# CS 255 Business Requirements Document Template

## 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 project is to create a system that aids DriverPass (the client) in offering its services in the form of packages to customers. DriverPass identified an opportunity to provide driving test preparation packages for individuals seeking to attain licensure through the DMV. They will offer instructional information in the form of classes and online practice tests as well as hands on driving lessons if needed.

### 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 DriverPass system must be able to store customer reservation data, modify that data, and delete the data as needed. The customer and secretary must be able to make, modify, and delete online reservations as the users of the system.
* Each reservation in the system must have some identifier that shows who created the reservation, modified it, and/or canceled it. This must all be logged in an activity report that is stored for later access and auditing.
* The reservations are for driving lessons that are two hours long each and must contain driver, time, and car fields. When initially making a reservation the system will need to populate fields for first name, last name, address, phone number, state, credit card number, expiration date, and security code. Users will provide this data in an input form available online to the secretary and the customer/user.
* The client wants to initially have 3 packages available and the ability to disable a package if they need to stop registration for any reason.
* The system must be connected to the DMV to receive updates on testing requirements or other relevant information.
* The system must be a web application that is deployed to the cloud so that the team can focus on the business and leave backup and security to the cloud provider.
* The interface should show test progress (tests taken), time taken, score, and status. It should also show user information pulled from the reservation fields such as name, address, phone number, email, and other relevant information. Driver notes must show information on lesson time, start time (hour), end time (hour), and comments from the driver.
* The system must have a contact page and a page for the client team to contact students if needed.
* In total the system will be comprised of a database to store reservations and user information, a web page with a form to submit reservation details and user information, a user information page that will display user information such as test progress and drivers notes, a location to store the log files in the file system, and a main application that handles routing to the different pages as well as CRUD operations.

### 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?*

* When completed the system should be a web based application deployed in the cloud in such a way that the technical aspects such as redundancy and security are abstracted away from the business.
* Users (customers and secretary/other delegated staff) should be able to navigate to the DriverPass web application and create reservations, update reservations, and delete reservations.
* All created reservations must be logged in a separate log file that shows who created, modified, or accessed it last.
* Users should be able to view their upcoming reservations as well as their personal information and related driver notes on a single web page.
* If a reservation is no longer available for any reason, staff must be able to immediately disable that reservation.
* There must be a section of the application dedicated to displaying new information from the DMV regarding testing. This can be a whole page or potentially a bulletin type object on the main page.
* Routing and CRUD must be evident in the main application and input fields must be validated and sanitized.

## 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?*

* The DriverPass web application must run in desktop and mobile environments within all modern web browsers
* The application will need to run in all modern browsers as users will use a variety of web browsers to access the application.
* The system must be fast and page load times should not be more than a couple seconds, given user’s preferences for fast load times
* DMV updates must be fast and not impact the operation of the platform
* Updates should occur at least once a month but a faster update schedule should be considered
* Updates should occur when web traffic is slow and ideally never during peak hours
* Security updates for known vulnerabilities must be applied as soon as technically possible
* All infrastructure and operating system updates will be handled by the chosen cloud provider, the team will be responsible for feature updates, application security updates, UI/UX updates, and bug fixes

#### 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 application must be web-based and accessible to all modern operating platforms including Windows, macOS, Linux, and mobile operating systems such as androidOS and iOS.
* The application must be compatible with all modern browsers such as Microsoft Edge, Google Chrome, Mozilla Firefox, and Safari.
* The backend will require a relational database that will be capable of storing the structured customer information, reservation information, driver information and availability, and test progress.
* The platform will need to be deployed to the cloud. Amazon Web Service (AWS), Microsoft Azure, or Google Cloud are all viable cloud providers.

