Southern New Hampshire University

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 create an online platform for users to sign up for different packages of driving lessons and/or additional materials to take advantage of the current void in the market with regards to training students for the driving test at their local department of motor vehicles (DMV).
- The client is *DriverPass* with the owner being Liam and information technology (IT) officer, Ian.
- The client wants their system to be able to:
    - Provide online classes as well as practice tests
    - Allow on-the-road training reservations
    - Access online from a computer or mobile device
    - Support offline viewing of reports/data exports (such as Excel).
    - Include role-based security permissions for employees and users

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* wants the system to provide a comprehensive training solution that combines classroom, online, and practical driving lessons.
- The problem they want to fix is that many people fail DMV driving tests due to a lack of an effective training institution in their area.
- Different components needed for this system are:
    - Student registration (to collect personal/contact/payment/pickup info)
    - Lesson scheduling/reservations (2-hour sessions, matched with driver, car, and time)
    - Package Options:
        - Package 1: 6 hours driving
        - Package 2: 8 hours driving + DMV policy lesson
        - Package 3: 12 hours driving + DMV policy lesson + online course + practice tests
    - Package management (disable packages, future ability to add/remove)
    - DMV updates support (rules, policies, test questions)
    - User account management (reset passwords, block accounts)
    - Activity tracking/logs (who made, changed, or canceled reservations)
    - Student interface for viewing progress (tests, scores, lesson notes)
    - Driver notes after each session (time, comments)

- Users permissions in the system:
    - Liam (Owner): full access, reporting
    - Ian (IT Officer): full admin control (account resets, blocking users)
    - Secretary: enter student info, make/cancel reservations
        - Schedule appointments by phone/in person

- Students/Customers: create account, register, schedule/cancel/modify appointments, take tests

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 complete, it should be able to:
  - Provide web/cloud-based access with secure login
  - Allow students to:
    - Create/modify an account
    - Reset password automatically
    - Register for packages
    - Schedule, cancel, or change reservations online
    - Take practice tests and track progress (with status: not taken, in progress, failed, passed)
    - View driver notes and lesson schedule
  - Allow employees/admins to:
    - Manage appointments (assign car, driver, time)
    - Track who made/modified/canceled reservations
    - Block/reset accounts when necessary
    - Export reports (Excel)
    - Allow DMV integration for rule/policy/test updates with notifications
    - Provide flexible package management (disable or update offerings)

- Measurable tasks that need to be included in the system design to achieve this include:
  - Account creation, login, and role-based access tested
  - Reservation system that supports 10 cars/drivers with no conflicts
  - Reporting functions (activity logs, lesson notes, test progress) available
  - DMV updates appear automatically in the system
  - System passes security testing (role restrictions, password resets)
  - Successful delivery of the product by May 9 with client sign-off on May 10

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?

- Due to the online nature of the business, the system will need to run in a web based environment that is accessible through modern web browsers on desktop and mobile devices.

- The system should run quickly and efficiently, with page loads and transactions completing within two to three seconds under normal network conditions to ensure a smooth user experience.
- The system should be updated every 3-4 months for system updates and feature enhancements. The system should be updated immediately in the case of emergencies such as critical security threats or dangerous bugs.

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 the most common and widespread platforms which include Windows, MacOS, and Linux/Unix to reach the most people and allow for flexibility for both users, developers, and administrators.
- The backend does require tools to support this application. Specifically, it will need a relational database management system (such as MySQL) to store user data, scheduling information, and system logs, along with a web server (such as Apache) and application framework support (such as Dropwizard or Spring Boot) to handle server-side operations and API communication.

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?

- Different users will be distinguished based on their unique credentials such as real name, username, email, or address linked to their assigned role based account (student, instructor, secretary, admin, owner).
- The input will be case sensitive to ensure secure and accurate authentication.
- The system should inform the admin of a problem when it is deemed a security threat or a threat to the running of the entire system. This includes repeated failed login attempts, unauthorized access attempts, system errors, or downtime that could affect performance or data security.

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?

- Changes can be made without changing the code. User management which includes adding, removing and modifying users is in the control of administrator accounts. They can dynamically manage users through a database connected interface.
- The system will adapt to platform updates by using modular architecture along with API based components to allow for adaptability and easy updating without disrupting the main functions.
- IT needs access to the entire system which includes database management, user role assignment, security monitoring, and configuration control, to be able to maintain and update the system efficiently.

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?

- Requirements for the user to login include users having a registered account and providing a valid username and password, which the system verifies against encrypted credentials before granting role-based access to the correct dashboard.
- The connection or data exchange between the client and the server can be secured by protecting it with HTTPS (SSL/TLS) encryption. Sensitive data such as passwords and payment info can be protected using encryption in transit and at rest with secure hashing.
- If there is a "Brute Force" hacking attempt, the system will detect the multiple failed login attempts and lock the account, capture the suspicious activity, and notify administrators for a possible brute force attack.
- If a user forgets their password, they can click on *forgot password* to receive a time sensitive password reset link to their verified email address. They can then create a new secure password that meets security complexity standards.

