# Matthew Oliphant

October 3, 2021

# CS 255 Business Requirements Document

## 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 build DriverPass a system that will meet their business requirements. DriverPass wants their system to be able to help clients improve their driving skills.
* The client is DriverPass, which is owned by Liam. Users include the owner, IT, admin, and customers.
* DriverPass wants their system to be able to take input in the form of customer contact and billing data, make and confirm appointments for driving lessons, and communicate between users. The purpose is help reduce driving test failures.

### 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?*

* A system overview is described above, but more details are provided below:
* Using customer data, the system needs to be able to create user profiles and allow users to make appointments by booking the package that they desire. Profiles need to store customer data and all tests/ past activity related to the user.
* System tracking for customer interactions/ reservations; printable report of issues for Liam.
* Connect with DMV to update site information
* Provide customers with practice tests and other learning material.
* Interface for site should be laid out to Liam’s design and presented for acceptance testing.
* The problem DriverPass wants to fix is related to the number of drivers who fail their driving tests. They want drivers to have the option to take their classes in preparation to pass their driving test on their first attempt.
* Additional components would include operating the site on a cloud, system reporting, security, scalability, usability and other functional and non-functional requirements.

### 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 should allow customers to create a profile and book appointments according to three available packages. The user should be allowed to choose a driver, but must use that driver throughout their interactions with DriverPass.
* Customers should be able to schedule two hour appointments with their driver based on the driver and customer availability.
* System should alert users of appointments, take payment for appointments, and keep users up to date with appointment details. System reporting should alert admin users of any issues.
* Measureable tasks needed to achieve this system design can be described with use cases. Profiles and reservations must be successfully created, alerts need to be sent out, and reporting needs to provide useful data.
* Communication between customers and staff.
* Packages need to be enabled/ disabled.

## 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 run on a cloud, but will be web-based. DriverPass is starting their business so they don’t require an app yet. As the business grows and has the ability to scale, an app might be considered by Liam.
* Liam wanted the system to work offline, but Sam convinced him that it wouldn’t work properly (duplicate entries).
* The system should perform quickly, which implies the code needs to be efficient. Checking for updates from the DMV should occur every couple of minutes and the system should be updated as often as possible. The cloud will help with this and usually has a policy regarding updating the system.

#### 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?*

* I’m assuming the system should be able to run on multiple platforms including Windows, Mac, Linux, and mobile platforms.
* The back end does require tools to allow for customer service and account fixes.
* Databases will be necessary to store customer information and possibly DMV data. The cloud should maintain these databases so that the site can be accessed from most popular browsers, while keeping the technical requirements for DriverPass to a minimum.

#### 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 be categorized as admin and customers, and each require authentication. All customers will have the same level of access.
* Admin will have a higher level of authorization, and can be broken down into three levels: owner, IT, and reception. Owners can view and print reports. IT may be able to as well, but mainly needs access to fix issues and block access if necessary. Reception should have about the same functionality as customers. A fourth admin user could be the drivers.
* Input should not be case sensitive. The input should converted to a capital first letter, and lowercase after that. For instance, a customer should be able to enter first name as “MaTtHEW” and the system should convert it to “Matthew.” Input should still be validated.
* The system should inform admin of a problem when a user forgets their password or otherwise cannot access their account, and should report on cancelled or modified reservations.

#### 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?*

* IT should be able to make changes to the system without changing code (such as resetting passwords).
* The customers should be able to modify reservations (add/remove/modify) as well.
* System should be able to adapt to platform updates since the platform will still maintain the same functionality. Major platform changes may require changing source code.
* IT admin needs access to maintain the system. They need to be able to modify user accounts and disable packages at the request of Liam. IT needs full access over all accounts so they can reset passwords or block access if necessary.

