# 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?*

* Purpose of the project: To design a system that provides online practice exams and on-the-road training for students preparing for driving tests.
* Client: Driver Pass
* Client’s Needs\*\*:
* Access to online practice exams.
* On-the-road training programs.
* A system that supports tiered client-level access to different driving packages.
* Deficiency Addressed\*\*: The proposed system aims to address the high failure rate (over 65%) of students in driving license exams by offering comprehensive training tools beyond just studying previous tests.
* Project Purpose: The main goal is to enhance the preparation of students for driving tests through a structured system that combines theoretical knowledge (via online exams) and practical skills (through on-the-road training).
* Client Identification: Driver Pass is the client seeking to improve student outcomes in driving tests.
* System Requirements:
* Online Practice Exams: This feature will allow students to take simulated tests that mimic the actual driving exam format.
* On-The-Road Training: Practical training sessions that provide real-world driving experience.
* Tiered Client-level Access: Different levels of access for users (students, instructors, administrators) to ensure that each user can access the appropriate resources and functionalities based on their role.
* The system is designed to combat the ineffective study methods currently used by students, which primarily involve rote memorization of past test, leading to a high failure rate. By providing a more interactive and comprehensive training approach, Driver Pass aims to improve student preparedness and success rates in driving tests.
* This framework sets the stage for further development of the system’s specifications and functionalities.

### System Background

*What does Driver Pass want the system to do? What is the problem they want to fix? What are the different components needed for this system?*

* Driver Pass wants to create an online platform for practice exams and on-the-road training to improve student success rates in driving tests. The problem they aim to fix is the high failure rate (over 65%) of students who only study previous test. The system will require components such as a user interface, a database for storing exam module, training module, and reporting tools for tracking progress.
* Driver Pass’s Goals:
* Online Practice Exams: Provide a platform where students can take practice tests that simulate the actual driving exam.
* On-the-Road-Training: Offer resources and scheduling for practical driving lessons with qualified instructors.
* Identified Problem:
* High Failure Rate: More than 65% of students fail their driving tests due to inadequate preparation, primarily relying on previous tests without comprehensive training.
* System Components Needed:
* User Interface e: A user-friendly interface for students to access practice exams and training resources.
* Database: A robust database to store exam questions, user profiles, and performance metrics.
* Online Exam Module: A system to administer practice exams, including timed tests and instant feedback.
* Training Module: A scheduling and management system for on-the-road training sessions.

### 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?*

* Online Practice Exams: Allow students to take simulated driving tests that mimic the actual exam format.
* On-the-road Training: Provide resources or scheduling for practical driving lessons with qualified instructors.
* Progress Tracking: Enable students to track their performance on practice exams and identify areas needing improvement.
* Goal: Increase the pass of students taking the driving test.
* Objectives:
* Development and launch an online platform for practice exams within 6 months.
* Create s database of qualified driving instructors and integrate scheduling features within 4 months.
* Implement a feedback system that allows students to receive performance analytics after each practice exam within 2 months.
* Design user-friendly interfaces for both practice exams and instructor scheduling.
* Conduct user testing with a sample group of students to refine the system before launch.
* Establish metrics for success, such as tracking the percentage of students who pass their driving tests after using the system.

## Requirements

### Nonfunctional Requirements

*In this section, you will detail the different nonfunctional requirements for the Driver Pass 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 designed for DriverPass should ideally run on multiple platforms, including Windows, macOS, and mobile operating systems (iOS and Android), to ensure accessibility for a wide range of users. Given the nature of the application, it should be optimized for performance, aiming for a response time of under 2 seconds for user interactions to provide a seamless experience during practice exams and training sessions. Regular updates are crucial for maintaining the system’s relevance and effectiveness; therefore, the system should be updated at least quarterly to incorporate user feedback, improve features, and ensure compliance with any changes in driving regulations or testing standards. Additionally. A robust backend infrastructure should be established to support real-time data processing and analytics, which will help in tailoring the training programs to meet the evolving needs of the students.
* The system shall ensure data security by implementing encryption for user data.
* The system shall maintain high availability,ensuring users can access it 24/7.
* The system shall provide a user-friendly interface to enhance user experience.
* The system shall be scalable to accommodate an increasing number of users.
* The system shall comply with relevant data protection regulations.

#### 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 for DriverPass should ideally run on a cross-platform environment, with a focus on web-based accessibility to cater to a diverse user base. Utilizing a cloud-base solution would allow the application to be accessible on various operating systems, including Windows, macOS, and Linux, ensuring that students can practice from any device with internet access. For the back end, a robust relational database management system (RDBMS) such as PostgreSQL or MySQL is recommended to store user data, practice exam results, and training progress securely. Additionally, implementing a server-side framework like Node.js or Django can facilitate the development of RESTful APIs, enabling seamless communication between the front end and the database. This architecture not only supports scalability as the user base grows but also enhances data integrity and security user information. Overall, a combination of a cloud-based platform. A reliable database will provide a solid foundation for Driver Pass’s online training 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 effectively distinguish between different users in the DriverPass system, we will implement a role-based access control (RBAC) mechanism that categorizes users into distinct roles such as students, Instructors, and administrators. Each role will have specific permissions and access levels tailored to their needs. For instance, students will have access to practice exams and training materials, while instructors can monitor student progress and provide feedback. The system will be case-sensitive for usernames and passwords to enhance that user credentials are unique and accurately recognized. Additionally, the system will be designed to inform the admin of any issues, such as failed login attempts, unauthorized access attempts, or system errors, through real-time alerts and logs. This proactive approach will enable the admin to address potential security threats and maintain system integrity, ultimately contributing to a more reliable and user-friendly experience for all stakeholders involved.

