Grant McCord
October 6, 2024
CS-255

CS 255 Business Requirements Document

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 purpose of this project is to develop an online system for DriverPass, represented by Liam (the owner) and Ian (the IT officer), to enhance driver education and prepare students for the DMV driving test.
- This system will address the high rates of student failures by improving access to driver training through online classes, practice tests, and on-the-road training, fulfilling a market need for better driver education.

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 system will provide an online platform for driver training with courses, practice tests, and on-the-road training.
- The system will enable online reservations, cancellations, and modifications for driving lessons.
- The system will include a secure interface for customers and administrators to protect sensitive information.
- The system will allow users to access data from any device, online and offline.
- The system will log all reservation actions and provide activity reports.
- The system will connect with the DMV to maintain current information and ensure adherence to regulations.
- The system will manage customer information, including contact details, payment information, and appointment history.

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?

System Capabilities

- The system shall use a user interface for students to access course materials and learning modules.
- The system shall administrate practice tests providing instant feedback for students.
- The system shall facilitate reservations for driving lessons with instructors and vehicles.
- The system shall offer various training packages to customers, with options to change offerings.
- The system shall implement role-based access controls and secure password resets.
- The system shall alert administrators about updates from the DMV related to driver training.
- The system shall facilitate secure online payment transactions.
- The system shall track user activity with detailed logs.

Measurable Tasks

- Develop a library of online courses covering driving topics, traffic laws, road signs, and safe driving techniques.
- Create web-based interfaces for online classes and reservation systems.
- Design a bank of practice tests simulating the format and difficulty of DMV exams.
- Build a reservation system for customers to schedule, modify, and cancel driving lessons.
- Add an administrative interface for package management.
- Develop user authentication mechanisms based on role-based permissions.
- Integrate the system with the DMV for updates on regulations.
- Develop a logging system to capture and store user actions related to records.
- Integrate payment processing gateways for online transactions and receipts.
- Implement offline access capabilities and resynchronization of data from offline updates.
- Conduct testing of all system functions, including usability, performance, and security testing.
- Develop report generation capability for activity reports, reservation summaries, user change history, and user performance metrics.

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 system must respond to user interactions, such as making or modifying reservations, within 2 seconds.
- The initial application interface should load in under 3 seconds upon login.
- The system must support at least 100 simultaneous users without performance degradation and be capable of handling up to 2,000 reservations per day.
- Data updates should occur in real-time to ensure users have the most current information.
- The system should be available 24/7 with minimal downtime.
- The system shall automatically backup all data every 24 hours.
- The system shall provide a web-based interface on cloud infrastructure, which can be accessed from any computer or mobile device.
- The system shall provide real-time notifications for DMV updates.

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 shall be web-based and compatible with all major browsers (Chrome, Firefox, Safari, Edge) and support connection from both desktop and mobile platforms.
- The system shall integrate with cloud services for hosting and scalability.
- The system shall use a relational database management system to support data storage and retrieval.
- The system shall contain RESTful APIs for integration of 3rd party services such as DMV notifications and payment processing.
- The system shall implement secure web browsing protocols for user connections to protect user data.

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 shall ensure data accuracy in user records and reservations.
- The system shall support different user roles each with specific access rights and functionalities, keyed from username login.
- The system shall treat usernames (or email addresses) as case-insensitive for user login.
- The system shall treat passwords as case-sensitive.
- The system shall synchronize data between offline and online activities.
- The system shall perform correct calculations of driving lesson durations and scheduling.
- The system shall notify the admin of multiple failed login attempts, critical system errors, data integrity issues, or automated system failures.
- The system shall validate user input for registration and appointments to prevent errors.

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 shall allow the IT admin to add, remove, or modify user accounts and roles through an administrative interface.
- The system shall support configuration of user roles and permissions through a management dashboard.
- The system shall have a modular architecture to allow component swapping without affecting other components.
- The system shall include a mechanism to apply updates from the DMV without requiring a system restart.
- The IT admin shall have access to manage user accounts, including password resets and user activity logs.

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?
- The system shall require users must enter a unique username and secure password to log in.

- The system shall support optional two-factor authentication (2FA).
- The system shall use HTTPS for secure client-server connections.
- The system shall support data encryption in transmission and rest.
- Secure session management practices, including session timeouts, shall be implemented.
- The system shall support user account lockouts after a predefined number of failed login attempts.
- Admins shall be alerted of multiple failed login attempts.
- Users shall be able to reset their passwords via a secure email link.
- The system shall send notifications to users for account lockouts and password reset requests.
- The system shall require use of strong passwords by all users.

