# CS 255 Business Requirements Document Template

## System Components and Design

### Purpose

The purpose of this project is to create a user-friendly and robust system for DriverPass so that they can simplify the process of learning how to drive. The client, DriverPass, aims to provide its users with a comprehensive and easy to use tool that can be of use to new drivers that have a variety of different experience levels. To achieve this, the program will offer a wide range of learning packages to its users. These packages will include:

* Online educational resources for tutoring.
* In-person road test scheduling
* Practice assessments so students can prepare themselves for the real thing.

The end goal is to create a system that combines online resources with in-person testing and practice assessments for a seamless and easy to use learning tool for new drivers.

### System Background

The system should offer a wide range of functionalities and curriculum and serve as a hub that offers users a platform where they can access all the necessary tools and teachings to help them pass their in-person driver’s test.

* The system will offer users a “one stop shop” where they can learn all the necessary skills to pass the in-person driver’s test.
* The system will accept students with a wide range of knowledge and skills and should adjust accordingly.
* The system should be user friendly and easy to navigate.
* The system should be kept up to date with DMV standards.
* The system will allow users to make personalized accounts.
  + From here, they can make reservations online or over the phone.
  + They can access online training materials.
  + They can access online tests.
  + Driving packages will be available in 2-hour time blocks.
    - The packages will be available to be pre-purchased and scheduled.
* The system owner will have administrative privileges so he can manage the system and the modules that are available. The system owner should have these privileges:
  + They can disable modules that are no longer available.
  + They can add modules or adjust existing modules to line up with DMV standards.
  + They will have the ability to block employees who no longer require system privileges.
  + They can give system privileges to the employees that need them.

### Objectives and Goals

The objectives in a system analysis will offer the development team direction and allow them to develop a strategy for developing the system. It outlines the desired outcomes of the DriverPass system and highlights what functionalities are important. The goals provide measurable targets that will help the development team turn objectives into completed tasks.

**Objective 1** **Seamless User Experience:** The first objective is to ensure that users have a seamless user experience from account creation to reservations and testing.

* **Goal: Easy account creation**
  + For this objective, the goal is to create a user account creation system that is intuitive and easy to use.
    - **Measurable**: The time that users spend on creating accounts should be minimized, the less time it takes will be an indicator that the system is efficient.
    - **Measurable**: We will also gather user feedback at the end of account creation that asks new users how they think the process can be improved upon.
* **Goal: Simplicity of reservations**
  + Another goal for this objective is to simplify the reservation process, both for reserving a testing vehicle and for setting up testing times.
    - **Measurable**: Monitor the time that it takes for users to make reservations, this will gauge the user-friendliness of the system.
    - **Measurable:** Collect user feedback regarding the reservation process so we can improve its efficiency.
* **Goal: User Feedback Integration:**
  + The last goal for this objective is to accurately incorporate user feedback to improve upon the user-friendliness and efficiency of the DriverPass system.
    - **Measurable**: Constantly collect, compile, and review user feedback regarding all areas of the DriverPass system so that we can be continuously improving the user experience.

**Objective 2 Student Progress Tracking:** The second objective will be to provide students with a reliable way of checking and tracking their progress with the learning modules. Allowing them to monitor their progress to achieving their driving goals.

* **Goal:** High user engagement with the progress tracking system
  + For this objective, the goal is to get as many users to engage with the progress tracking system as possible. The number of users that engage with the system will indicate the effectiveness of the system.
    - **Measurable**: We will monitor the engagement levels of the progress tracking system; high user engagement will indicate that the system is well made and comprehensive. Low user engagement will indicate that the system is not well made or not user-friendly enough.

**Objective 3 User Privacy and Data Security:** The third objective will be to ensure that user data is kept private. Since sensitive user data will be stored on user accounts, we need to be able to guarantee the privacy of both the users and employee data within the system.

* **Goal**: Data Security Assurance
  + The primary goal is to maintain a high level of system security using routine checks and evaluations.
    - **Measurable**: Frequent security assessments will allow us to ensure that there have been no breaches or leaks of user and employee data.

**Objective 4 Remote Accessibility:** The fourth objective focuses on the need for remote access to provide flexibility and convenience for both students and employees.

* Goal: Efficient remote access
  + Users should be able to access their accounts and all the curriculum within the system on their mobile devices as well as their desktop computers.
    - **Measurable**: Review user login patterns to determine when and where users are accessing the system. If the system is optimized for both mobile devices and computers, user access data should be roughly even between both platforms.
    - **Measurable**: Ask students and employees what their preferred platform is, so that we can get a better gauge of where and how users are interacting with the system.