#### 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?*

* I disagree with the method DriverPass is choosing to implement resetting passwords, as users may not want to have to call someone to do that, however this may be more secure.
* To login, a user will need a profile that the system will create. In order to create the profile, users need to enter personal information that includes a username and password.
* The connection or data exchange between client and server can be secured by utilizing a secure socket layer SSL. Data should only be transferred when absolutely necessary and shouldn’t go to anyone other than the system/ admin.
* A brute force hacking attempt should block access to the system from the account that is doing it. One method to do this could be locking the users’ profile if the incorrect password is entered a certain number of times. IT should be able to maintain the system and block access to any profiles.
* I think there should be a one-time password sent to the email account on file of the user profile and or utilize two-step authentication, but DriverPass wants the user to call in to have it reset by IT.

### 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 signing up for a profile.
* The system shall create user profiles that interact with the sites’ interface.
* The system shall validate user credentials when logging in.
* The system shall accept and validate payments from users.
* The system shall provide access to online testing and educational materials.
* The system shall allow customers to purchase driving practice packages (6, 8, or 12 training hours).
* The system shall allow customers to book appointments with a driving instructor/ car based on the availability of the driver/ car and the customer.
* The system shall remove availability of a time slot once a customer books it.
* The system shall alert customers and admin/ staff of updates to the system which includes booking appointments and appointment status.
* The system shall deliver reports to admin.
* The system shall allow user profiles to be modifiable.
* The system shall provide special IT access to the system that allows them to modify user profiles.
* The system shall connect to the DMV site/ database to update testing materials and other rule changes.
* The system shall maintain user data and profiles.

### 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 was designed by the owner, Liam, and needs to incorporate his design.
* Interface shall show online test progress, user info, driver notes, special needs, a driver photo and customer photo.
* Online test progress should show future, current, and previous testing attempts.
* User info will include first/ last name, full address, phone, email, and billing information.
* Driver notes will include lesson time, start/ end hour, and driver comments for the customer.
* Special needs should describe accessibility requirements for the customer.
* The interface should allow the customer to contact DriverPass admin and should allow admin to contact customers.
* Each user listed should have their own access to the interface. The owner might not need access, but will probably want it.
* Admin needs to be able to create profiles for users who call in.
* IT needs to be able to modify profiles and fix any issues, but will not need access to the interface. Admin and staff may interact with the interface when working on profiles, but won’t have a profile themselves.

