# CS 255 Business Requirements Document Template

Michael Cary

06/09/2024

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?*

* **Primary Objective:** The purpose of this project is to develop a comprehensive digital solution for DriverPass, a company dedicated to improving the success rate of individuals attempting driving tests. The system is designed to provide both theoretical and practical training tools online, helping users prepare effectively for their driving exams.
* **Client Profile:** DriverPass is the client, a new enterprise focused on filling a market gap in driving education. Their target is to reduce the high failure rates observed at DMV tests by offering more structured and accessible training options.
* **System Functionality:**
  + **Online Practice Exams:** The system will offer a variety of interactive, web-based practice exams that mimic the format and difficulty of actual DMV tests.
  + **Scheduling for On-the-Road Training**: Users will be able to schedule, reschedule, and cancel on-the-road training sessions directly through the platform.
  + **Progress Tracking:** The system will track user progress through both theoretical lessons and practical training, providing detailed reports and feedback.
  + **Access and Management:** The system will be accessible on multiple devices and platforms, ensuring that users can engage with their training anywhere at any time. It will also allow administrators to manage content, user roles, and view system-wide analytics to ensure the training remains effective and up to date.

This purpose-driven approach aims to create a user-friendly, scalable, and effective training tool that significantly improves the likelihood of success for individuals taking driving tests, thereby establishing DriverPass as a leader in the driving education industry.

### 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?*

* **System Functionality Requirements**: DriverPass wants the system to:
  + **Provide Online Learning and Testing:** Offer comprehensive online courses and practice exams to help students prepare for their driving tests more effectively than traditional study methods.
  + **Facilitate Scheduling for On-the-Road Training:** Enable users to book, modify, and cancel driving lessons with ease, providing flexibility in scheduling their on-the-road training.
  + **Track and Report Progress:** Implement tools that allow both students and instructors to monitor progress through various stages of learning and practice, enhancing the ability to tailor training to individual needs.
* **Problem They Want to Fix:** The primary problem DriverPass aims to address is the high failure rate among individuals taking DMV driving tests. Research indicates that many fail because their preparation is often limited to studying previous tests without practical training. DriverPass seeks to fill this gap by providing a more holistic training approach that combines theoretical knowledge with practical driving experience.
* **System Components:**
  + **Learning Management System (LMS):** A core component that facilitates the creation, management, and delivery of online educational content and practice tests.
  + **Scheduling Module:** This component allows users to schedule driving lessons based on availability of instructors and vehicles. It should integrate seamlessly with the LMS for updating lesson statuses and tracking completion.
  + **Analytics and Reporting Tools:** To assess user progress and effectiveness of the training programs, providing insights to administrators and users for continuous improvement.
  + **User Management:** A secure system to manage user roles, permissions, and data access, ensuring that different users (students, instructors, administrators) have appropriate access levels based on their needs.

This detailed system background outlines the functions the system needs to perform and the components required to address the high failure rates at DMV tests, thereby improving the overall effectiveness of driving education offered by DriverPass.

### 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?*

## Comprehensive Training Provision: The system should provide a full suite of resources for both theoretical and practical driving training. This includes interactive online courses, practice driving tests, and scheduling capabilities for in-person driving lessons.

## Enhanced User Engagement and Success Rates: Aim to reduce the failure rates for DMV tests by improving the preparation process. Measurable tasks include tracking user progress through the courses and practice tests, and recording pass rates before and after system implementation to evaluate effectiveness.

## Accessibility and User-Friendliness: Ensure that the system is easily accessible and navigable on various devices, including desktops, tablets, and smartphones. Performance metrics could include user satisfaction ratings and ease-of-use assessments through surveys or user feedback.

## Efficient Scheduling and Resource Management: Implement a robust scheduling module that allows for efficient booking, modification, and cancellation of driving lessons. Key performance indicators (KPIs) could include the reduction in scheduling conflicts, user-reported satisfaction with the scheduling process, and utilization rates of instructors and vehicles.