#### 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, email addresses and/or usernames will be used to identify the user.
* Role-based access control (RBAC) should be implemented to ensure that each distinct user type (customers, staff, and administrators) only has access to information and operations within the system that their role specifies. This will be critical for security and compliance.
* Input for usernames and email addresses will not be case sensitive. Instead they will be converted to lowercase upon form submission.
* Passwords must be case-sensitive. This is for security purposes, as upper and lowercase passwords increase the number of viable permutations, increasing the overall security and resistance to brute force attacks.
* All sensitive information will use input validation to ensure that the information being entered is of the correct type and form. Sensitive information includes contact and payment information.
* The system should inform the admin immediately as a problem arises. If a user submits invalid credentials or if a transaction is not successful, then that must be logged immediately. Notifications must be made available to the admin in real-time and logged for later analysis and review. Activities such as updating or deleting reservations, changing user roles, log in events, log off events, failed login attempts must all be kept in an activity log.
* The system should also immediately inform the user that their form submission was not valid. It should not reveal what information was not valid but should prompt them to resubmit their credentials.
* Periodic reports should be generated by the system for predefined time intervals that can be reviewed by system administrators.

#### 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 must allow for the addition, removal, and modification of users without changing code. An interface available to admins must be available to perform these operations. This interface should be where Role-based access control is handled.
* In future iterations of the service Driver Pass would also like to update, add, remove packages as well.
* The system should adapt to platform updates. Being a web-based application it must conform to all current browser standards. Using modern web frameworks such as React or Angular will help ensure the platform is performant and up to date with current web standards.
* As this application will be deployed to the cloud all of the underlying operating system and infrastructure updates will be handled by the cloud provider.
* The team must have a regular schedule for updating the application when changes must be made that require manual changes to be made. This can be weekly, bi-weekly, or monthly.
* Users designated as IT staff must have full permissions within the system to make changes to user accounts, perform password resets, disable accounts for security reasons, take sections of the application or the whole application offline due to security breaches, and any other critical functions that must be carried out.
* All logs must be accessible to IT staff.
* All system updates will be the responsibility of the IT staff.

#### 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?*

* In order to log in a user will require a valid username/email and password.
* Passwords will have some predefined schema that they must follow such as a mixture of upper and lower case letters, numbers, and special characters. Minimum password lengths should also be enforced as the length of a password is a critical component of its security. The more permutations the more difficult a password is to attack.
* The system should enforce multi-factor authentication for IT staff. Regular employees should also be required to use multi-factor authentication (MFA) but as to not hinder operation maybe it is only required after a set time or after so many logins. Users should be strongly encouraged but not forced to use MFA.
* All communications between the client and server should be encrypted using SSL for security and privacy.
* For highly sensitive data such as user information, credentials, and payment information end-to-end encryption should be used. It must be encrypted on the client side before being sent over the internet to the server where it will then be decrypted. This ensures that even if the data is intercepted it does not reveal the sensitive information within the packet.
* All communication must be over HTTPS.
* If there is a “brute force” hacking attempt the system must take immediate action. It is common to lockout the account after a specified number of failed login attempts that will reset after some predefined time period. When this happens the lockout must be logged and the admin must be notified. If this is a recurring problem for an account or set of accounts that happens regularly, IP banning could be a logical next step.
* If the user forgets their password there should be an option on the account login form and in a help section where they can request a new password. This should send a change of password email to their registered email address.

