CS 255 Business Requirements Document Template

Paloma Rodriguez

May 29th, 2023 SNHU CS 255

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 client is named DriverPass and the purpose of the driver system is to provide a more advanced platform for efficiently managing and overseeing the process for driver's license and permit tests. Its main objective is to streamline the process of testing individuals for their driver's licenses and permits while ensuring compliance is in mind with testing regulations and the system design is a system for managing test sessions, scheduling appointments, tracking test results, and report generation. DriverPass wants their system to offer a platform for managing and administering driver's license and permit tests and include features such as registering users, scheduling tests, tracking test results, generating reports, and ensuring compliance with testing 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 client DriverPass wants their system to facilitate the administration of driver's license tests making it easier for administrators to manage test sessions and for the test takers to be able to register for tests, schedule appointments, complete tests, and receive timely results. They also want to make some services automated and provide a user-friendly interface and pretty much want to improve the overall experience for both test administrators and test takers but have the same integrity and security of the testing procedures. They want features like registering users, scheduling tests, tracking test results, generating reports, and ensuring compliance with testing regulations.
* The different users of the system will be the administrator, the test taker, instructor.
* The different components needed will be test session management, the scheduling, the actual testing portion, and then the scoring of the test, generation of report of the test, compliance assurance and accessibility for users.

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?*

When the system is completed it will be able to allow users to do their registration and provide management for those accounts, test session management, scheduling for test and administration for those tests, scoring and reporting, compliance with regulations and accessibility for users with needs.

Some tasks that can be used:

* + Implement a user registration process that allows users to create accounts and input their personal information.
  + Create and manage test sessions like test duration and availability.
  + Implement functionality for test takers to select and reserve test appointments
  + Design an interface for test takers to access test materials and complete the tests within the time limit.
  + Have an automated scoring system that evaluates test taker responses based on the passing criteria
  + Create reporting functionality to generate report of the test
  + Have in place an identity verification to ensure that the right person takes the test.
  + Incorporate accessibility features to accommodate users with disabilities like support for screen readers or alternative formats for test materials.

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 DriverPass system needs to run in a web based environment accessible through commonly used browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge.
* The system should aim for fast response times to ensure a smooth user experience. The specific speed requirements may depend on factors like the complexity of operations, network conditions, and user expectations.
* The frequency of system updates may vary based on the needs of the client and any regulatory or policy requirements. We will need to consider bug fixes, feature enhancements, security patches, and regulatory compliance updates. The system should have an update schedule that it will follow to address emerging issues, ensure compatibility with evolving technologies, and maintain system performance and security.

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 be capable of running on platforms that are commonly used like windows, linux or macOS but some back end tools and components to consider to support its functionality would be the web server like apache HTTP server, nginx, or microsoft IIS to handle HTTP requests and serve web pages. A database to store and manage data related to user accounts, test sessions, scheduling, test results, and other relevant information and security tools to ensure the privacy is kept and tools like firewalls and intrusion detection systems can assist with this.

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 in the DriverPass system we can use the usernames assigned to each user during the registration process or their personal information like date of birth, address, full name etc. Input case-sensitivity can be used if usernames were the primary identifier but can be softened if personal information would be the primary identifier. The system should inform the admin of a problem if there's problems with the system itself if the database fails, server crashes. User issues where there's lockouts or too many password attempts to ensure availability and security. Testing issues where the test is not able to generate reports effectively to the user and if there are any compliance violations where testing policies aren't being met.

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?*

If changes to users can be made without changing code will depend on the design of the system but if we design the code in mind to be flexible, it is possible to make changes without altering the code itself. To adapt to platform updates and to follow best practices in software development and system maintenance we need to stay on track with updates, patches, and security advisories related to the platforms and frameworks used in the system and regularly update the system to incorporate bug fixes, security patches, and performance enhancements provided by the platform developers. We should follow industry standards and best practices while developing the system to ensure that the system follows widely accepted practices and remains compatible with future platform updates that align with these standards. The access required for the IT admin will depend on the responsibilities assigned to them but could include user information, system configuration, security management and system monitoring.

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?*

What should be required for a user to log in would be their credentials and they should provide their username and password during the login or their personal information. To secure the connection and data exchange between the client and the server we can incorporate transport layer security which will provide encryption and integrity checks to establish a secure connection between the client and the server but if there was a "brute force" hacking attempt on a user's account, we can lock the account that after so many login attempts, you are then locked out or even send a notification to the user that there was attempt on their account so they can take action to changing their password and enabling additional security measures like two-factor authentication. If a user forgets their password, we can offer a password reset function or account verification so that they can prove they are who they say they are.

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 log user activities and system events for auditing and troubleshooting purposes.

The system shall provide an administrative interface for IT admins to manage user accounts.

The system shall allow users to login

The system shall provide a searchable database of available test locations and times.

The system shall allow users to register for a new account.

The system shall provide a "forgot password" functionality for users to reset their passwords.

The system shall allow users to schedule driving tests.

The system shall allow users to view their scheduled driving tests.

The system shall generate and provide test results to users upon completion of a driving test.

The system shall allow users to request a retest if they fail a driving test.

The system shall provide a secure and encrypted connection for all user interactions.

The system shall store and maintain user account information securely.

The system shall provide an administrative interface for IT admins to manage user accounts.

The system shall notify admins of critical errors or failures affecting system functionality.

The system shall enforce security measures to detect and respond to potential hacking attempts.

The system shall ensure data integrity and accuracy throughout user interactions and data storage.

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.)?*

The DriverPass system needs to fulfill the needs of various users like test takers who need to schedule and take driving tests where they will be able to access through the interface to register for an account, log in, schedule driving tests, view their scheduled tests, access test results, and request retests if necessary and should be able to do this through a web browser using a computer or mobile device. Admin and the IT staff will also need to manage user accounts and ensure the system's proper functioning so they would need to have an administrative interface to manage user accounts, handle system errors, monitor system logs, and perform administrative tasks. This should also be done through a computer and mobile device but can be more limited information on mobile. The interface should be responsiveness, usability, and accessibility to users accessing the system through different devices and platforms. It should provide an intuitive and user-friendly experience, allowing users to navigate through the system, input necessary information, and perform desired actions efficiently.

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?*

I have not mentioned the layout, and navigation of the user interface or the payment processing. If the system involves payment for services such as scheduling tests or retests, the design would need to include a secure payment processing mechanism. Assumptions made will be the user has some sort of proficiency with basic computer skills and they have internet connectivity or they have the correct device compatibility.

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?*

There can be issues with scalability because as the user base grows and the system becomes more in use, we should consider that the system can handle increased user load to avoid challenges and potential performance issues. There could be security limitations and we should put in place enhanced security measures like secure coding, encryption, access control to avoid any cyber threats. As far as resource limitations, we should consider the company budget for development team size, expertise and hardware resources that may impact the system's design and implementation and time constraint where the project end time is around May 10th and could affect the overall system complexity. Navigating these limitations requires a balanced approach and adaptation throughout the development process can help mitigate these limitations and ensure the successful delivery of the system within the given constraints.

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.*