**Objective 5 Scalability:** The fifth objective focuses on the ability for the system to scale up or down as needed to incorporate new curriculum or remove curriculum. The system also needs to stay up to date with DMV standards, which may change over time.

* **Goal**: Flexible management of curriculum.
  + The main priority of this goal is to ensure that adding and removing curriculum is a straightforward process.
    - **Measurable**: Evaluate the ease of module addition and removal. This can be done by observing the time that it takes for modules to be uploaded and removed. A well-made system should take no longer than a few minutes to add, and seconds to remove.
    - **Measurable**: Evaluate the impact of content that is added on the performance of the system. Does the system get noticeably slower when new packages are added?

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

The DriverPass system will need to be run on a web-based environment so that the system is accessible from various devices and locations. Having the system be web-based will also allow users to access the system any time that they have a stable internet connection.

Fast performance is necessary to ensure that the users have a responsive user experience. Quick load times between the pages and minimal latency when users need data retrieved from the website. Speed of response times will be especially important when students wish to make reservations and when they try to access learning resources.

Regular updates will be required to ensure that the system stays up to date and in-line with current DMV standards and regulations. The system should be updated at least quarterly (every 4 months), so that the system can constantly be adapted to changing driving requirements or updated driver testing procedures. This will ensure that students receive the most up-to-date information.

#### Platform Constraints

The system should not be built specifically for one system because it will need to be accessed from a variety of different platforms. The system needs to be able to run on windows, apple, and mobile operating systems. By making the system independent of a specific platform, it allows for more users to access the system regardless of what device they use.

A database will be required so that many of the system features can be implemented successfully, including:

* User account information and management
* Reservations
* Data storage for curriculum

Using a structured database, the DriverPass system will be able to retrieve data efficiently and maintain the integrity of the data being stored. Using database querying techniques, the system will be able to store and retrieve data efficiently, which will make for a smoother user experience. We can also incorporate access controls to ensure that data is kept secure. User authentication and data encryption are possible ways of maintaining data privacy, all of which can be accomplished with a database.

#### Accuracy and Precision

We can distinguish between different users by having unique user IDs that will be assigned to each user account. When each account is created, the unique user ID will be generated so that no two users have the exact same ID number. Each user ID will be generated sequentially, so that no two user IDs will be identical. Student ID numbers can be appended with a -1 on the end so indicate that they are student accounts. While administrator accounts can be appended with a -0 on the end to indicate that they are administrator accounts. Each user ID will be hidden from the users and will be associated with the unique username that is selected during account creation. Case sensitivity during login will be necessary to prevent unauthorized access to certain accounts, this is because there can be variation in case lettering when users are creating their usernames.

Error notification to admins should happen immediately, because this is essential to identifying and solving issues. By informing the system administrator immediately, problems within the system can be addressed quickly. This will help to reduce system downtime and help to mitigate security risks.

#### Adaptability

The system will be designed to allow non-developer administrators to manage user accounts without needing to alter the code. This can be accomplished through a user-friendly administrator’s dashboard that will allow administrators to add, remove, or update user accounts. This will include adding or removing certain system privileges from users when employees are hired or leave the company.

The system will be intentionally designed to adapt to platform updates to ensure that it is always compatible with all platforms. This can be accomplished using system version control and incremental version updates. When a platform is updated the modules within the system can be updated or removed/changed if necessary. This will make sure that the risk of incompatibility is minimal, and that system performance will not be affected.

IT administrators are granted unique system privileges such as:

* System configuration
* Module updating/management
* User account management

This type of system access will allow administrators to perform routine maintenance, reset passwords, manage user account privileges, and adjust modules.

#### Security

To log in, users must provide their username and password. Each username will need to be unique since they are associated with the account’s unique user ID, which is a string of numbers with either a -1 or -0 appended to it. The suffix will indicate to the system if the user is a student or an administrator. The system owner has special privileges that allow them to alter and manage other administrator accounts. The system owner’s user ID will be the first user ID and will therefore have the smallest number to make it easily identifiable to the system (ex. user ID# 000001 = system owner). Usernames and passwords will also be case sensitive so that they can be properly linked to the correct user ID. Passwords will need to meet a certain complexity criterion in order to be accepted and assigned to the user ID. This will include being a certain length and including a special character. When creating a new account, both students and administrators will need to set up a series of security questions that will allow them to initiate the password recovery process.

