# 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?*

* Client: DriverPass
* Problem: Too many people fail their driving test when they go to the DMV.
* Solution proposed by DriverPass: DriverPass wants a system that will allow drivers to take practice tests online and review up-to-date materials from the DMV so that they can pass the driving exam. They also want a way for students to schedule in person driving sessions with experienced driving instructors so that they can practice the actual driving portion as well as the written exams.

### 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 feels that drivers often fail the driving test as they are not familiar with DMV regulations or that these may have changed since the drivers last took the test.
  + To fix this DriverPass wants the system to be able to synchronize with the DMV’s database to provide up-to-date testing information, such as new rules, practice questions and new regulations so drivers have accurate information to assess their knowledge against.
  + DriverPass should receive a notification when this information is updated.
* The owner may need to review the information offline or generate reports.
  + To help with this they would like to be able to download the information so they can work on it in Excel or other similar programs.
* Security and roles
  + DriverPass has requested a variety of levels for distinct roles in case someone forgets their password or in case an employee is let go they need to be able to retain control of information at an appropriate level.
* Packages
  + Currently offering three different packages, in future want to be able to customize these.
  + Package 1
    - 6 hours in a car with a trainer (3x 2-hour lessons)
  + Package 2
    - 8 hours in a car with a trainer (4x 2-hour lessons)
    - In-person lesson where DMV rules and policies are explained to the student.
  + Package 3
    - 12 hours in a car with a trainer (6X 2-hour lessons)
    - In-person lesson where DMV rules and policies are explained to the student.
    - Access to online class with all online content and materials
    - Practice tests
  + Scheduling for each lesson may be separated and taken at separate times till it equals the package that was bought.
  + Ability to cancel or pause registering for packages without needing a developer to modify the system to remove the visibility of the package option.
* PII
  + The company will securely store customer information.
* Communication
  + Two-way communication ability built into the system to allow DriverPass and students to communicate directly in the portal.
* Future features
  + For now, DriverPass wants to focus on these base features first and revisit possible new features at a later time once these have been completed.

### 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?*

* Roles:
  + Liam - “big boss”
  + Ian – IT
    - Administrative role (modify data, reset password, maintaining system)
  + Secretary
    - Able to modify/set appointments.
  + Users
    - Can modify/set/cancel own appointments only, all done online. Can automatically reset password online.
* Tracking
  + DriverPass wants to have a straightforward way of generating reports for scheduling so they can have a paper trail and determine who canceled an appointment and address it properly.
* Scheduling
  + DriverPass needs a way of seeing:
    - Customer
    - Time requested for lesson.
    - Date requested for lesson.
    - Driver assigned to customer.
    - Car assigned for lesson.
    - Ability to schedule appointment in variety of ways (online, call office, or visit office)
* Cloud-based
  + Backup and security should be managed independently of business operations by a third-party, should provide technical capabilities with little direct input from DriverPass.
* Design
  + Should be like design presented during interview with Liam and Ian
  + Displays
    - Tests customer has taken.
    - Shows Tests that are in progress.
    - Test fields
      * Test Name
      * Time taken.
      * Score
      * Status (not taken, in progress, failed, or passed)
    - Driver notes
      * Lesson time
      * Start hour.
      * End hour
      * Driver comments
  + Registration
    - Completed by student or secretary on student’s behalf.
      * First Name
      * Last Name
      * Address
      * Phone Number
      * State
      * Credit Card Number
        + Expiration Date
        + Security Code
      * Pickup/drop-off location where customer desires to be picked up for lesson.
        + Should be same for pickup and drop off.
  + Communication
    - Page for students to contact DriverPass.
    - Page for DriverPass to contact student.

## 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?*

* Application – Most students are likely to connect to the service or will want to connect to the service through their phone as this can be done when they are on the go, on a break at work, or even using this to review materials before they take a driving exam.
* Web based – Secondary, there will be some users that opt to have a web-based approach as they are more familiar with this method and want to have time that they feel they will be less distracted to study. Some users may feel if they are on their phone that they will be tempted to browse social media and not to properly pay attention to their studies.
* Load times – For both web and application approaches screen load times of less than 2 seconds for a full screen load are ideal. This will maintain that DriverPass is a mature system and that they can put their trust in the system. Load times may be able to be reduced by limiting large content or compressing it depending on the type of device that the user is connecting to. Also loading text before loading images or videos is another customary practice and one that many consumers are familiar with.
* Updates – Updates should be maintained periodically. From a server side they should undergo routine maintenance at least once a week to clear the cache and prevent errors or slow loading times for users. If the system begins to become strained this may need to be revisited and adjusted. Ian should be easily able to help maintain the server and keep the backend running without needing to rebuild the complete system each time. For larger updates where new content is added will need review later, but once the system is completed may be done on a monthly – bi-monthly basis. Updates can also be used to help encourage users that the system is being well maintained and is being added to frequently so they will feel their money is being well spent.

#### 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?*

* Platforms – Wider platform usage will increase the number of potential customers that DriverPass can target so a wide range is ideal. However, because of cost and time target 2-3 may be best initially. To this end, Android, Windows, and Apple would be best to target initially. Android has one of the largest ranges of user bases of any smartphone architecture, Windows is seen as the default operating system on most standard computers, and Apple products will work on both Apple iPhone and Mac Books. This will allow for the targeting of virtually all of the market. If only two could be performed Android and Apple are close in design and should work well.
* Back-end – This application will need to have a database connection as it will need to store user information (logins, test performed, rides scheduled), as well as connecting to the information for test questions and the schedule for drivers as well as the cars that they will be in.
* API’s – This application would ideally be able to connect to a publicly accessible API endpoint from the DMV, this will help facilitate communication with the DMV’s most up-to-date rules and procedures.
* Data entry automation – In the event and API endpoint is not available in order to keep Ian or others with DriverPass from having to babysit the system and automated process will need to be implemented that new rules and regulations can be fed into the database and these are automatically converted into the databases format so it can be used with existing questions provided by DriverPass. This option will take an increased amount of time, and could require revisiting if the DMV substantially changes the manner in which they put out information about new changes to rules and procedures.

