# CS 255 Business Requirements Document - Schmidt

## 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 create a system for DriverPass that makes it easier for people to prepare for their driving tests and pass. DriverPass wants the system to let customers take online classes, complete practice tests, and schedule driving lessons. Customers should be able to make, cancel, or change reservations online through their accounts or by contacting the company by phone or in person. The system also needs to track reservations, show lesson details like time, driver, and car, and provide reports on activities such as who made or changed a booking. DriverPass also wants to stay connected with the DMV to receive updates on new rules, policies, or test questions.

### 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?*

* DriverPass wants the system to fix the problem of too many people failing driving tests by giving them better training options and a more organized way to practice. The system needs several components to work well. It will need a reservation system that tracks which driver, car, and time are connected to each lesson. It will also need a simple web-based interface for customers and employees to use, with features like password resets, role-based access control, and the ability to disable training packages if needed. For example, the secretary should be able to schedule appointments for customers who call, while the IT officer needs full control to manage accounts and security.

### 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?*

* When the system is finished, it should let customers book lessons online or through the office, access online classes and practice tests, and track their progress on tests. For example, customers will be able to see details like the test name, score, and status (not taken, in progress, passed, or failed). Employees will also be able to generate reports that show who made or canceled reservations, helping resolve any issues that come up. The system will connect with the DMV to ensure training materials and tests match the latest rules. It will be cloud-based for secure data storage and easy access from anywhere, making the process simple and reliable for everyone involved

## 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 DriverPass system must be a web-based, cloud-hosted platform to ensure accessibility from any device with an internet connection. It should provide fast response times, ideally loading pages within 2-3 seconds, to ensure a smooth user experience. The system must be updated regularly to incorporate changes in DMV rules and policies, with real-time notifications for any updates. Additionally, it should support high concurrent user loads, especially during peak times, to prevent slowdowns or crashes. Backup and security measures must be automated to minimize downtime and protect sensitive customer data. The system should also be scalable to accommodate future growth and feature additions without compromising performance.

#### 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 DriverPass system should be platform-agnostic, running seamlessly on major operating systems such as Windows, macOS, and Unix-based systems. The back end will require a robust relational database management system like MySQL or PostgreSQL to store and manage user data, driving schedules, and test results. Additionally, the system should integrate with cloud services for scalability and reliability, ensuring smooth operation across different environments.

#### 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?*

* Accuracy and precision are critical for distinguishing between users, with unique identifiers such as usernames and email addresses ensuring no overlap. Input fields, such as passwords, should be case-sensitive to enhance security. The system should notify the admin immediately of any critical issues, such as failed login attempts, data inconsistencies, or system errors, through automated alerts or logs.

#### 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?*

* The system must be adaptable, allowing IT admins to add, remove, or modify user roles and permissions without requiring code changes. It should also be designed to handle platform updates seamlessly, ensuring compatibility with new operating system versions or browser updates. The IT admin will need full access to manage user accounts, reset passwords, and monitor system activity to maintain smooth operations.

#### 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?*

* Security is a top priority, requiring users to log in with a username and password, with optional two-factor authentication for added protection. Data exchange between the client and server should be secured using HTTPS and encryption protocols. In the event of a brute force hacking attempt, the system should automatically lock the account after a set number of failed login attempts and notify the admin. If a user forgets their password, the system should provide a secure, automated password reset process via email verification to ensure only authorized users regain access.

### 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 to ensure secure access. The system shall allow customers to schedule, modify, or cancel driving lessons online or via phone. The system shall provide access to online classes and practice tests that are updated based on DMV requirements. The system shall track and log all user activities, including reservations, cancellations, and modifications, for reporting purposes. The system shall allow IT admins to reset passwords, block accounts, and manage user roles and permissions. The system shall enable customers to view their test progress, including completed tests, scores, and status. The system shall generate and download reports for management, such as activity logs and reservation details. The system shall notify users and admins of updates to DMV rules, policies, or sample questions in real time. The system shall support role-based access control to ensure that users only access features relevant to their roles. The system shall allow the owner to disable or enable specific packages without requiring developer intervention.

### 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 user interface must be intuitive and accessible, catering to different users such as customers, the secretary, IT admins, and the owner. Customers will use the interface to schedule lessons, take online tests, and view progress. The secretary will use it to manage appointments and customer interactions. IT admins will need access to user management tools, while the owner will require reporting and package management features. The interface should be accessible via web browsers on desktops, tablets, and mobile devices to ensure flexibility and convenience for all users. It should include features such as input forms for customer information, progress tracking, and contact options for support.