To ensure that there is a secure connection between the client and the server during data exchange we can implement data encryption protocols. This will allow for data to be transferred between the client and the server securely and privately. We can incorporate HTTPS which stands for hypertext transfer protocol secure which is an industry standard for transferring private data over the internet securely. It is used by almost all companies that make use of private data transfer.

To deter brute force hacking attempts, the system will lock out accounts that enter the wrong password too many times consecutively. When an account is temporarily locked down, a notification will be sent to the system owner as well as the owner of the account that is trying to be accessed. The administrator will receive the username and user ID of the account that is trying to be accessed. The student will receive an email informing them that their account may be under attack.

If a user forgets their password, they will need to provide the correct answers to their security questions that they created during account creation. Once the user has successfully identified themselves using these security questions, they will be allowed to begin the password reset process.

### Functional Requirements

**Password** **Authentication**

* The system shall prompt users to enter their username and password before logging in.
* The system shall check if the provided username and password are case sensitive.
* The system shall require the user’s chosen password to be of a certain level of complexity, like minimum length and special characters during account creation or password change.
* The system shall ask users to set up special security questions that will help identify them in case of password reset.
* The system shall allow users to reset their password if they answer their security questions correctly.
* The system shall lock user accounts for a certain amount of time after a specified number of failed login attempts.

**Account Creation/Management:**

* The system shall allow users to create new accounts provided they have a unique email address and username.
* The system shall assign a unique user ID for each new account.
* The system shall increment the user ID number based on the last user ID number that was created to ensure that all user IDs are unique.
* The system shall append a -1 or a -0 to the end of each user ID to differentiate between student and administrator accounts.
* The system shall differentiate between student and administrator accounts and assign appropriate roles and privileges to each.
* The system shall give administrators the power to add, remove, or manage user accounts without requiring alterations to the code.
* The system shall allow administrators to reset user passwords and assign/revoke system privileges.
* The system shall keep track of how long users are spending on account creation to determine if the process needs to be refined.

**Curriculum Engagement/Management:**

* The system shall provide an online curriculum for new drivers that includes training videos, text resources, and practice tests.
  + The curriculum shall be structured into modules, with each module being a bin that contains materials relevant to that area of driving.
* The system shall allow administrators to update, add, and remove module bins so that the curriculum is always up to date with DMV standards.
* The system shall track student progress across modules and allow students to check their progress towards their goals.
* The system shall allow students to check scores that they got on previous tests and highlight the areas that a student got incorrect, allowing the student to focus on the areas that need the most improvement.
* The system shall allow students to visit previously completed modules to study areas that they may need further clarification on.

**Scheduling Reservations:**

* The system shall allow users to schedule road tests by specifying a date and time.
* The system shall display date and time slots that are available, and grey out date and time slots that are unavailable.
* The system shall maintain a record of appointments set up by students for administrative review.
* The system shall be easy to navigate, and scheduling should take no longer than a few minutes.
* The system shall track how long users spend scheduling appointments to determine if the system is user-friendly enough or if the system needs to be refined.

**Platform Compatibility:**

* The system shall be designed to work on multiple platforms, ensuring a seamless user experience across all available platforms.
* The system shall be designed to adapt to platform updates, to ensure that the system is always compatible with evolving web-based technologies and security standards.
* The system shall allow for individual updating or removal of modules when a platform update occurs.

**System and Account Security:**

* The system shall encrypt data transfer between the client and the server using industry standard encryption protocols such as HTTPS.
* The system shall be designed with ways to detect brute force hacking attempts, including account lockout after a certain amount of consecutive failed attempts.
* The system shall inform the administrator immediately when an account is temporarily locked, and another email notification will be automatically sent to the user informing them that their account has been locked because of multiple failed password attempts.
* The system shall provide a password recovery process in case of a forgotten password.
* The system shall require users to answer their secret questions before initiating the password reset process.
* The system shall keep tabs on all user login activity.

### User Interface

The interface for the DriverPass system will depend on what type of account is logged on. Since the two different account types (student and administrator) have different roles and privileges within the system the user interfaces need to be designed differently.

