Jordan Barnes

Prof. Eppenger

CS-255

10/30/2020

# CS 255 Business Requirements Document

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?*

Liam (the Owner of Driver Pass) is the client. He is attempting to build a platform for soon-to-be or current drivers that are taking their driving exam. In essence, he wants to build a system capable of doing the following: online and offline viewing that provides his customers with practice exams and classes, record class and test progress, allow customers to register with proper credentials and add their pickup and drop off address, and be able to make appointments, modify, or cancel appointments as well. He is attempting to assist new drivers with the ability to practice more before taking their initial exam. He expressed that he feels there is a gap in the driver’s education market that doesn’t allow for this demographic to properly prepare for their practical or written exam. In essence they would like their system to help aid and fix the problem of people failing the driving test.

### 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?*

* Driver Pass wants to display tests and classes to their customers. There needs to be a full backend and front end built for this project. Their front end will need to be able to efficiently query data from the client server with form submissions. They will need a user repository, instructor repository, class repository, etc. It is recommended that the back end be fitted with an appropriate relational database. I personally recommend that the service be built with an external framework like spring boot. Several physical components will be necessary to support the backend service as well including but not limited to a high-level managed switch, network access control device, potential network access sharing drive, etc. They will also need to hire an efficient and capable sustainment team to keep up with bug fixes and updates.

In essence they would like their system to help aid and fix the problem of people failing the driving test. They plan to accomplish this by providing online practice tests and materials along with practical testing in the form of pick-up drop-off driving practice with an instructor.

The system should allow you to access data through an internet connection, along with an offline read only mode. The system should track all reservations, cancelations, modifications, and offer three different packages to choose from for on-the-road training.

### 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?*

* By the time this product is finished, it should be a fully functional website that facilitates the booking, modification, and cancellation of on-road driver training. Online practice tests and classes should also be available to customers. It is also recommended that certain employees only be allowed to access the system to make necessary changes or to further improve it. We can visualize the system by using object models, process models, and UML diagrams. It will be necessary to select an operating system and language, as well as to deploy a client server, to create the website. Using a functional roadmap will give the software team and the company the ability to accurately map their progress and what steps they need to take.

## Requirements

### Nonfunctional Requirements

*In this section, you will detail the different nonfunctional requirements for the Driver Pass 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?*

* For Driver Pass the system should be a web-based platform. It should be as responsive and other production level web platforms. This can be determined through testing if required. There should be reasonable access and connectivity. Load times of no more than 2-3 seconds for users with reasonability web connections. The system should support automatic and manual monthly or bi-monthly updates depending on security requirements. These updates should not have anything more than 2.5 hours of downtime with a minimum of 1 week warning to users.

#### 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 front end of the application should be compatible with all major browsing services (chrome, firefox, internet explorer, etc) and equivalent mobile browsers (safari, search, android browser, etc). The form submission protocol and HSTS protocols should also be adaptable for each browser. HTML/JSP/Thymeleaf (Whatever framework that is used) templates should automatically update div ratios for mobile users and/or have separate UI for mobile users. The system requires a minimum of one database with multiple tables containing user (student, instructor, administrator, etc) data, system logging information, and class object data. It is also recommended that the back end use an external logging tool to maintain error logs and database backups.

#### 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?*

* Most standard web front end’s use session cookies to identify individual users with their principal names and authorization tokens. Passwords should be case sensitive; all other inputs should not be case sensitive. All fatal errors should be immediately addressed and reported by a automatic error logging system to the sustainment team. Otherwise, warnings and non-fatal errors/vulnerabilities should be aggregated and reported daily.

#### 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?*

* Updating the database by modifying, adding, or removing users should be implemented in a way that can be performed at runtime. There should not be any changes to the backend code to complete those changes. As mentioned in the above section (accuracy and precision), all fatal errors should be reported immediately to the sustainment and development team. Otherwise, the sustainment team should continue to make updates in adherence with any platform dependent software changes or updates to avoid fatal incompatibilities. The IT admin should have full access to the database and the client server. He should not have access to any sensitive PII like user social security numbers. These should be encrypted and stored safely to maintain data integrity.

#### 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 are required to input their correct username/email and correct password to log in. There should be an option 2 factor authentication method implemented. 2FA cand be done through SMS or email on preference. Accounts should be locked after a certain number of bad log in attempts (Usually 5) to avoid a brute force attempt. Forgotten or locked accounts should have the ability to request an email for a password reset.

### 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 accessible through mobile and desktop platforms
* The system shall update administrative staff when DMV updates occur.
* The system shall validate the user’s unique username and password when logging in.
* The system shall send a verification SMS or email to the user’s registered phone number or registered email when 2 factor authentication is enabled.
* The system shall lock user accounts after five consecutive failed logins.
* The system shall send email with a temporary password in the event of account lock or forgotten password.
* The system shall update user information on the backend in response to user or admin commands.
* The system shall be able to add/modify/remove user accounts
* The system shall have an easy-to-understand UI and navigation interface
* The system shall be able to be used offline and online for the administrative staff
* The system shall track available appointment times and user appointments.
* The system shall schedule user appointments in response to user or admin commands.
* The system shall maintain a numbered list of updates as they occur

### 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 UI (user interface) will be web-based and will need to be modular. This means the interface should be able to adapt to mobile browsers as well as desktop. Users will be accessing the frontend through a web browser. Each user should be able to update their personal data including but not limited to contact details, passwords, physical addresses, etc. Administrators should be able to add, modify, and remove other non-administrative users and make changes to system level data like DMV requirement changes, user schedules, class information, etc.

### 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?*

* There are several assumptions being made in the above requirements and specifications. These include but are not limited to: Internet connectivity, the user has an email address, the DMV rules can be tracked via an external API/interface, the development team has experience with database and web development, and that our customers can afford the price Driver Pass is setting.

### 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 most obvious limitation is that we don’t have any control on how often the DMV changes policy. We need to design the web interface to be cross-platform and usable on all widely used web browsers for mobile and desktop. This makes it difficult to maintain updates for each platform. The budget will fluctuate depending on the experience level required to build the product, which is a limitation. This should take a minimum of 16 weeks to complete.

### 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.*

