# CS 255 Business Requirements Document

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 DriverPass project is to create a comprehensive system that provides better driver training to individuals preparing for their driving tests at the local Department of Motor Vehicles (DMV). The client, Liam, who is the owner of DriverPass, envisions a platform that offers online classes, practice tests, and on-the-road training for driving test candidates.

### 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 the system to provide comprehensive driver training services, addressing the issues of inadequate preparation for driving tests at the local Department of Motor Vehicles (DMV).
* The system's main goals are as follows: Online Driver Training, On-the-Road Training, Reservation Management, User Management, Tracking and Reporting, Flexible Packages, Data Accessibility, Security and Access Control, Interface Design, and Integration with DMV Updates.
* Components needed for the system include User Interface, User Management System, Reservation Management System, Activity Tracking and Reporting, Content Management System, Security and Access Control, Integration with DMV, and Database.
* The project schedule involves various stages including requirement collection, use case and activity diagram creation, UI design, database design, business logic implementation, system delivery, and sign-off.

### 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 system that needs to be completed for DriverPass should have the following capabilities:

1. Online and Offline Access to Data: Users, especially Liam, should be able to access data from anywhere, both online and offline.
2. Security and User Roles: Different employees should have different rights and roles within the system. Ian, the IT officer, requires full access to manage accounts and access rights.
3. Tracking and Activity Reporting: The system should track user activities, such as reservations, modifications, and cancellations.
4. Reservation System: Customers should be able to make reservations for driving lessons. Reservations include selecting a package, specifying the day, time, and driver for each lesson.
5. User Management and Interaction: Different types of users include Liam (the owner), Ian (IT officer), secretary, and customers.
6. Connection to DMV and Compliance: The system needs to be updated with current DMV rules, policies, and sample questions.
7. Cloud-Based System: The system should run on the web, preferably on the cloud. Backup and security measures should be taken care of to minimize technical problems for the business.
8. User Interface Design: The system should have a user-friendly interface that aligns with Liam's sketch and specifications.
9. Future Flexibility: The system should be flexible enough to add or remove packages in the future. Features for adding/removing modules might require developer intervention.

## 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 system for DriverPass needs to run in a web-based environment to provide online and offline access to data enabling users to access information from anywhere. The system's speed and responsiveness are pivotal. It should be designed to provide a seamless and efficient user experience, ensuring that users can interact with the system without delays or performance issues. Regular system updates are necessary to align with the objective of maintaining compliance with DMV regulations. Given the requirement to stay up to date with DMV rules, policies, and sample questions, the system should be updated periodically to accommodate any changes or additions.

#### 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 choice of platforms for the system's operation and the back-end tools required to support the application should align with the stated objectives and goals for DriverPass. The system should be designed to run on platforms like Windows, Unix, and potentially cloud environments. The back end should include a database for user management, reservations, activity tracking, and integration with external services like the DMV. Cross-platform compatibility, security, and scalability are key factors to consider when selecting platforms and back-end tools. Cloud-Based System: Cloud-based databases and storage solutions, along with security measures, are necessary for a reliable cloud-based system.

#### 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 and address the case-sensitivity of input for the DriverPass system:
* User Identification and Differentiation Objective: The system aims to differentiate between various users based on their roles and access rights. User Authentication: The system utilizes secure authentication mechanisms to ensure that only authorized users can access their designated functionalities. This authentication process helps distinguish between users.
* Case-Sensitivity of Input Objective: The system's approach to handling case sensitivity in input is outlined. Uniform Input Processing: The system processes input uniformly, irrespective of case. This means that input is treated as case-insensitive, ensuring a consistent user experience regardless of capitalization.
* Problem Notification to Admin Objective: The system's approach to alerting the admin about issues is defined. Admin Alert Thresholds: When a problem surpasses predefined thresholds, such as critical errors affecting user interactions or security breaches, the system triggers notifications to the admin or designated personnel. The system ensures that the admin is informed in real-time when significant problems arise, enabling prompt action and issue resolution.

#### 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?*

* It is possible to add new users to the system without changing the underlying code. An administrative user interface can allow authorized personnel, such as the secretary or an admin, to create new user accounts and specify their roles and permissions. Similarly, authorized users can deactivate or delete user accounts through administrative interfaces without altering the code.

Ian, the IT officer at DriverPass, needs various levels of access and responsibilities, as mentioned in the interview:

1) Full Access to Accounts

2) System Maintenance

3) Security Oversight

4) Integration with DMV

5) Data Management

#### 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 user is required to provide their information, including first name, last name, address, phone number, state, and credit card details (number, expiration date, and security code). Additionally, for online access, they would set up a username and password. To secure the connection and data exchange between the client and the server, implementing encryption protocols such as HTTPS (SSL/TLS) is crucial. This ensures that data transmitted over the internet is encrypted and secure from eavesdropping. To mitigate brute force hacking attempts, the system can implement account lockout policies. After a certain number of unsuccessful login attempts, an account can be temporarily locked, preventing further login attempts. For users who forget their passwords, the system should provide a "Forgot Password" or "Reset Password" functionality. These considerations are crucial for building a robust and secure system.

