# 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?*

* + Reduce DMV exam failures by giving students:
  + Study content,
  + Practice tests,
  + Access to on the road training,
  + Internal tools to manage operations.

### System Background

*What does DriverPass want the system to do? What is the problem they want to fix? What are the different com*ponents needed for this system?

* + DriverPass is entering a market gap in driving test preparation with online and in person content - the system is the operational portal for registration, scheduling and study.

### Objectives and Goals

*What should this system be able to do when it is completed? What measurable tasks ne*ed to be included in the system design to achieve this?

* + Allow access to operational data,
  + Let students schedule and study driving content,
  + Let staff schedule by phone,
  + Keep materials synchronized with DMV rules,
  + Provide traceable activity reporting.

## 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 s*ystem need to run in? How fast should the system run? How often should the system be updated?

* + The system should handle:
  + Student scheduling,
  + Testing activity,
  + Generate reports on demand,
  + Block offline updates to avoid data issues while still allowing reporting to function for offline review.

#### 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 a web application operating in the cloud with mobile support.
  + The operating system is purely up to choice, as both a Windows Server and a Linux machine could efficiently operate and provide reliable uptime.

#### 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 tied to a key in the database individually, with their access / communication tagged with their cookie (or authentication method of choice) to relate which actions are committed by which user.
  + Password input should always be case sensitive, while username should by pushed to lowercase on the backend and checked for duplication.
  + The system should inform the admin of an issue anytime the server is pinged and no response is returned, or if the any sort of bad actor is using the system incorrectly (such as a brute force attempt).

#### 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?

* + Yes for user accounts, no for system modules.
  + Admin and staff can add / edit / deactivate users through the UI, and students can register with no code change needed. Adding or removing system modules is explicitly out of scope right now, though disabling a package is allowed from the UI.
  + Platform updates are handled by running it as a cloud web app so backup and security are managed by the provider.
  + The IT admin needs full control: reset passwords and block access when someone leaves. Logging is required so they can track who created, modified or canceled reservations are traceable.

#### 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 the*re is a “brute force” hacking attempt? What happens if the user forgets their password?

* + To log in the user needs have an account with creds (username and password). If they forget their password, the system must allow an automatic reset - administrators can also reset or block accounts.
  + The system should enforce encrypted communication between client and server via HTTPS/TLS and hash passwords.
  + If there is a brute force hacking attempt, the account should be temporarily locked after repeated failed login attempts, with administrator notification so someone is aware.

### 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 allow customers to create, view, modify, and cancel reservations for two hour driving lessons online or by phone.
  + The system shall match each reservation to a specific driver, car, date, and time.
  + The system shall provide online access to cIasses and practice tests, including scoring and status tracking.
  + The system shall allow customers to view test progress details including test name, time taken, score, and status.
  + The system shall display driver notes for each lesson, including start time, end time, and comments.
  + The system shall let secretaries input student registration data, including name, address, phone, state, and credit card details.
  + The system shall allow customers to reset forgotten passwords automatically.
  + The system shall notify staff when DMV updates rules, policies, or sample questions.
  + The system shall provide administrators with full account management, including password resets and blocking access.
  + The system shall track and report who created, modified, or canceled reservations, with printable activity reports.
  + The system shall allow administrators to enable or disable training packages for registration.
  + The system shall run as a web application.

### 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 needs to be usable by students, administrators, and IT staff.
  + Students need to register, schedule or cancel lessons, take practice tests, view test progress, see driver notes, and reset passwords.
  + Administrators and IT staff must manage accounts, reset or block access, assign drivers to cars, enable or disable packages, and review logging reports.
  + The interface needs to have forms for entering information like student and payments, tables to display progress and comments, and pages for communication (contact us, contact student, etc.).
  + Users will interact via a browser over the internet, with the system hosted in the cloud.

### 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?*

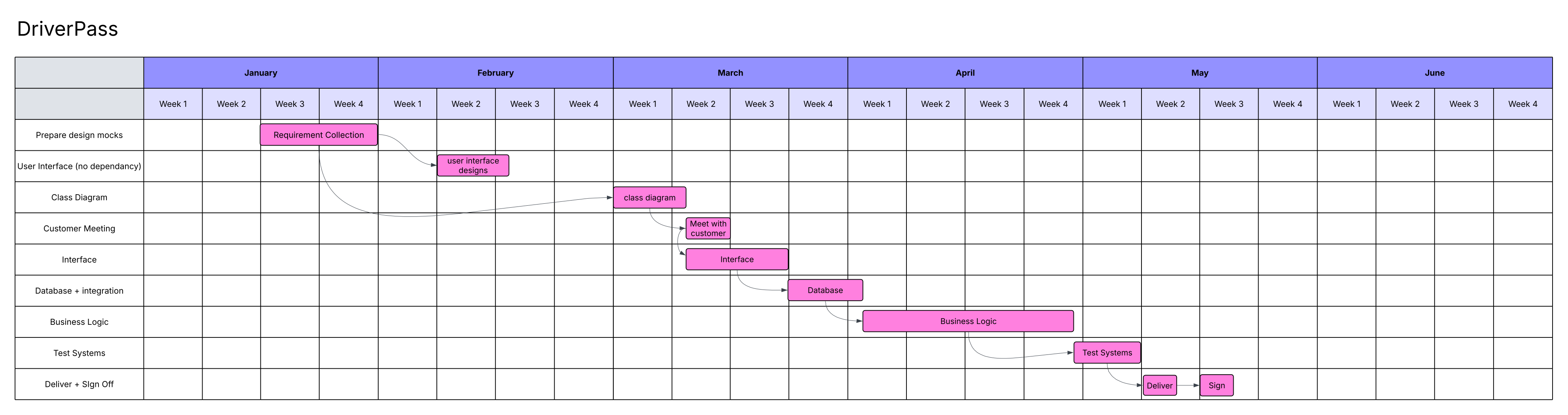
* + Some things werent directly addressed. The interview didn’t cover payment processing beyond collecting credit card fields, so fraud prevention and refunds weren’t defined. Reporting was mentioned, but the scope wasn’t fully specified. Also the design focus of the user interface was only sketched briefly so we do not know their style.  
    The design assumes that all users will have reliable internet access, web browsers, and the knowledge to use the software. It also assumes cloud services will handle backup and security, that DMV updates will arrive in a usable format consistently, and that each car has exactly one driver assigned at a time (most likely will, but it’d be good to confirm).

### 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?*

* + Limitations will always exist, but for this project - on the backend the data cannot be modified offline, only viewed, to avoid redundancy. Packages can only be enabled or disabled, not fully customized, so flexibility is a no. The system depends on reliable internet access, cloud hosting, and DMV updates, which are outside of developer project control.
  + For resources, the project has a fixed timeline and is scheduled from February to May, meaning there is practically no room for delays. The team seems relatively small, with only a few workers splitting design, interface, database, and business logic.

### 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.*