#### 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?*

* Users types – Public users will be offered to register an account through a public online portal on the app or site. Administrative roles will be added to the database directly by Ian through the backend portal. This will provide Ian with the ability to control who has administrative access to the application, and also allow him to make changes if there are changes in employment status.
* Case-sensitivity – The site will be case sensitive as this can help to add another layer of security for users particularly with their passwords.
* User login – Each user will be prompted to create a new account the first time they access the portal and will use this login to keep their unique information separate from all other users.
* Error handling – The system should inform Ian of an issue as soon as it arises. This should be able to be changed so that the system will send an e-mail notification to Ian once a problem is detected so that he can know what is going on and can handle it at his earliest convenience.

#### 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?*

* Update user – Backend portal will have the ability to allow Ian directly modify user information without needing to code this in. This is frequently done through a backend API endpoint that is exposed only to admins. This will allow him to directly add/remove/modify user information on a business process side this should only be done in the event that someone is terminated, a request to delete user information is submitted to be in compliance with Consumer Data Privacy laws, or in the event that an admin forgot their password. Users forgetting their password should be automated as this can happen at random times and may be at a time that Ian is not in office to address the problem.
* Updates – The platform itself will need to update information as it becomes available. As such having the data stored in a back end database that is then presented to a front end application is a standard practice that would lend itself well to keeping the system updated.
* IT administrators access – Ian can perform most all of the roles needed by simply having a back-end administrator access. This will allow him to make sure that the database is correctly updating, and to modify user information as the need arises.

#### 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?*

* User login – User will login using their email and a password that meets a criteria of being at least 16 characters long with at least one upper and lower case letter, a symbol, and a number.
* Secure connection – In order to keep the connection secure standard IT security principles will be followed such as requiring a public and private security key in order to decrypt messages between the user’s machine, and the server hosting the application.
* Hacking – In the event that an account is found to be hacked it will be frozen till the user can verify that it is them, in the meantime the account logins will be reset and will only be revealed to Ian. After proper verification the user will be provided with a link to reset their password so they can regain control of the account.
* Password reset – Should a normal user forget their password and their account has not been hacked they may reset their password themselves through links on the site. If an administrator user forgets their password they will need to speak to Ian and he will issue a manual password reset and provide them with new login information.

### 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 validate user credentials when logging in.
* The system shall provide a learning platform for users with up-to-date DMV rules and procedures.
* The system shall allow users to schedule driving time with instructors.
* The system shall allow users to modify or cancel drive time as needed.
* The system shall allow users to communicate with DriverPass.
* The system shall allow tracking for administrative purposes to determine who canceled an appointment and allow further follow-up as to why.
* The system shall have multiple levels or user types so that varying levels of reports and use can be maintained.

### 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 user interface needs separate sections for Online Test Progress, Driver Notes, Information about the student, Special Needs, Driver Photo, and Student Photo.
* The primary user will be the Students, there will also be a user type for Liam that has full access to the system to generate reports, one for Ian so that he can monitor the system and issue new administrator accounts and maintain the database, and one for the secretary that can modify student’s driving schedule for them if they call in or come into the office.
* Most users will likely connect through a mobile application, but some will still connect through a browser like administrators (Liam, Ian, and secretary).

### 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?*

* The system will need to be up 24/7, this is due to students accessing the system at various times to take tests.
* Since the system will need to be up near constantly, any updates will need to be short, under 30-45 minute downtime would be ideal and at times when the fewest number of users log into the system, this is usually very early in the morning at 3-4.
* The system will be cloud based, this means that DriverPass has room to grow and does not have to worry about directly managing the physical servers themselves.
* Most user will connect to the system through mobile devices.

### Limitations

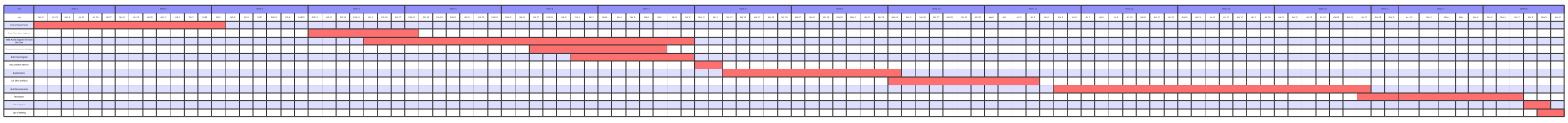
*Any system you build will naturally have limitations. What limitations do you see in your system design? What limitations do you have as far as resources, time, budget, or technology?*

* The system will have a maximum amount of content it can have on any given page before load times are affected.
* This system will be completed and delivered by May 9.
* Final sign-off of the project will be completed by May 10.
* The system will be cloud based so cloud provider may further limit specific aspects of the project depending on what host is chosen.
* As the budget is not unlimited, care must be taken to keep within DriverPass’s budget constraints. As this seems to be a new venture this is especially poignant.

### 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.*





I included both the screen shot and pdf as an attachment as the screen shot is very small and difficult to see fully.