## Real-Time Reporting and Analytics: The system must provide real-time analytics and reports on user performance, lesson completion, and test outcomes. These should help administrators and instructors make informed decisions to improve educational content and teaching methods.

## Security and Data Integrity: Establish and maintain high standards of data security and user privacy. This includes implementing secure authentication, data encryption, and compliance with relevant data protection regulations. Measures of success here include audit results, absence of data breaches, and adherence to privacy standards.

## Scalability and Adaptability: The system should be designed to scale with the growth of the business and adapt to changes in technology or user demand. This involves designing with modular components and ensuring that the system can handle an increasing number of users and data without degradation in performance.

## These objectives and goals define what the system must achieve to be considered successful, with specific, measurable tasks that can be monitored and evaluated to ensure they are met. This approach helps to align the system design and development with the strategic needs of DriverPass.

## 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?*

#### Operating Environments:

#### Web-Based Platform: The system must be primarily web-based to ensure accessibility from any device with internet connectivity, including desktops, laptops, tablets, and smartphones. This allows for a broader reach and convenience for users who may access the system from various locations.

#### Cloud Infrastructure: Utilizing cloud services for hosting the system ensures scalability, reliability, and easier management of the backend processes.

#### System Performance Metrics:

#### Response Time: The system should respond to user inputs and load content within 2 seconds under normal conditions, ensuring a smooth and efficient user experience.

#### Uptime: The system should aim for a minimum of 99.9% uptime, excluding scheduled maintenance, to provide reliable access to its users around the clock.

#### Update Frequency:

#### Regular Updates: The system should undergo routine updates monthly to ensure it remains secure, incorporates the latest functionalities, and improves existing features based on user feedback and technological advancements.

#### Emergency Patches: In case of security vulnerabilities or critical bugs, emergency updates must be deployed as soon as possible to maintain system integrity and user trust.

#### Maintenance Schedules:

#### Scheduled Maintenance: To minimize disruption, maintenance activities that might impact system availability should be scheduled during off-peak hours and communicated to users in advance through system notifications.

#### These performance requirements are crucial to ensure the system operates efficiently, securely, and remains user-centric, providing a seamless and reliable experience for all users of the DriverPass system.

#### 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?*

* **Windows and Unix Platforms:** Ensuring compatibility with both Windows and Unix is crucial for flexibility and broad adoption in diverse IT environments. These platforms are widely used in business environments and offer robust support for enterprise applications.
* **Additional OS Support:**
  + **macOS:** Including support for macOS can be beneficial, particularly for organizations or users who prefer Apple’s ecosystem. macOS is known for its robust security features and smooth user interface, which can enhance the user experience.
  + **Linux:** While Unix provides a solid foundation, specifying support for popular Linux distributions like Ubuntu, CentOS, or Red Hat Enterprise Linux can ensure the system is accessible on free and open-source software platforms that are commonly used in servers and cloud environments.
* **Backend Tools Required:**
  + **Database Management System (DBMS):** A robust DBMS is crucial for managing user data, reservations, and training materials. The system requires a relational database like PostgreSQL or MySQL for structured data storage, ensuring transactional integrity and security. These databases will handle queries, transactions, and data management efficiently.
  + **Cloud Services:** Integration with cloud services such as AWS, Azure, or Google Cloud is recommended to provide scalable storage, computing power, and other managed services like database hosting and backups, which are essential for maintaining data availability and security.
  + **Middleware:** Application servers and middleware are needed to facilitate communication between the front-end user interfaces and the backend database, managing user sessions, and processing business logic.

These tools ensure that the backend of the DriverPass system is capable of supporting the application’s operations, security requirements, and data integrity needs efficiently and effectively.

#### 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?*

* **User Distinction:** The system will implement role-based access control (RBAC) to differentiate between user types (e.g., administrator, instructor, student). Each user role will have specific permissions and access levels defined in the database, ensuring that users can only interact with the system in ways appropriate to their roles.
* **Case Sensitivity:** Input fields that are sensitive to format, such as usernames or email addresses, will be treated as case-insensitive to ensure user accessibility and reduce errors during data entry. However, passwords will remain case-sensitive to enhance security.
* **Admin Notifications:** The system will automatically alert administrators in several scenarios:
  + Security breaches or repeated failed login attempts.
  + System errors or failures that impact service availability.
  + Any changes to user permissions or critical data modifications.

