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 goal of this project is to create a comprehensive system for DriverPass, a business that wants to transform the driver testing and education sector. The client requests that the system provide scheduling, online instruction, and practice exams for driving examinations.

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 give students an easy-to-use interface to schedule practice exams, access learning materials, and get feedback. The main issue with the existing driver education and testing program is its inefficiency and accessibility. A practice test engine, scheduling module, learning management system (LMS), and feedback mechanism are among the parts of the system that are required.

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?

- Students should be able to safely register and log in through the system.
- Students ought to be able to access and engage with educational resources over the internet.
- Practice test scheduling and progress tracking ought to be made easier by the system.
- It should give teachers and students performance metrics and comments.

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 function as a browser-accessible online application that runs on several web browsers.
- It should be able to support at least 1000 concurrent users and load pages in under two seconds.
- The system should be updated every two months with little to no downtime.

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 Linux, macOS, and Windows operating systems ought to support the system.
- A stable database such as PostgreSQL or MySQL is needed to support the application.

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?

- Usernames will be distinct, and input fields will take case sensitivity into consideration.
- Any problems with login or defects in the system should be immediately reported to the administrator.

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?

- User role modifications have to be feasible without requiring changes to the source.
- Updates to operating systems and browsers should be handled by the system with ease.
- Full access to logs, system configurations, and user management is required for IT administrators.

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?

- A username and password are required for login, though two-factor authentication is an option.
- Using SSL/TLS, data communication between the client and server should be encrypted.
- After five unsuccessful tries to log in, accounts should lock, and users can reset their passwords by email verification if they forget them.

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."

Upon login in, the system will verify the user's credentials.

- Users will be able to sign up and make new accounts on the system.
- Students who have registered will be able to access online course materials using the system.
- Students will be able to plan their practice exams at convenient times thanks to the system.
- The system will monitor the performance and advancement of students throughout time.
- Based on the results of practice tests, the system will create and deliver feedback reports.
- Teachers will be able to upload and manage course materials using the system.
- The system will make it easier for students and teachers to communicate through forums or messaging.

- The payment processing for test scheduling and course registration will be supported by the system.
- Students will be able to check their course progress and forthcoming test schedules on a dashboard provided by the system.
- Instructors will be able to access and examine student performance data thanks to the technology.
- A search feature will be provided by the system to assist users in locating particular courses or resources.
- A wide range of user bases will require multilingual support for the system.
- To protect user privacy, the system must guarantee data encryption both during transmission and storage.
- For security and monitoring reasons, the system must log every action taken by users.

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 will be able to access learning resources and monitor their progress using a dashboard that the system will provide.
- Teachers will be able to plan exams, upload materials, and monitor student progress through the site.
- Both desktop and mobile device accessibility and responsiveness are required for the interface.

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?

- Every user has access to contemporary web browsers and a dependable internet connection.
- The customer has set aside funds and resources for regular system updates and maintenance.

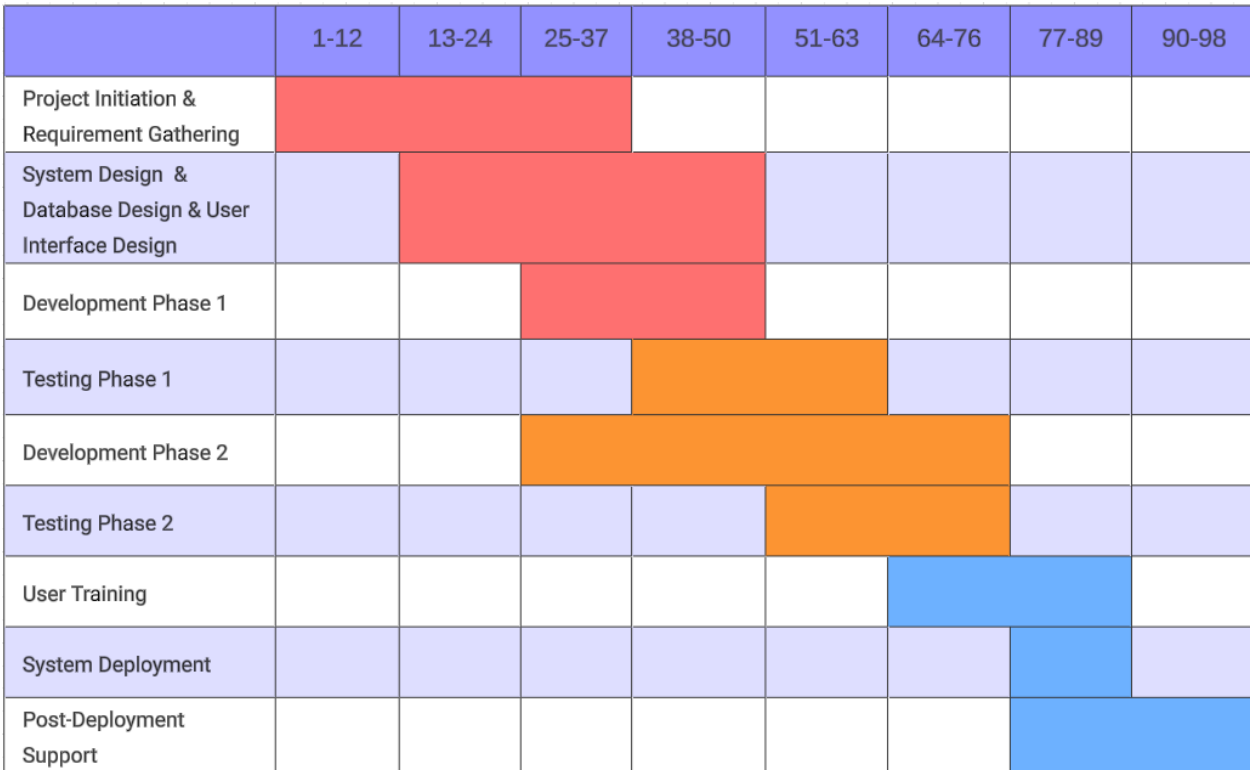
Limitations

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?

- When there are more than 2000 concurrent user sessions, the system's performance could deteriorate.
- Budgetary constraints on initial rollout impact feature completeness.

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.


1-12
13-24
25-37
38-50
51-63
64-76
77-89
90-98
Project Initiation & Requirement Gathering
System Design & Database Design & User Interface Design
Development Phase 1
Testing Phase 1
Development Phase 2
Testing Phase 2
User Training
System Deployment
Post-Deployment Support