#### 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?*

* To ensure the system designed for DriverPass is adaptable and user-friendly, it should incorporate a robust role-based access control (RBAC) framework that allows the IT administrator to manage user accounts dynamically without requiring code changes. This can be achieved by implementing a user management interface where the IT admin can easily add. remove, or modify user roles and permissions through a graphical user interface (GUI). The system should utilize a database-driven approach to store user information and roles, allowing for seamless updates and modifications. Additionally, to accommodate platform updates, the system architecture should be designed with modular components, enabling individual modules to be updated independently without affecting the overall system functionally. This modularity ensures that as new features or updates are introduced, the existing system remains stable and operational. The IT admin will need elevated access rights, such as administrative privileges, to perform these tasks, ensuring they can manage user accounts and system configurations effectively while maintaining security protocols. This approach not only enhances the system’s adaptability but also empowers the IT team to respond swiftly to changing user needs and technological advancements.

#### 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?*

* Login Requirements: Username and password.
* Securing Connection/Data Exchange: Use HTTPS and SSL/TLS encryption.
* Brute Force Attack Response: Lock the account after a set number of failed attempts.
* Forgotten Password Procedure: Implement a password reset mechanism via email verification.
* Login Requirements: Users should provide a unique username and a strong password to log in This ensures that only authorized users can access the system.
* Securing Connection/Data: To secure the connection between the client and server implement HTTPS, which uses SSL/TLAS protocols to encrypt data during transmission. This prevents eavesdropping and man-in-the-middle attacks.
* Brute Force Attack Response: To prevent against brute force attacks, the system should lock the user account after a predetermined number of failed login attempts (E.g. 5 attempts). This can be accompanied by a temporary lockout period to deter further attempts.
* Forgotten Password Procedure: If a user forgets their password the system should provide a password reset option. This typically involves sending a verification link to the user’s registered email address, allowing them to securely create a new password. This process helps ensure that only the legitimate user can reset their password.

### 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 provide online practice exams for students.
* The system shall offer on-the-road training scheduling and management.
* The system shall track student progress and performance on practice exams.
* The system shall allow users to create and manage their profiles.
* The system shall send reminders for upcoming training sessions and exams.

### 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.)?*

Identify User Types:

* Students: Primary users who will take practice exams and schedule on-the-road training.
* Instructors: Users who will provide training and assess student performance.
* Administrators: Users who manage the system, track progress, and generate reports.

User Needs:

* Students: Access online practice exams. Schedule and manage on-the-road training sessions. Track their progress and performance analytics.
* Instructors: Create and manage practice exams. View student performance and provide feedback. Schedule training sessions and manage student assignments.

Administrators:

* Manage user accounts (students and instructors).
* Generate reports on student performance and settings and configurations.
* Oversee system settings and configurations.

Interaction Methods:

Mobile Applications:

* Students can take practice exams and schedule training on-the-go.
* Instructors can manage sessions and provide feedback from their mobile devices.

Web Brower:

* All users can access the full functionality of the system, including detailed reports and analytics.
* Administrators can perform comprehensive management tasks.

User Experience Considerations:

* The interface should be user-friendly with intuitive navigation for all user types.
* Responsive design to ensure usability across devices\*mobile and desktop).
* Accessibility features to accommodate all users, including those with disabilities.

### 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?*

*User Experience (UX Design):*

* It is assumed that the target users (students)will have varying levels of tech-savviness, The design should accommodate both novice and experienced users, ensuring an intuitive interface that is easy to navigate.
* The assumption is that students prefer interactive and engaging content, which may not have been explicitly discussed. This could include gamification elements or multimedia resources to enhance learning.

Technology Integration:

* It is assumed that DriverPass has the necessary infrastructure (hardware and software) to support an online platform. This includes reliable internet access and compatible devices for students.
* The design may assume that the IT team at DriverPass can maintain and updating the system, which could impact the choice of technology stack and system architecture.
* There may also be assumptions about the integration of the system with existing tools or platforms used by DriverPass, such as scheduling software for on-the-road training sessions.

Data Security and Privacy:

* It is assumed that there are no specific regulations or concerns regarding data privacy that need to be addressed, which may not have been covered in the interview. This is critical for handling student information and performance data.

Feedback Mechanisms:

* The design may assume that there will be a mechanism for collecting user feedback to continuously improve the system, but this was not explicitly discussed.

Scalability:

* It is assumed that the system should be scalable to accommodate future growth in user numbers and additional features, which may not have been a focus during the initial discussions.

### 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?*

* Limited Resources: The consulting company may have a small team. Which can restrict the amount of work that can be done simultaneously. This can lead to delays in system development and implementation.
* Time Constraints: There may be a tight deadline to deliver the system to DriverPass, which can limit the thoroughness of testing and refinement. Rushed development can lead to bugs and a less user-friendly interface.
* Budget Restrictions: The budget allocated for the project may be insufficient to cover all desired features, leading to compromises in functionally or quality. This can affect the overall effectiveness of the training system.
* Technology Compatibility Issues: The system must be compatible with various devices and operating systems that students might use. If the technology stack chosen is not widely supported or complicate development and maintenance.

### 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.*

[Insert cA paper with a graph on it

Description automatically generatedhart]