**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?*

DriverPass wants to fill the void in the market to provide a thorough training system that increases driver test preparation and driving education. The objective of this project is to design, develop, and implement a cloud-based system which provides students with access to online training, practice exams, a way to schedule in-person lessons and a tracker with their progress. Liam, the owner of DriverPass wishes to increase the pass rate in driving tests for his students by offering personalized on-the-road training, structured online learning material and by becoming integrated with the DMV and their regulations.

**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 team believes many individuals fail their driving test due to inadequate preparation and they feel like they can change that by filling gaps in the market. DriverPass wants to provide their students with a structured learning environment that includes online lessons, practice exams, and in-person driving lessons personalized to their needs. To get this done, the DriverPass system will have the following key components:

* A cloud-based platform that can be accessed by mobile devices and computers.
* A system to schedule students’ in-person lessons with options to book, modify and cancel.
* A feature that tracks test progress and completed lessons.
* Access control based on role (Administrator, IT personnel, secretaries and students).
* A system that generates logs of student progress, modifications and reservations.
* Secure payment processor.
* Notifications from the DMV when updates are made to ensure students are learning the proper material.

**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?*

After completion, the DriverPass system should be able to:

* Give students training materials and be able to schedule lessons all from a cloud-based platform.
* Students will have the ability to book, modify, cancel, and reschedule driving lessons.
* A way to assign instructors and their vehicles for driving lessons.
* Progress tracker for online practice exams with status (not taken, in progress, failed, passed).
* For accountability, records of reservations and modifications are to be maintained.
* Only administrators and IT personnel can manage user accounts and role-based access for other accounts.
* All data is secured according to regulations.
* Notifications for DMV policy changes/updates to keep material up to date.
* Reports for tracking students with instructor feedback and for business analytics.
* Support for training package modification if needed and new features.

**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?*

 DriverPass’ ideal system environment is a web-based application accessible on desktops and mobile devices. It should have a fast response time and be able to support multiple users actively at the same time with little to no latency. To prevent service interruptions for the majority of users, system updates should be scheduled for non-peak hours.

**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 should run on all web browsers. The backend will need a database to store customers’ data, track users’ status, progression, and other activities, and keep logs. A cloud-based infrastructure is the perfect infrastructure for this web-based application because it eliminates issues of scalability and reliability.

**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?*

A secure login system will handle user authentication with an optional but highly suggested two-factor authentication. The system will be designed to distinguish different users from their unique credentials. It will provide real-time notification for system errors, critical administrative alerts, and failed login attempts, with a warning message indicating how many attempts are left until the account is locked out. Also, the login window will inform the user that all inputs are case-sensitive.

**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?*

Yes, the system will allow Admins to add, remove, and modify users’ accounts or schedule appointments without changing the code. The system will adapt to the platform’s updates seamlessly. Also, IT personnel will have unrestricted access to manage user roles and permissions.

**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?*

To log in, users must use a unique username and password. There will also be optional but highly suggested two-factor authentication. There will be encryption for all data exchanges between the client and the server. If an account has three failed login attempts, then the account will be locked out until the account user's identity can be verified for reactivation. If the user forgets their password, a password reset mechanism will be available.

**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 authenticate all user credentials upon login.
* The system shall lock accounts after three failed login attempts, and the user will be notified.
* The system shall allow users to schedule, modify, and cancel appointments.
* The system shall enable Admins to manage user access and roles and make changes on behalf of the user.
* The system shall have a record of all training scores and track all test attempts that are available to the user.
* The system shall give users the option to purchase service packages and register for online classes.
* The system shall provide real-time notifications for system issues, appointment changes, and all test scores.

**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.)?*

* **Students** can **s**chedule appointments, view all past purchases, track all training progress, and take exams.
* **Adim** will manage user accounts, oversee all training, assign driver instructors to students, modify appointments, and track user schedules. (The Owner will be a special Admin account, which is permitted for all admin access and more.
* The **IT** personnel can **r**eset locked accounts, manage system security, and oversee platform updates.

**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?*

One assumption is that the users will have access to an internet connection and have up-to-date web browsers. Another assumption is that the system can scale when user demand increases. One more assumption is that all the DMV updates will be seamlessly integrated into the system, and all users will get a notification. Lastly, making a strict mobile application such as DriverPass will increase accessibility for users.

**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?*

Obviously, the biggest constraint is the budget, which impacts the future development of the application and cloud storage capacity. Without a dedicated mobile app, the strict reliance on the web-based platform might limit user engagement. Also, if a surge of high user traffic occurs, performance could suffer. The system functionality would be disrupted if a server breach occurred.

**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 graph of a project gantt chart

AI-generated content may be incorrect.*