### 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?*

* Assumptions include that all users will have access to a reliable internet connection and a compatible device to use the system. It is assumed that the DMV will provide timely updates to rules and policies, which the system can integrate seamlessly. Additionally, it is assumed that users will follow the password reset process correctly and that IT admins will have the necessary technical skills to manage the system effectively. The design assumes that the cloud infrastructure will handle backups and security without requiring significant intervention from the DriverPass team.

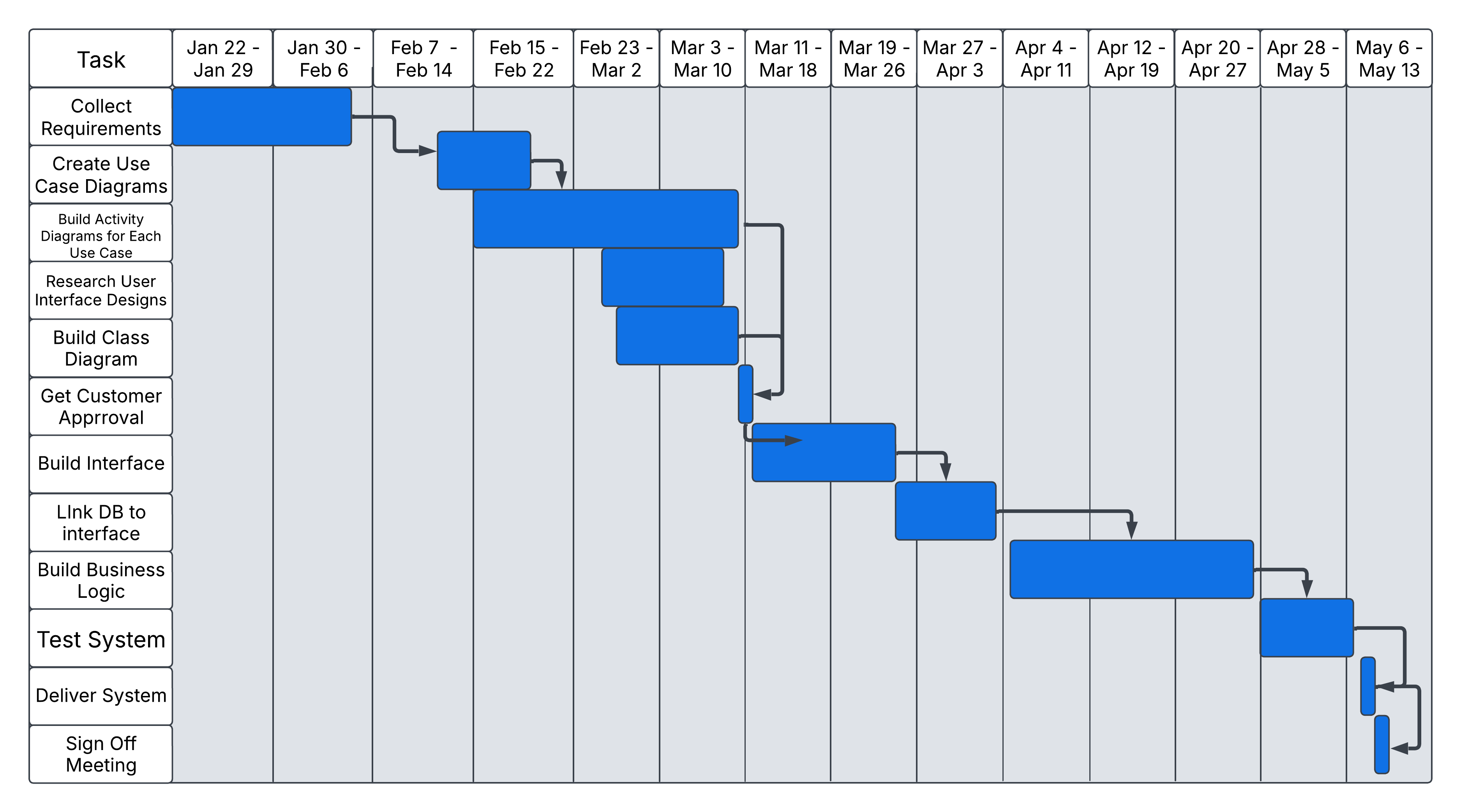
### Limitations

*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 of the system include potential scalability challenges if the user base grows rapidly, which may require additional resources or infrastructure upgrades. The system’s reliance on real-time DMV updates means any delays in receiving updates could impact the accuracy of the training materials. Budget constraints may limit the ability to implement advanced features such as AI-driven personalized learning paths. Time constraints could also affect the thoroughness of testing, potentially leading to undiscovered bugs or usability issues. Additionally, the system’s performance may be limited by the user’s device capabilities or internet speed, which are outside the control of the system design.

### 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.*

**