* The requirements needed for a user to log in to the system would be comprised of 3 steps. User creates an account with their email and creates a case sensitive password for the account. The system will then send a verification email to the email the student signed up with to verify they are the owner of the email account. Finally, the system would ask if the user would like to set up multifactor authentication to ensure that only the owner of the email and phone is able to log in using the credentials provided.
* To secure the connection between the client and server, the system would store data and connect through a secure cloud service like AWS. Data would be stored in an encrypted storage system, and security inspection of the system would happen frequently to catch any issues or hacking attempts.
* Should a brute force hacking attempt occur, further log in attempts would be suspended for a specified amount of time. Simultaneously, an email would be sent to the email address where the account was used to sign up with, informing the user that there was an attempt to log in. The email would then ask the user if this was them trying to attempt to log in, if so, it would ask the user if they would like to change their password.
* In the case of a forgotten password, the user would submit a password reset inquiry by submitting the email they used for signing in to the account. Following this, an email would be automatically sent by the system with a timed password reset link. The user would click the link which would proceed to a page where the user would be able to input a new password, which they would confirm by re-entering. This new password would follow the same instructions as when the first password was initially created (Case-Sensitive, must contain one upper case letter, one lower case letter etc.)

### 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 offer driving packages based on DriverPass’ package offerings.
* The system shall dynamically change when a certain package(s) is still available.
* The system shall store all information using a secure and encrypted database.
* The system shall be accessible on different OS.
* The system shall provide study materials based on DMV rules and regulations.
* The system shall stay up to date with current DMV rules and regulations via bi weekly updates.
* The system shall provide instructor availability for one-on-one driving sessions.
* The system shall access student location only when it is time for instructor/student driving sessions.
* The system shall provide notifications on when grades are posted.
* The system shall provide a list of times and instructors for the user to choose from to reserve a driving session.
* The system shall differentiate between the different roles of users who are in the system.

### 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 easily readable and user-friendly. It needs to display the DriverPass logo. It needs to have display links to separate pages such: Online test progress, user information, driver notes, special needs, driver photo, and student photo. Upon clicking these links, it will take them to a new page that corresponds to what the user clicked on. The interface also needs to display separate links for the different roles of users such as instructors needing to post grades, whereas this would not be part of the user’s interface.
* The different users for this interface will be the students, administrators, instructors, and IT personnel.
* If the user is a student, they would need to be able to do many different actions such as: create log in credentials, check online test progress, check driver notes, reserve a driving instructor, contact a help desk assistant, read current DMV regulations and materials, complete online driving prep tests, access to IT help desk.
* For Instructors they would need to have the ability to: access student tests, input grades, access student information for driving session pick up, access to IT help desk.
* For Admins: input student information upon registration, uploading driver/student photos on user dashboards, access to student registration, access to payments made through the system, access to packages offered by the system access to IT help desk.
* For IT: They would need access to all levels of user roles as well as access to the systems infrastructure.
* The user would be able to interact with the interface via mobile and browser.

### 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?*

* Some of the assumptions made are as follows:

-the user has a basic understanding of navigating a web browser/application

-the user has at least one of the platforms that the system is available on

-the system can easily pair with DMV’s website to reflect the DMV’s current rules and regulations

-the DMV wants to collaborate with Driverpass with the implementation of the DMV’s rules and regulations

-Admins will not forget their password or have log in issues as well.

### 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?*

* One of the biggest potential limitations I could see from the system is scalability. The system is going to be fluctuating between varying amounts of users at any given time which could affect the overall performance of the system.
* Another limitation is the number of available driving instructors there are for one-on-one sessions. If this system is supposed to be nationwide, we’d need staffing, system structure, and funding to hire thousands of driving instructors.
* Not only is the amount of driving instructors a limitation, but the availability of the instructors as well. There will be a larger number of students to instructors, meaning depending on availability, it may take a long time to book a session with an instructor.
* Our biggest limitation is that the user must be able to have access to the internet to benefit from this system. Users who don’t have regular access to the internet will not be able to use the system.
* Budget is another limitation, using a cloud based system can get expensive depending on the amount of data storage that is needed to have the features that Driverpass wants to offer.

### Gantt Chart

*A screenshot of a project

AI-generated content may be incorrect.*