### 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.”*

1. Data Access and Availability:

The system shall allow users, including Liam and authorized employees, to access data from anywhere, both online and offline.

Data modification and updates shall only be allowed when the user is online to prevent data redundancy.

1. Security and User Roles:

The system shall implement role-based access control, with different user roles, including administrators (e.g., Ian), secretaries, and students.

Administrators shall have full access to user accounts, including the ability to reset passwords and block user access.

1. User Activity Tracking:

The system shall record and track user activities, such as making reservations, cancellations, and modifications.

Activity logs shall include information about who made the changes and when they were made.

1. Reservation Management:

The system shall allow customers to make reservations for driving lessons.

Reservations can be for various packages, including Package One, Package Two, and Package Three.

Each driving session shall be two hours long, and customers shall specify their preferred day and time for the lesson.

The system shall match users with specific drivers, times, and cars for their scheduled lessons.

1. User Registration:

The system shall enable user registration, either through phone calls with a secretary or online self-registration.

User registration shall collect information such as first name, last name, address, phone number, state, and credit card details.

The system shall validate credit card information and store it securely.

### 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.)?*

UI Needs:

1. User-Friendly Web Interface: The system should have a user-friendly web interface accessible from both computers and mobile devices.
2. Data Access and Reporting: Users should be able to access data online and offline. The interface should allow users to download reports and data for offline work, such as using Excel.
3. Security Features: The interface should incorporate security features, including user authentication and authorization. It should support role-based access control to restrict access to sensitive functionalities.
4. Tracking and Reporting: Users, particularly administrators, should be able to track user activities and generate activity reports to identify changes made to records in the system.
5. Reservation Management: Users should be able to view, schedule, modify, and cancel driving lesson reservations online.

Different User Roles:

1. Liam (Administrator): As the owner of DriverPass, Liam has full access to the system. He can access data, review reports, track activities, customize packages, and monitor system compliance. Liam may also use the interface to contact users.
2. Ian (IT Officer): Ian is responsible for system maintenance and modifications. He has full access to user accounts and can reset passwords or block access as needed. Ian may also use the interface for administrative tasks.
3. Secretary: The secretary answers phone calls, schedules appointments, and interacts with customers. They use the interface to manage customer appointments, including scheduling, modifying, or canceling them.
4. Students/Customers: Students use the interface to register, schedule appointments, access online classes, practice tests, and view their progress. They may also contact the company through the interface and reset their passwords if needed.

### 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?*

1. Usability Testing: It's essential to conduct usability testing to ensure that the web interface is genuinely user-friendly. Assumptions about what is user-friendly can vary, so user feedback should be actively sought and incorporated into the design.
2. Mobile Accessibility: While I mention that the interface should be accessible from mobile devices, to consider whether it should be a mobile-responsive web application or if there should be a dedicated mobile app for better mobile user experience.
3. Data Backup and Recovery: There's no mention of data backup and recovery procedures. It's important to have robust data backup mechanisms in place to prevent data loss in case of system failures or data corruption.
4. Offline Data Access: I mentioned users should be able to download reports and data for offline work, but details about how this offline mode will work and how data synchronization will be handled when users go back online are not specified.
5. Data Privacy and Compliance: Ensure that the system complies with data privacy regulations, such as GDPR or HIPAA, if applicable, and outline how user data will be handled and protected.

### 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?*

Offline Data Access: Providing offline data access can be challenging, as it requires synchronization mechanisms and potential data conflicts when users modify data offline and then go online. Data Redundancy: Restricting data modification only to online users may prevent redundancy, but it could lead to inconvenience for users who need to make changes when offline. Administrative Privileges: Giving administrators full access to user accounts should be handled with caution, as it poses security risks if not properly controlled and monitored. Matching Algorithm: Developing an effective algorithm to match users with specific drivers, times, and cars based on user preferences can be complex, especially if there are many users and limited resources. Scalability: Handling many reservations and ensuring smooth scheduling can be resource intensive. Data Privacy and Security: Storing sensitive user information like credit card details requires robust security measures to protect against data breaches and compliance with data protection regulations. Validation: Validating credit card information can be challenging, and false positives/negatives can lead to user frustration or security vulnerabilities.

Technological limitations could also be a factor, depending on the chosen technology stack and the capabilities of the development team.

It's essential to carefully plan and prioritize features, allocate resources appropriately, and continuously monitor and improve the system to address these limitations effectively.

### 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.*

*A screenshot of a chart

Description automatically generated*