This setup ensures that the system maintains high levels of accuracy and precision in user management and data handling, while promptly informing administrators of issues that require their attention.

#### 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?*

* **User Management Flexibility:** The system will include a dynamic user management interface that allows administrators to add, remove, or modify user accounts without needing to change the underlying code. This functionality will be facilitated through an admin dashboard that provides tools for managing user roles and permissions directly.
* **System Updates and Platform Adaptability:** To ensure the system remains compatible with ongoing platform updates, it will be built on a modular architecture. This approach allows individual components of the system to be updated independently without disrupting overall functionality. Regular updates will be scheduled, and the system will be tested on new versions of operating systems and other software to ensure continuous compatibility.
* **IT Admin Access Requirements:** IT administrators will have comprehensive access within the system. This includes full permissions to manage all aspects of the system’s configuration, user management, and maintenance. Admins will also have the ability to access logs and audit trails for security monitoring and troubleshooting.

These measures ensure that the system remains adaptable to changing requirements and technologies, while empowering IT administrators to maintain system efficiency and security effectively.

#### 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:** Users will be required to log in using a unique username and a strong password combination. Multi-factor authentication (MFA) will also be implemented to add an additional layer of security, requiring users to verify their identity using a second factor, such as a phone number or email verification.
* **Connection and Data Security:** Secure connections will be established using HTTPS with SSL/TLS encryption to protect data transmitted between clients and the server. This prevents interception and unauthorized access to data in transit. For data at rest, encryption techniques such as AES (Advanced Encryption Standard) will be used to secure sensitive information stored in the database.
* **Brute Force Protection:** To protect against brute force attacks, the system will implement account lockout mechanisms. After a defined number of unsuccessful login attempts, the user’s account will be temporarily locked, and an alert will be sent to both the user and the system administrators to investigate the potential security threat.
* **Password Recovery:** In case a user forgets their password, a secure password recovery process will be implemented. This will involve identity verification steps such as answering security questions or confirming a code sent via email or SMS. After verification, users will be allowed to reset their password securely.

These security protocols ensure robust protection against unauthorized access and data breaches, maintaining the integrity and confidentiality of user information and system data.

### 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. This ensures secure access and verifies that the user's identity matches the records before granting access to the system.
* The system shall allow users to schedule, modify, and cancel driving lessons through both the online interface and by phone. This flexibility supports diverse user preferences for interaction with the service.
* The system shall provide a dashboard for users to track their progress in courses and practice tests. This feature helps users monitor their learning and preparation for the driving test.
* The system shall enable administrators to add, remove, or modify the details of the training packages offered. While immediate changes by non-developers are restricted, administrative control over these elements allows for business flexibility and responsiveness to market needs.
* The membership shall generate reports on user activity, including lesson attendance, test scores, and progress tracking. These reports are crucial for administrative purposes and to provide feedback to users on their performance.
* The system shall send notifications to users and administrators about important updates or changes, including DMV regulations or system enhancements. Timely communication ensures all parties are informed and can adapt to changes effectively.
* The system shall ensure that all data exchanges between the client and the server are encrypted using contemporary cryptographic standards. This security measure protects sensitive information from being intercepted during transmission.
* The system shall automatically lock a user account after multiple unsuccessful login attempts and notify the administrator. This feature prevents unauthorized access and alerts administrators to potential security threats.
* The system shall allow users to reset their passwords securely after identity verification through predetermined security questions or verification codes sent via email or SMS. This process helps maintain security while accommodating users who need to recover their login credentials.

