Southern New Hampshire University

CS 255 Business Requirements Document Template

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 was for the client, Liam owner of DriverPass, to develop a system that would allow students to be able to take online classes and practice tests for the driving test at the DMV.

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 systems to be able to do a multitude of things. First of all, it should be able to be accessed anywhere, online as well offline. Also, the system needs to run off the cloud. The DriverPass team also needs to have different levels of rights and roles. Ian, the IT officer, for example would need full access over all accounts. The system should also need to be able to print an activity report.

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?
- Once this system is completed, the customers through the system should be able -
  - Make reservations (each lesson block being 2 hours long)
  - Identify the driver who is scheduled with the customer
  - Choose a lesson package
  - Take online tests, which also saves their online test progress to the administration

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 needs to be able to run off the web, preferably over the cloud. This is because the client does not want to deal with backup and security. They instead want their focus to be on running the business with minimal technical problems. The system should be fast enough where the customers do not feel any hinderance while browsing the website and more importantly while taking any online tests. Lastly, the system should be updated as often as there are any bug fixes to make the website as smooth as possible.

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 able to run in at least both Windows and Mac OS as they are the two most used platforms. Lastly, the back end would utilize the cloud computing as their database to support the system.

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?

- Users will have to input their first and last name, as well as their email address when signing up. If two people have the same names somehow, their email address will distinguish between them, which the administration will have access to. The only problem that should occur is if someone tries to sign up with an email address that was already signed up with during the account creation process.

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?

- Making changes to the user, such as add/remove/modify/ should be possible without needing to change the code
- The system will adapt to platform updates by updating the codebase
- The type of access that the IT admins need would be full access to all accounts, which include both user accounts and employee accounts.

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?

- Users will log in with their email address and password
- To secure the connection between the client and the server, using HTTP would be a viable way to do it
- A brute force hacking attempt should alert the IT admins about which user account / employee account is being brute forced. This would be detected by having a certain number of login attempts, such as 5 tries before the admins are notified and the account is locked until recovered by email address. The IT admins would have set up an automatic email that would alert the owner of the account of the failed login attempts through email, which they can recover by clicking a link provided in the email.
- If a user forgot their password, they can click a link which would make the system send the user an email which would let them reset their password.

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 be able to validate user credentials when logging in
- The system shall provide a detailed user interface for ease of browsing on the system
- The system shall be able to provide 3 packages of lessons for customers to choose from
- The system shall be updated according to any DMV changes
- The system shall have online tests for customers to be able to take
- The system shall be available online, and some features offline
- The system shall be able to make reservations
- The system shall be able to make an activity report for the administrators

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 needs of the interface are for the customer to be able to navigate the system in an effective manner.
- The different users for this interface are the admins, and the customers themselves.
- The admins should be able to see and navigate the administration section of the system, while the customer should be able to navigate the rest of the system, which does not include the administration section.
- The user should be able to interact with the interface in order to make reservations, choose a driving package, and take online tests.

Assumptions

3

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 am assuming that users are using either Windows or Mac OS. Thus, the system might look different on a Linux.
- This is partly because there is no budget listed so being efficient with not having to update the system on Linux would save some money.

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?

- Time and budget are a limitation, because of this the system will only be updated for Windows and Mac OS. This is because they are the two most used OS, which makes it efficient for both our time and budget.

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.

Gantt chart
Joshua Asari | August 7, 2022

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12
Collect Requirements	Jan 22 - Feb 4											
Create Use Case Diagrams			Feb 11 - Feb 18									
Build Activity Diagrams for Each Use Case			Feb 15 - Mar 9									
Research User Interface Designs					Feb 27 - Mar 7							
Build Class Diagram					Mar 1 - Mar 9							
Get Customer Approval						Mar 10 - Mar 11						
Build Interface						Mar 12 - Mar 24						
Link DB to Interface							Mar 24 - Apr 3					
Build Business Logic								Apr 5 - Apr 27				
Test System											Apr 27 - May 7	
Deliver System												May 8 - May 9
Sign-off Meeting												May 9 - May 10