### 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?*

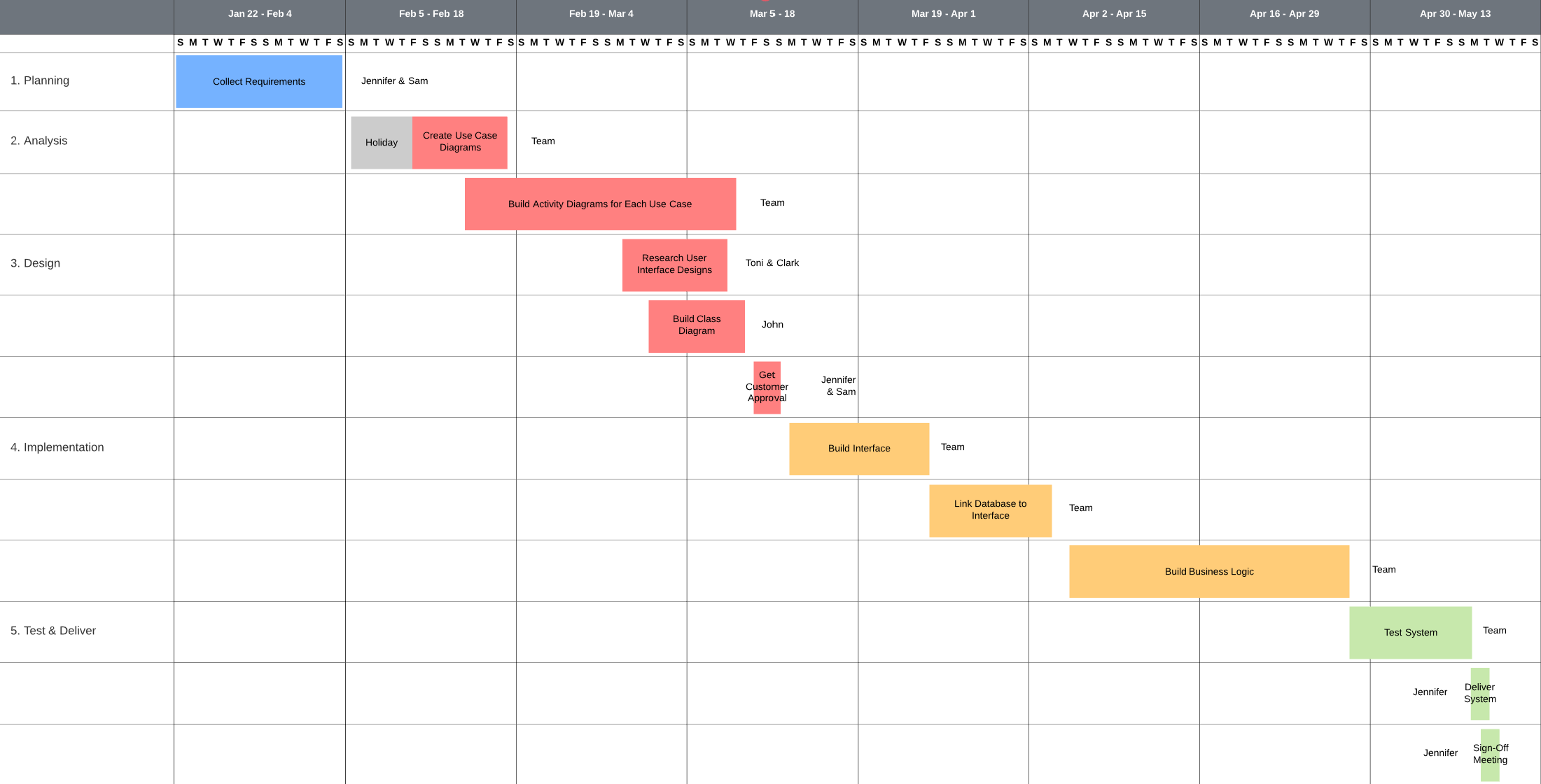
* When to collect credit card info. I think it should be done when the user purchases a package and not during sign-up, unless signing-up requires purchasing access to educational materials.
* I do not think the customer can switch the driver they use, but this is an assumption.
* Users have access to internet and their devices are functional.
* Drivers are not included in the system except when it relates to customers; they don’t have a profile and will not interact with the interface, other than to add notes to the customer profile. These could be noted on paper and the receptionist could enter the notes.

### 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?*

* I see a scalability issue in that admin could receive 100% of user profile requests by phone, which will inevitably overwhelm the receptionist. Cost of additional staff isn’t worth it in this situation (GeeksforGeeks, 2020).
* Resources will be limited since there are only 10 drivers/ cars. This will limit the number of customers the system can handle. Too many users can mean the drivers are booked months in advance and turn some users away.
* Time is limited for the customer as described above. A small staff for DriverPass will mean that IT can only dedicate so much time to fixing issues. Technical requirements take a lot of time, so they want their site to run on a cloud, but this still has time consuming requirements.
* Budget is limited to 10 drivers/ cars and current staff. DriverPass is just starting out, so their resources/ budget is limited to the ability of stakeholders and investors.
* Technology is limited for IT. Using the cloud will remove a lot of technical work, but also requires some. IT will not be able to control the system as much as they would if the server was kept “in-house.”
* Users may have old or outdated technology and may not meet system minimum requirements.
* IT cannot modify packages, but can enable/ disable them.

### Gantt Chart



## References

GeeksforGeeks. (2020a, June 4). *Non-functional Requirements in Software Engineering*. https://www.geeksforgeeks.org/non-functional-requirements-in-software-engineering/