**Student UI:** The main need of the student side UI is for learning from modules, taking practice tests, scheduling reservations, and checking progress. Students will be accessing their accounts from both computers and mobile devices so the system needs to be designed to accommodate all screen sizes. I believe that the best way for the student interface to be designed is as follows:

* Learning Modules
  + Module 1
    - Module 1 Resources
    - Module 1 Quiz
  + Module 2
  + Module 3…
  + etc...
* Practices Tests
  + Module 1
  + Module 2
  + Module 3…
* Schedule a Road Test
  + Schedule new road test
  + View your scheduled road tests
  + View previous road tests
* Check your Progress
* Edit your Profile
  + Update Email Address
  + Update Phone Number
  + Change Password
  + Change Payment Method

Having each of the different functionalities stacked up in this way will allow for consistent user experience across all platforms and makes things easy to find. Each of the black dots represents a different tab that will be at the center of the user’s screen. Learning modules will contain all the pertinent information that is required for each aspect of learning how to drive. Each learning module will be concluded with a short quiz to ensure that students are retaining information from the module. Below that, students can take practices tests that will correspond to the modules that the information is found in. They can also schedule road tests, view upcoming road tests that they have scheduled, and view the results of their previous road tests. Student accounts can also update their user profiles by changing their associated email address, phone number, and password.

**Administrator UI:** The main need for the administrators UI is to monitor user account activity, user account management, checking student progress, and seeing all reservations. The administrator UI can be a bit less compact than the student UI because administrators will be accessing the system through their work computers, however the system still needs to be accessible and legible on mobile devices. I believe the best way to format the Administrator UI is like this:

* Manage User Accounts:
  + Student Accounts
    - Reset Password
    - Check Student Progress
    - View scheduled appointments
  + Administrator Accounts: **(System Owner Only)**
    - Reset Password
    - Assign system privileges
    - Account Deactivation
      * Suspend or deactivate new or existing administrative accounts
    - User Communication
      * Send notifications or messages to users.
* Manage Learning Modules: **(System Owner Only)**
  + Add Module
  + Remove Module
  + Alter Existing Modules
* System Reports:
  + Logging
    - Record user activities
    - Monitor how long users are spending on certain system functions
  + Generate reports for user account activity
* Reservations and Scheduling
  + View scheduled driving tests
  + Remove scheduled driving test
  + Add driving test

This administrator UI layout allows the administrators to manage user accounts for both student and other administrator accounts. It also allows the administrator to add, remove, or update learning modules if necessary. The system reports tab will record user login activity, as well as how long users are spending on account creation and appointment scheduling to determine if the systems need to be revised. It will allow for the administrator to print these logs as well. The administrator will also be able to view all upcoming scheduled road tests and can move, remove, or add driving tests if necessary. The UI will be compact with nested tabs similar to the student UI, to accommodate for smaller mobile screens.

### Assumptions

These things were not specifically addressed in the system design and are being assumed by the system designer:

* **New User Coaching:** It is assumed that the user will be able to figure out and navigate the system because of the simple UI layout.
  + There is no new user onboarding that walks the new user through each part of the system.
* **Stable Internet Connection:** The system will be inaccessible if the user is not connected to the internet.
* **Accessibility:** was not addressed to users that may have disabilities.
* **Engagement with user feedback tools:** The system being updated and improved upon is reliant heavily on users interacting with the feedback tools that are available to them.
* **Data Accuracy:** It is being assumed that users are providing accurate information about themselves when creating their accounts.

### 

### Limitations

When it comes to system design, here are some limitations that need to be considered. Even though the UI is designed to be easy to navigate, there will still be somewhat of a learning curve for people who are not tech savvy. Being able to successfully navigate the system will play a major part in having good user experiences. The system also requires the users to maintain a stable internet connection the entire time they are logged on, if at any time during a session the user loses internet access, they will be automatically logged out to prevent data breaches or leaks. This will of course limit the number of people that can use the system, however it is necessary for a web-based application.

Budget, resources, and time limitations may not allow for some of the more advanced system features to be implemented on time. Budget constraints may not allow for as much extensive user testing as may be necessary to work out any bugs or to streamline any systems that may need to be simplified. Financial restrictions may limit the development, system testing, and refinement which will affect the overall end product. Budget will also limit the amount that the system can be scaled up in the future. It is not specified whether the system will be hosted locally or on cloud services, which will greatly affect how much money can later be spent on system development. Locally hosting the system will require a higher up front cost, while the cloud services will allow DriverPass to purchase storage as they need it.

Time limitations may also limit the amount of system testing that can be done, especially load testing. To ensure that the system can handle many users logging in at once, the system needs to be load tested. Which will also need to be factored in when calculating cost and time requirements.

### Gantt Chart