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 create accounts by entering personal, contact, and payment information.
- The system shall validate user credentials at login and allow users to reset forgotten passwords securely.
- The system shall provide role based access, assigning specific permissions for students, instructors, secretaries, administrators, and the owner.
- The system shall allow students to register for training packages (of 6, 8, or 12 hours) and enable administrators to enable or disable packages as needed.
- The system shall allow students and secretaries to schedule, modify, or cancel driving lessons online or in person.
- The system shall assign and track each lesson's student, driver, car, date, and time to prevent scheduling conflicts.
- The system shall allow drivers to enter lesson notes and comments, and students to view completed lesson details and feedback.
- The system shall allow students to access online courses and practice tests, view test scores, time taken, and test status (not taken, in progress, failed, or passed).
- The system shall store and display a student's training progress through dashboards showing lessons, tests, and completion status.
- The system shall log all user activity, including who created, modified, or canceled reservations, for auditing purposes.
- The system shall allow the owner and administrators to generate, view, and export reports (such as activity, progress, and scheduling) in a well used system such as Excel.
- The system shall allow the IT officer (administrator) to manage user accounts, including creating, resetting, blocking, or deleting accounts.

- The system shall connect with DMV systems to receive updates about rules, policies, and test questions, and notify administrators when updates occur.
- The system shall provide a contact form or communication page to allow students to reach DriverPass staff.
- The system shall store all business data (users, cars, schedules, tests, and packages) securely in a centralized online database.

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.)?

- Needs of the interface are that it must be web based and cloud hosted to allow users to access the system from any device with a stable internet connection such as a computer, tablet, or phone. The interface should also be user friendly that allows easy navigation between different pages. The interface should also only allow the user to see what the user is allowed to do based on their role (student sees a different page than an administrator or developer).
- Different users for this interface are:
    - Students and Customers who will access the system using a Web browser or mobile device. These users will need to be able to:
        - Create and manage their user account.
        - Reset password automatically if forgotten.
        - Register for driver training packages.
        - Schedule, modify, or cancel driving lessons.
        - View lesson schedules, driver notes, and session times.
        - Access online classes and take practice tests.
        - View test scores, progress, and lesson status (such as in progress, failed, passed).
        - Receive notifications about DMV updates or lesson changes.

    - Secretary and Office Staff who will access the system using a Web browser. These users will need to be able to:
        - Create new student accounts for phone/in-person customers.
        - Schedule, cancel, or reschedule lessons on behalf of students.
        - Assign students to drivers and vehicles.
        - Update or correct reservation details.
        - Print or export schedules and activity reports for office use.

    - Instructors and Drivers who will access the system using a Web browser or mobile device. These users will need to be able to:
        - View their assigned lessons and schedules.
        - Access student details for each lesson.
        - Enter notes and comments after each driving session.
        - Mark completion of sessions.

    - The IT Officer (Ian) and Administrator who will access the system using a Secure web browser interface. These users will need to be able to:
        - Manage user accounts (create, reset, disable, or delete).

- Maintain system settings, security roles, and permissions.
- Monitor system activity and generate reports.
- Ensure system updates and backups run correctly.
- Respond to technical issues or configuration needs.

o The Owner (Liam) who will access the system using a Web browser (desktop or mobile). He will need to be able to:
- View overall business performance and reports.
- Track lesson statistics, student progress, and instructor activity.
- Enable or disable training packages.
- Review financial data, scheduling trends, and customer feedback.

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?

- It is assumed that everyone will have basic computer skills to access the website and know how to type, navigate the internet, log in/out, access materials, and charge their device.
- It is assumed that everyone has access to a stable internet connection.
- It is assumed that everyone using the website is going to do their drivers test soon (else it defeats the purpose of the system).
- It is assumed that everyone is proficient enough in English to benefit from the system or have a means for translation.
- It is assumed that everyone using the website is legally fit and allowed to drive and pursue a driver's license.

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?

- The system has limited resources such as six workers (Sam, Jennifer, Tony, Clark, John, and Me).
- The system has a limited amount of time (it has to follow the planned schedule/cannot work on it for more than a year)
- The system does not have a specified budget but it cannot use up an unnecessary amount of money.
- The system is limited by technology in terms of what platforms it has to work with.
- The system will allow offline access but will not support data updates or modifications offline which would otherwise cause a duplication of data.
- Since the system is web-based, it is dependent on the internet so a power outage could stop or limit the use of the system.

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.

DriverPass Gantt Chart