These functional requirements define the critical operations and security measures the system must support to meet the needs of DriverPass effectively and 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.)?*

* **Interface Needs**: The user interface should be intuitive, user-friendly, and visually appealing, ensuring ease of use for all functionalities. It should provide quick access to all necessary features, such as lesson scheduling, progress tracking, and accessing training materials, with a clear layout and responsive design to accommodate different devices.
* **Different Users of the Interface:**
  + **Administrators:** Need comprehensive access to manage system settings, user roles, reports, and course content. They should be able to view and edit all aspects of the system to perform administrative tasks efficiently.
  + **Instructors:** Require access to schedules, student progress, and course materials. They should also be able to update details about the lessons they are conducting and communicate with students.
  + **Students:** Need the ability to register, schedule lessons, view course materials, take practice tests, and track their progress. They should be able to easily navigate through learning resources and booking systems.
* **Functional Capabilities for Each User:**
  + **Administrators should be able to:**
    - Monitor user activity and system health.
    - Update or remove training packages.
    - Generate and access various administrative reports.
  + **Instructors should be able to:**
    - View and manage their teaching schedules.
    - Access and update student progress notes.
    - Communicate changes or updates to students.
  + **Students should be able to:**
    - Easily schedule, reschedule, or cancel lessons.
    - Access learning materials and results of practice tests.
    - View their progress and upcoming schedules.
* **User Interaction:**
  + **Desktop and Mobile Browsers:** The interface should be accessible through all major web browsers (e.g., Chrome, Firefox, Safari, Edge) and optimized for both desktop and mobile platforms to ensure a seamless experience across devices. This includes responsive design elements that adapt to the screen size and orientation of the device being used.

The design of the user interface should cater to the specific needs of each type of user, ensuring that it is efficient, effective, and enhances the overall user experience. This includes making sure that navigation is intuitive, the visual layout is clear, and that users can perform their required tasks with minimal effort and learning curve.

### 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?*

* **Technological Proficiency:** We assume that all users (administrators, instructors, students) possess basic technological proficiency, including the ability to navigate web interfaces and use common online tools. This ensures they can interact effectively with the system’s functionalities without requiring extensive training.
* **Device Accessibility:** It is assumed that users have access to internet-connected devices capable of running a modern web browser. This includes smartphones, tablets, or computers, ensuring they can access the system’s web-based interface from anywhere.
* **Robust Data Handling:** There is an underlying assumption that the backend database is robust and capable of handling the expected load of user queries, data entries, and retrievals without performance degradation. This is essential for maintaining smooth operation and a good user experience.
* **Internet Connectivity:** We assume that users will have continuous access to a stable internet connection to interact with the system, especially for real-time functionalities like scheduling or modifying appointments and accessing the latest training materials.
* **Security Awareness:** The design assumes that users are aware of basic security practices, such as safeguarding login credentials and recognizing the importance of logging out after sessions, to help maintain the system's security integrity.

These assumptions help guide the design and development process, ensuring that the system meets expected functionalities while addressing potential user and technological variables.

### 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?*

* **Resource Allocation:** The system development is constrained by the availability of skilled personnel. Limited human resources may delay certain aspects of development and deployment, particularly in areas requiring specialized expertise.
* **Budget Constraints:** Financial limitations impact the scope of technology and features we can implement. High-end solutions like advanced analytics or custom-built interfaces may be unfeasible, necessitating compromises on some of the more ambitious system capabilities.
* **Time Constraints:** The project timeline is tight, restricting the amount of testing and refinement that can be done before launch. This may affect the system’s stability and the thoroughness of user testing, potentially leading to undiscovered issues post-deployment.
* **Technological Limitations:** While the system aims to be compatible across various platforms and devices, disparities in device capabilities, operating system versions, and browser support could affect functionality and user experience for some users.
* **Scalability Challenges:** Given budget and time constraints, the initial deployment of the system may not fully support scalability to handle a significant increase in user numbers or data volume without additional investment and system upgrades.
* **Integration Complexity:** Integrating the system with existing databases or third-party services (like DMV updates) may be more complex than anticipated, potentially leading to longer development times and increased costs.

These limitations outline potential challenges that could impact the system’s effectiveness and efficiency. Identifying these early helps in planning mitigation strategies and setting realistic expectations with stakeholders.

### 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 white screen with black lines

Description automatically generated