### 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.”*

* User Authentication and Account Management:
  + The system shall validate user credentials at login.
  + The system shall allow users to reset their password upon request.
  + The system shall allow administrators to manage user accounts.
  + The system shall allow administrators to modify, add, remove user roles and permissions.
  + The system shall allow administrators to disable lesson packages.
* Scheduling and Reservation management:
  + The system shall allow customers to create, modify, and cancel reservations.
  + The system shall enable customers to select their preferred lesson package if available and schedule lesson times.
  + The system shall keep track of which drivers, car, and timeslot are assigned to each distinct reservation.
  + The system shall allow secretaries to also create, modify, and cancel reservations on behalf of the customer.
* Test and Progress Tracking:
  + The system shall provide a dashboard that shows user test progress and relevant details such as test status, name, time, and score.
* Activity logging and reporting:
  + The system shall track all changes to reservations and their creation and log them.
  + The system shall generate reports on all user activity such as lessons bookings, cancellations, and test results.
  + The system shall track and log all user account creations, modifications, and deletions.
  + The system shall log all login attempts, times, and usernames for later review.
  + The system shall store all logs for later review and provide periodic reports for review.
* Integration:
  + The system shall integrate with the DMV system to automatically receive updates on current policies and procedures as well as curriculum updates.
* Notifications:
  + The system shall send notifications to all administrators when a system issue, security breach, sequence of failed login attempts, password change or any other relevant change takes place in the system.
* Feedback:
  + The system shall maintain records of all driver comments submitted after the completion of each lesson and maintain this information review.

### 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 must be a responsive user friendly dashboard that allows each type of user to interact with the system in accordance with their role.
* Customer Interface:
  + Must allow for account creation
  + Must allow users to log in and reset their passwords
  + Must allow users to book, modify, and cancel driving lessons
  + Must allow users to purchase lesson packages
  + Must provide the users scheduled lessons
  + Must provide users exam progress
  + Must provide relevant contact information for the user
  + Must show notes for their driver
  + Must show their photo and their drivers photo
  + Must show any special needs they may have
  + Users will primarily interact with this interface through a web browser either on desktop or mobile
* Secretary:
  + Schedule, modify, and cancel lessons on behalf of the user
  + Access user contact information, bookings, and driver information
  + The secretary will interface with the application through a web browser on a company computer on-premise. Their main function will be to help customers manage their lessons, bookings, and scheduling.
* IT Officer:
  + Must have full administrative control for the purpose of resetting user accounts, blocking access for terminated users, monitoring system security, responding to incidents, etc.
  + Must be able to access and view all system logs through this interface.
  + Must be able to manage various system settings such as enabling and disabling packages
  + They must be able to create, modify, or destroy any account, lesson, booking, or package in the system in case of internal errors or incidents.
  + Must be able to generate activity reports and view them from this dashboard
  + The IT officer will interact with the system through the web browser through a desktop or laptop and have full administrative control over the system
* Admin
  + Must be able to view overall system information such as bookings, user progress, financial reports, users in the system, drivers, etc.
  + Must be able to enable and disable packages
  + Must be able to generate activity reports and view them from this dashboard
  + The admin will interact with the system via a browser on a laptop or desktop and have access to reporting tools and tools for managing user accounts, packages, lessons, and other system details.

### 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?*

* In my design above I did not specifically address security policies
* I did not address the cost to implement this system
* I did not address the cloud provider and the cost benefit analysis of choosing one over another
* I did not address the technical skills of the staff or the users of the system or any training that will be required
* I did not address the legal implications of what happens in the event of dispute or data breach
* I did not address the future scalability concerns for the system as more users are onboarded
* I did not address how to implement new features
* I did not address how the team will integrate with the API and the DMV
* I did not address how data will be stored (schema, database type, etc)

### 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 current system does not allow the team to add new packages. Existing packages can only be disabled
* For all future updates and feature additions, a developer and system analyst at minimum will be needed to build out new features
* The system will be limited as it brings in more users and scaling solutions must be implemented. This is possible in the cloud but will still require planning
* The integration with the DMVs API will likely need to be updated periodically if endpoints change or new restrictions on access are put in place
* The team only has a handful of months to complete the project, any delays will set them back and may push back the Sign-off meeting.
* It is unlikely that the team has an unlimited budget and will need to make tradeoffs as they choose cloud providers and provisioned resources. They will need to consider their operational expenses, scalability needs, and profit as they grow as a business.
* Since they are using a cloud provider, it will be important to consider how they will handle the storage and retrieval of sensitive data with respect to regulatory laws and compliance guidelines.

### 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.*