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 allow users to take practice tests that simulate the DMV exam.
- The system shall validate user credentials when logging in.
- The system shall allow users to create and manage their accounts.
- The system shall track and display user progress through online courses.
- The system shall allow users to customize their learning plans.
- The system shall enable users to schedule, modify, and cancel driving lesson reservations online.
- The system shall require customers to provide a drop-off location, defaulting to the pickup location unless specified otherwise.
- The system shall allow users to leave feedback and ratings for their driving instructor.
- The system shall provide options for customers to select from different driving lesson packages.
- The system shall notify users if a selected driver becomes unavailable.
- The system shall allow staff to manage appointments on behalf of customers.
- The system shall track and display the status of driving lessons, including scheduled, completed, and canceled lessons.
- The system shall connect to the DMV for updates on rules and regulations.
- The system shall log all user activities.
- The system shall maintain an audit trail of all changes made within the system.
- The system shall allow users to contact the company through a contact form.
- The system shall offer a secure payment processing feature.
- The system shall encrypt all sensitive data during transmission and in storage.
- The system shall notify users about account lockouts.

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.)?

- *Access Functions*
  - The interface shall provide students access to online instructional materials and practice tests.
  - The system shall provide a web interface accessible via mobile devices and desktop computers.
  - The interface shall offer search functionality.
  - The interface shall support means for students to contact DriverPass.
  - The interface shall support means for staff to leave messages for students.
- *User Management*
  - The system shall provide an input form for students (or staff) to fill in student information.
  - The interface shall allow students to create and manage their accounts.
  - The system interface shall allow the IT admin to manage user accounts.
  - The system interface shall enable the IT admin to generate reports on user activity and system performance.
- *Test and Progress Tracking*
  - The system shall display online test progress for each student, including test name, time taken, score, and status.
  - The system shall allow students to view and manage personal information such as names, addresses, phone numbers, emails, student photos, and special needs.
- *Scheduling and Appointments*
  - The interface shall enable students to schedule, modify, and cancel driving lessons.
  - The system interface shall enable staff to view and manage all scheduled appointments and driving lessons.
  - The system interface shall allow staff to modify student reservations and manage inquiries.
- *Feedback Mechanisms*
  - The interface shall allow students to leave feedback and ratings for instructors.
  - The interface shall allow driving instructors to leave feedback for students.
- *Driver Management*
  - The system shall provide a driver availability interface that displays the current availability of each driver.
  - The system shall enable users to filter drivers based on availability, location, and vehicle type.
- *Notifications and Alerts*
  - The interface shall display notifications and alerts clearly to users.
  - The system shall support notifications through push alerts and/or email.
  - **Security**
  - The interface shall provide a secure login page with options for password recovery.

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?

- The current schedule suggests a waterfall development method, whose linear and inflexible nature can pose significant risks in system development, especially in environments where requirements frequently change. This rigidity can lead to increased costs and delays if revisions are needed late in the process. It is assumed that by proposing this approach, the project team is aware of the schedule's vulnerabilities and is prepared to manage the associated risks.
- The design assumes potential integration with the DMV to receive updates on regulations and testing requirements. However, the specific methods and mechanisms for this integration have not yet been defined. Further analysis will be needed to determine the technical requirements and protocols for seamless data exchange.
- Compatibility with commonly used web browsers and operating systems is assumed, but specific versions or legacy systems were not addressed.
- The design assumes that data synchronization will be necessary to ensure consistency between online and offline records. However, the specific processes and technologies for real-time or periodic synchronization have not been established.
- The design lacks specific statements regarding adherence to data protection regulations. This includes requirements for obtaining user consent for data collection, and ensuring user rights to access and delete personal data.
- The design does not specify requirements for password strength during user account creation and login.
- The design does not account for variations in regulations, driving laws, or user requirements based on different states, countries or regions.
- The budget for hosting cloud services is not known at this time; it is assumed that sufficient funding will be allocated to cover the costs of reliable and scalable cloud infrastructure.
- It is assumed that non-developers will have limited capability to add or modify system modules without requiring developer intervention. This was a potential requirement for future updates.
- It is assumed that the system will handle all aspects of data backup and security management without requiring direct involvement from the organization.

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 was no mention of specific accessibility features to support users with disabilities.
- Limited knowledge about potential integration with the DMV and other third-party services may hinder the system's ability to provide real-time updates.
- There may be constraints on how easily non-technical users can customize or adapt offerings without developer assistance.
- The design currently lacks detailed strategies for ensuring compliance with various data protection regulations.
- Financial limitations may restrict the scope of features, integration options, and the hiring of additional personnel.
- There may be a shortage of personnel with the necessary skills to implement advanced features.

- Resources for maintenance and support are not known; if limited, this could lead to delays in addressing user issues or system updates.
- The system may encounter issues with compatibility across different web browsers and versions between mobile and desktop computers.
- Limitations in mobile device capabilities may impact usability.

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.

Powered by: onlinegantt.com