Southern New Hampshire University

CS 255 Business Requirements Document Template

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 is to provide better driver training.
- The client is DriverPass, owner is Liam and IT officer is Ian.

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?
- The system should:
  - allow the user to take online classes
  - allow the user to take practice tests
  - provide the user with a way to make reservations for on-the-road training

    - user information collected would be:
      - first name
      - last name
      - address
      - phone number
      - state
      - credit card number
      - expiration date
      - security code
      - pickup/drop-off location

    - day and time should be collected online through their account

    - they can also call and schedule an appointment

    - the driver and car taking the reservation should also be recorded

    - only 10 cars are available

    - user can choose from 3 packages (these should be editable and disablible in the future):
      - 6 hours in a car with a trainer
      - 8 hours in a car with a trainer and an in-person lesson where DMV rules and policies are explained
      - 12 hours in a car with a trainer, an in-person lesson, and access to online class with all content and material which includes practice tests
      - **only 2 hours of driving at a time**
  - provide owner with access to data from any computer or mobile device
  - provide owner the option to download reports/data
  - provide logins with different rights and roles

- owner should have full access over all accounts with the ability to reset them if someone forgets their password or block an account's access
- owner should be clearly notified when a reservation is made, who made it, who canceled it, who modified it
- it officer should have access for maintaining the system, modifying it, etc
- secretary should have access to on-the-road training schedule
- users can make, cancel, and modify appointments
- users can reset their password if forgotten
  - an activity report should be tracked and be made printable at any time by the owner
  - a notification from the DMV whenever an update occurs with new rules, policies, or sample questions
  - run off the web, over the cloud
  - provide a table where driver notes are, form for student information
  - provide a contact us page

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?

- The system overall should:
  - provide the user with a way to make reservations for on-the-road training
  - provide the user with acces to online classes
  - provide the user the ability to leave notes for the driver
  - provide a way to enter any special needs
  - allow a user to upload a photo and see a photo of their driver
  - allow a user to login and have specific rights to the system
- The system will need to:
  - have a cloud database to hold user information, driver information, test progress, driver notes, the drive training schedule, practice tests, activity report
  - have authentication implemented that allows for different roles
  - a connection to specific DMV systems
  - follow the sketch design

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 should seamlessly transition between pages such as exams and online scheduling with minimal downtime.
- The system should be updated as DMV records are updated, this should occur without a lengthy amount of maintenance time in order to allow students access when they need to.

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?
- It should be built to be compatible with any computer, tablet, or mobile device.
- It should also provide browser compatibility: Safari, Firefox, etc.

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?
- Authentication and user roles will be implemented to allow different users to have different permissions. (Owner, IT Officer, Secretary, user)
- Only a user profile can be created from the public page. Other roles will be given on the backend if necessary.

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 to the user can be done through the user profile.
- Behind the scenes, the information is sent up to the database, where it is stored, and is edited with the user's input.
- IT officer should have an IT Officer permission in order to access and modify code.

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 a password will be required for a user to login.
- Input validation will be implemented anywhere a user input is allowed to prevent SQL injections which is a common security threat to the database.
- If a user forgets their password, they can reset it from a link in their email.

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 when logging in.
- The system shall collect and store information about the user's reservation.
- The system shall facilitate uploading and viewing of photos, allowing users to see a photo of their assigned driver.
- The system shall have a clour database implemented to store user information, driver details, test progress, training schedules, practice tests, and activity reports.
- The system shall comply with regulations regarding data security, privacy, and user confidentiality.
- The system shall provide a "Contact Us" page for user inquiries and support requests.

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.)?

- Regular users:
    o can register for an account
    o can login
    o can make, modify, or cancel reservations for on the road training
    o can access online classes and practice tests
    o can leave notes for the driver
    o can specify any special needs
    o can upload and view photos of their assigned driver
- Owner (Liam):
    o can access and download reports/data
    o can manage user accounts (reset passwords, block/unblock accounts)
    o can receive notifications about reservation activities
- IT Officer (Ian):
    o can access the system for maintenance purposes
    o can make modifications to the system infrastructure or functionality
- Secretary:
    o can access on the road training schedules

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 user should:
  - have familiarity with technology
  - have internet access
  - have a device (computer, tablet, mobile)

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?

- Resource limitations: based on how big this application grows in popularity, having an office of 4 people to provide tech support and for faqs will be difficult.
- Given the amount of features in the requirements, the budget should be adequate.

Gantt Chart