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?

- **Client** is: DriverPass, DMV related company.
- **Purpose**: "to take advantage of a void in the market when it comes to training students for the driving test at their local department of motor vehicles (DMV)". Making money through drivers test preparations while also creating a good tool for people who struggle with their test.
- System should be able to host a website where the user can schedule driving times with an instructor, take practice tests, study, and see their data/progress.

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?

- **Problem**: Because so many people fail their drivers test at the DMV, DriverPass felt there needed to be better training.
- **Solution:** Website is created where the user can take online classes and practice tests. The company will also "provide them with on-the-road training if they wish".
- **Security Component**: Mentions limited access to private data within the employees at the company.
- **Admin Component:** Liam has the right to disable packages, access system data to download reports, have full access to all accounts, tracking history, customize packages in the future
- **Individual User Component:** Users must be able to login to their respective accounts and be able to alter their password if forgotten.
- **Cloud Component:** System should run off the web, with cloud being the preferred server and data storage method**.** We do NOT want to be troubled with cybersecurity and backups.

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?

- **User Features:**
  - Able to register and create personal account online
  - Able to create, move, or cancel driving reservations for driving lessons over the internet or on call
    - Able to specifically choose from available date and times for lessons
    - Able to include drop off and pick up locations
  - Able to pick from three different lesson packages
  - Able to reset password
- **Employee Features:**
  - Be able to track which users are matched up with a driver, time, and car
  - Able to create, move, or cancel driving reservations for a customer
- **System Features:**
  - Connected to the DMV to keep up to date with new rules, policies, or sample questions.
    - Get notification every time there is an update from them
    - Cloud connection
  - *User interface:*
    - **Online test progress** that shows tests the customer has taken/has yet to be completed and the data/statuses on them
    - **Driver notes** that show any comments their driver has made on their driving, as well as the time of their lesson together so the lesson can be identified by the user and tracked by the system.
    - **User data** shown and updated here

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?

- **Environment:** Should be a web-based system.
- **System Speed:** Should not be more than a two second load time.
  - This is essential for users trying to take exams
- **System Updates**:
  - For *regular maintenanc*e, every 2-3 months should be enough as the system is not extremely advanced. System maintenance should be an hour and a half at most.
  - System is connected to the DMV -> System should be *updated* every time the DMV makes a change, this should be a short update that does not require hard maintenance, but instead a refresh.

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?

- **Platforms:** As it is a web-based system, the system should be able to be compatible with any platform: Windows, Unix, Mac.
    - Should also be able to run on any internet browser including mobile ones.
        - Examples of Some Browsers: Google Chrome, Opera, Safari, Microsoft Edge
- **Back-End Requirements:**
    - *Database:* Necessary for storing user information
        - Account information: (names, username, password, type of plan)
        - Driving Schedule (open times, who has what time)

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?

- **Distinguish Between Users**
    - Email/phone + passwords are used to differentiate each account
    - Passwords are case sensitive, emails are not
- **Inform Admin of Problem**
    - Admin should be informed when an account is temporarily locked (preventing brute force)
        - More information on account locking in “Security” section
    - Admin should be informed per user request if a user has no access to their registered phone or email, and has forgotten their password
    - Admin should be notified if system has any error in code or is down (not for scheduled maintenance).
    - Admin should be notified if scheduled maintenance taking longer than scheduled

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 user information will be done in the database (backend)
- Platform updates will be dealt with by the system by notifying developers and allowing them to modify
- IT officer needs to be able to maintain the system and modify it
    - Should have access to the database and the admin page
    - Should be blocked if let go/quit

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?

- **Sign Up:**

    - Email AND Phone required
        - Send confirmation number to the one of their choosing
    - Password required
        - Password should have at least one uppercase, at least one lowercase, at least one number, at least one special character, and at least 8 characters.
- **Sign In:**
    - *Forgot Password*
        - Sends link to reset password to phone or email
    - *Forgot/Lost Access to Email*
        - Sends reset email link to phone
    - *2FA*
        - For every sign in on a new device, send confirmation number to either the user's phone or email.
        - This way even if there is a successful brute force hacking attempt, they can't break in.
    - Account Lock
        - If wrong password is entered ten times, then lock the account
            - This account cannot be accessed for ten minutes
        - Send a warning email about the attempted breach, along with a link to reset password if necessary.
- General Security:
    - Because the site is connected to the Cloud, they can manage data security in the servers

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 the user's password and username when they log in.
- The system shall allow users to reset their password if necessary
- The system shall do 2FA when the user signs in from an unknown device/location
- The system shall allow users to sign up for driving lessons
- The system shall allow users to buy a driving lesson practice
    - The system shall validate the user's identity before the purchase is made using password/contact verification.
- The system shall track schedule planning to relay it back to the admin

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.)?

- **User Interface:**
    - *Registration Page if new user*
    - *Home Page*
        - Subjects separated into different boxes on the home page as per the sketch

        - Online Test Progress in top left box
        - Driver Notes section underneath Online Test Progress
        - Student Photo and Driver Photo in bottom right corner next to each other
        - Special Needs and Requirements above the student/driver photos
        - Account Information in the top right corner above Special Needs.
    - *Practice Exam Page*
    - *Scheduling Page*
    - *Account Information Page*
    - *Payment Plan Page*

- **Admin Interface:**
    - Should be able to see user information
        - This is in case the user needs to change their information (password, etc.)
    - Schedule Tracking
        - See the driving schedule and when its updated and by who
        - Printable activity reports
        - Be able to add, remove, and modify appointments

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 all users have access to the internet/have internet connection for the duration of their usage.
- It is assumed that students will want to learn how to drive in accordance with the guidelines/rules of the DMV.
- It is assumed that the website will always be up and running besides regular maintenance schedule
- It is assumed that the website will be safe to use, and security will be maintained.
- It is assumed that the system will be able to connect to the DMV and automatically update when their rules are changed.

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?

- Only have access to 10 cars
- No mobile app just a web page
- Only 15 weeks (about 3 and a half months) to complete the project
- Need internet connection to be accessed
- No set budget which leaves room for uncertainty

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.