Southern New Hampshire University

CS 255 Business Requirements Document Template

Felix Carela
2/6/2024

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?

- The purpose of this project is to design and develop a comprehensive system for DriverPass, a company aiming to improve the driver training process. The system is intended to provide online practice exams, on-the-road training scheduling, and management functionalities to better prepare students for driving tests. The client, DriverPass, requires a system that supports these services effectively and efficiently.

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 seeks to address the high failure rate (over 65%) among students taking driving tests, attributed to inadequate preparation and reliance solely on studying previous tests. The proposed system will encompass:

- An online platform for accessing practice exams.
- A reservation system for scheduling on-the-road training with instructors.
- A management interface for tracking and organizing student progress, instructor schedules, and vehicle availability.
- Integration with DriverPass's internal processes and DMV requirements to ensure updated and relevant training material.
- The system will be web-based, accessible both online and offline, and should offer different access levels for management, IT, instructors, and customers.

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?

To successfully meet DriverPass's needs, the system should:

- Enable students to register for, access, and complete online practice exams.
- Allow students and DriverPass staff to schedule, modify, and cancel on-the-road training sessions.
- Incorporate a flexible package system for different training options.
- Provide different access rights and functionalities for various user roles including administrators, IT personnel, instructors, and customers.
- Include a tracking and reporting feature for activity logs, lesson details, and customer progress.
- Ensure data security and integrity, particularly for customer personal and payment information.
- Be adaptable for future expansions, such as adding or modifying training packages or adapting to DMV policy changes.
- Maintain high performance and availability, with a user-friendly interface accessible on various devices and platforms.

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 run in a secure, standalone hardware environment typical for ATM systems.
- Transactions should be processed rapidly, ideally within a few seconds, to ensure customer satisfaction.
- Regular system updates are necessary for security and functionality, recommended to be performed monthly during off-peak hours to minimize disruption.

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 a secure and stable operating system, such as a customized version of Linux, known for its robustness in ATM environments.
- The backend infrastructure requires a secure and reliable database system, capable of handling concurrent transactions and maintaining data integrity. This could be an SQL-based system known for its stability and security features.
- Network infrastructure must support encrypted data transmission to safeguard customer information during transactions.

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 (customers, administrators, technicians) will be distinguished based on the mode of interaction: customers through their ATM cards and PINs, and administrators/technicians through secure internal protocols.
- User input for PINs and transaction amounts should be case-insensitive and have error-checking mechanisms to reduce input errors.
- The system should alert the admin in cases of repeated failed login attempts, potential security breaches, or hardware malfunctions, ensuring prompt response to any issues.

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 should be designed with a modular architecture, allowing for easy updates and additions of new features or services without major overhauls.
- It must be able to adapt to changing banking regulations and standards, with the ability to implement updates in compliance with new policies.
- The system should support scalability, capable of handling an increasing number of transactions as DriverPass expands its services or customer base.
- Interface adaptability is key for future integration with different types of banking software or third-party services.

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?

- Authentication for user transactions will solely rely on ATM card validation and PIN verification. This two-step process ensures a secure yet straightforward method for user identification.
- The system must encrypt the PIN entry and ensure secure transmission to the backend for verification, protecting against interception or unauthorized access.
- Implement real-time monitoring for unusual transaction patterns or multiple failed login attempts, triggering alerts for potential fraudulent activities.
- Security measures should include an automatic lockout feature after a predefined number of consecutive incorrect PIN entries to prevent brute force attacks.
- Regular updates to the system's security protocols are essential to safeguard against evolving threats and vulnerabilities, maintaining alignment with industry standards and best practices.
- The ATM interface and backend systems should be fortified with firewall protections and intrusion detection systems to prevent unauthorized access.
- Conduct periodic security audits to assess the effectiveness of the security measures and make necessary adjustments to enhance system protection.

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 validate user credentials, specifically ATM card validation and PIN verification, during login.
- The system shall allow users to view their account balance upon successful login.
- The system shall enable users to withdraw cash in predefined denominations.
- The system shall allow users to deposit cash and checks.
- The system shall provide users with a transaction receipt option after each operation.
- The system shall update the user's account balance in real-time post any transaction.
- The system shall provide an option for users to change their PIN.
- The system shall log out users automatically after a period of inactivity.
- The system shall display error messages for failed transactions or incorrect inputs.

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 interface shall be user-friendly, with clear instructions and prompts displayed on the ATM screen.
- Different users, primarily customers, will interact with the interface via a touchscreen or physical buttons on the ATM.
- The interface shall display options for balance inquiry, cash withdrawal, deposit, PIN change, and receipt printing.
- The interface shall provide visual and auditory feedback for successful or failed transactions.
- For visually impaired users, the interface shall include accessibility features such as voice guidance or Braille keypads.

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's assumed that users are familiar with basic ATM operations such as card insertion, PIN entry, and touch/button interface usage.
- The design assumes a stable and continuous power supply for ATMs, as well as regular maintenance for hardware and software components.
- It is assumed that users have secure, functional ATM cards and their corresponding PINs.
- The system design assumes a secure network connection for real-time transaction processing.

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 is limited to transactions permitted by the user's bank, such as withdrawal limits or types of transactions allowed.
- Hardware limitations include the dependency on the ATM's cash dispenser capacity and the reliability of its card reader and keypad.
- The design is constrained by budget allocations, particularly in implementing advanced security features or high-end hardware components.
- Time constraints may limit the extent of user interface customization and additional features like multilingual support.
- The system’s performance is dependent on the backend banking systems and network infrastructure, over which there might be limited control.

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.