Southern New Hampshire University

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?

The objective of this project is to create an integrated system for DriverPass, a driving training company owned by Liam. Liam's vision is to address the existing gap in driver training by providing an all-encompassing solution for individuals preparing for their local department of motor vehicles (DMV) driving tests. The system's core purpose is to deliver online courses, practice tests, and practical on-road training, thereby enhancing students' driving skills and increasing their success rates in passing the driving test. Key functionalities of the system include efficient management of driving lesson bookings, instructor assignments, user accounts, and training resources. By developing this system, Liam aims to offer a comprehensive and superior driving training experience to his clientele, thereby filling the void in the market.

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 has a specific requirement for the system's availability, needing it to operate seamlessly in both online and offline modes. However, a key concern for the development team revolves around potential conflicts arising from saved changes in these different modes. To address this, the team recommends running the system via a cloud-based interface, which offers the necessary flexibility to handle these scenarios effectively.

Additionally, the paramount issue of security looms large. Different employees will require varying levels of access to sensitive information within the system. Hence, robust security measures will be implemented to ensure that only authorized personnel can access and manage private data.

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 completed system should have the following capabilities:

1. Package Selection and Disabling: Users can select different training packages, and the system allows the client to disable fully booked packages.

2. Test Progress Tracking: The system displays students' test progress, including ongoing, completed, or pending tests. It provides information such as test name, time taken, score, and status (not taken, failed, passed, or in progress)

3. Driver's Notes: Users can access driver comments and lesson durations for their training sessions.

4. Driver-Consumer Match Tracking: The system tracks driver assignments to consumers, along with the scheduled time and vehicle.

5. Offline Access: Users can access system information and functionality offline.

These goals translate into measurable tasks for system design, including package management, progress tracking, driver notes retrieval, driver-consumer matching, and offline functionality. The system design must ensure efficient scheduling, progress monitoring, and data synchronization for both online and offline usage.

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?

System Update Frequency:

Regular updates are essential to address bug fixes and security vulnerabilities promptly. Timely updates are also crucial to ensure alignment with any changes in DMV guidelines for the benefit of DriverPass students.

System Environment:

The system's effectiveness is deeply tied to its web-based nature, enabling easy access and convenience for users across devices. This web-based architecture facilitates seamless updates and maintenance, aligning with DriverPass's goal of providing up-to-date and secure services.

Performance Expectations:

Efficient system operation, especially in managing requests to and from servers, is imperative. Speed is of the essence at DriverPass, particularly for tasks like exams, to provide a responsive user experience.

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 must fulfill the following specifications:

1. It should be compatible with multiple web browsers, including Chrome, Microsoft Edge, and others.

2. The user interface should be responsive, ensuring proper display and functionality on various mobile devices.

3. The backend infrastructure must incorporate a database for efficient data storage and management.

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?

The system effectively distinguishes between user email and password, retaining input case sensitivity to bolster security. Furthermore, it implements a lockout mechanism that temporarily restricts user access after a predefined number of unsuccessful login attempts. Users can reset their passwords through email verification, adding an extra layer of protection to the system's security measures.

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's flexibility allows for the addition, removal, and modification of users without necessitating changes in the underlying code. This functionality is achieved through the implementation of code, POST requests, and controllers. To accommodate platform updates, the system remains responsive to requests from programmers. IT administrators require comprehensive access, including user account management, password control, and the ability to remove unavailable employees.

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?

When users log in, whether they are students or administrators, they must provide their username and password to gain access. To ensure secure data exchange between the client and server, HTTP is a fundamental requirement. In the event of a "brute force" hacking attempt, an administrator will receive an alert after a specified number of consecutive failed login attempts, within the range of 1 to 10 attempts. When four failed attempts occur, two actions will follow: the login input will temporarily become inaccessible for the user, and an alert will be sent to the administrator. If a user forgets their password, they can initiate a password reset request. The system will then send the necessary password reset information to the user's registered email address, enabling them to complete the password reset process.

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 will validate consumer selection among the three available packages specified by the client. It will also verify and store consumer details, including first and last name, address, credit card information, phone number, and state. While the system primarily operates online, select study materials will be accessible offline. User login authentication will determine whether the user is a student or administrator. The system will present the three client-suggested package types and allow the client to disable any unavailable packages. Disabled packages will be visible to consumers. Additionally, the system will offer a password reset feature for users. Exam progress and scores will be displayed for students, and user login information will be confirmed for access. The system will remain adaptable to accommodate any DMV guideline changes.

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.)?

User Interface:

- Registration (for new users)

- Instructor's notes

- Exam

- Home Page

- User Information (including password updates, address changes, etc.)

- Exam status

- Contact information

- Access to Grades

Administrator Interface:
- User Management (Adding, modifying, or deleting users, especially in cases of forgotten passwords and other account-related tasks)

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?

- Continuous availability of the system.

- Predominantly DMV student users.

- Regular updates to align with DMV guidelines.

- Potential development of a future DriverPass mobile app.

- Compliance: Legal and regulatory requirements are not explicitly addressed.

- Development Expertise: The development team is assumed to have required skills.

- DMV Integration: The DMV's data integration capabilities are assumed to support system needs.

- Compliance: The system is assumed to comply with laws and regulations.

To create a comprehensive system, it is crucial to further refine and expand upon these aspects, address additional requirements, and validate assumptions through collaboration with stakeholders and domain experts during the system development process.

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?

- A reliable internet connection is imperative for system access, as it governs data updates, additions to the database, and overall accessibility.

- Legal Compliance: Specific legal requirements and compliance complexity are not fully explored.

- Time and budgetary constraints are inherent, as per the client's stipulations.

- Maintaining an accurate schedule for students, considering the availability of 10 cars, is a key operational requirement.

- The system's content aligns with DMV guidelines, given DriverPass's role in aiding DMV exam preparation.

Effective planning and collaboration are essential to address these limitations and align project goals with available resources and budgets.

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.