# CS 255 Business Requirements Document Template

## System Components and Design

### Purpose

* The purpose of this project is to design and develop a system for DriverPass, a client that provides driver training for students who want to pass their driving test. The consulting company aims to provide a high-quality, secure, and user-friendly system that meets the needs of DriverPass and their clients. The system should allow students to access online practice exams and on-the-road training to help them prepare for their driving test. It should also provide driving instructors with access to information about their students and the ability to schedule on-the-road training sessions.

### System Background

* DriverPass has identified a problem in society related to the lack of adequate driver training resulting in many people failing their driving tests at the DMV. To address this issue, they want to offer a solution that includes online classes, practice tests, and on-the-road training sessions, with packages tailored to the specific needs of each student. Our task is to design and develop a system that can handle these needs, allowing students to access online practice exams, make driving lesson reservations, and providing instructors with access to student information and scheduling capabilities. The system should also be able to track student progress and connect to DMV updates to ensure that the training is current and relevant. The components needed for this system include a secure web interface that runs off the cloud, a database to store customer and lesson information, an online reservation system, and a business logic layer to manage user roles and permissions. The overall goal is to provide a high-quality, user-friendly system that can help DriverPass and its clients prepare for driving tests more effectively.

### Objectives and Goals

* When completed, the system for DriverPass should be able to provide a comprehensive and effective driver training program that includes online classes, practice tests, and on-the-road training sessions tailored to the needs of each student. Measurable tasks that need to be included in the system design to achieve this goal include:

1. Design and develop a secure web interface that runs off the cloud.
2. Implement a database to store customer and lesson information.
3. Create an online reservation system that allows students to schedule driving lessons.
4. Develop a business logic layer to manage user roles and permissions.
5. Provide instructors with access to student information and scheduling capabilities.
6. Connect the system to the DMV to receive updates on new rules, policies, and sample questions.
7. Enable students to access online practice exams and track their progress.
8. Offer multiple packages for training sessions and allow for customization of packages.
9. Ensure the system allows students to make appointments, cancel, and modify appointments online if they wish.
10. Provide a way for customers to contact the company through the system.
11. Track student progress, including test results and driving session notes.
12. Allow for easy printing and access to activity reports.

The overall goal is to provide a high-quality, user-friendly system that can help DriverPass and its clients prepare for driving tests more effectively.

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

* The DriverPass system should be a web-based system that can be accessed from any device with an internet connection. It should be compatible with different operating systems and browsers.
* The system should be designed to be fast and responsive, with a maximum response time of a few seconds for user requests. The system should be optimized for speed and should not experience any lag or latency. The system should also be designed to handle many users and transactions simultaneously.
* The system should be updated regularly to ensure that it is up to date with the latest DMV rules, policies, and sample questions. The frequency of updates will depend on the frequency of updates from the DMV. The system should also be updated regularly to address any security vulnerabilities or performance issues.

#### Platform Constraints

* The system will be designed to run on different platforms, including Windows, Unix, and Linux.
* The back end of the system will require a database to support the application and will be designed to store user information, appointment details, training materials, and other relevant data.

#### Accuracy and Precision

* The system should assign unique user IDs or usernames to each user to distinguish between different users. Different levels of access and permissions should also be used to differentiate between different users based on their roles and responsibilities.
* The system should be designed to be case-sensitive for input to make it more difficult for attackers to guess usernames and passwords.
* The system should inform the admin of any problems as soon as they are detected, such as errors in the reservation process or security breaches. Mechanisms should also be in place to monitor for potential problems and alert the admin as needed.

#### Adaptability

* The system should allow for changes to user accounts, including the ability to add, remove, or modify them, without requiring changes to the code.
* The system should be designed to be adaptable to platform updates, with compatibility testing and adjustments made as needed to ensure continued functionality.
* The system should use modular and scalable architecture to ensure that it can adapt to platform updates or changes in the business requirements.
* The IT admin should have full access to the system's back end, including the ability to manage user accounts, update system configurations, and monitor system performance.
* Access control mechanisms should be in place to ensure that only authorized personnel have access to sensitive data or system settings.

#### Security

* The system should require users to provide a unique identifier or username and password to log in.
* The system should use secure authentication methods, such as encryption and multi-factor authentication, to protect the connection and data exchange between the client and the server.
* The system should use SSL encryption to encrypt data in transit to secure the connection or data exchange between the client and the server. Other security measures such as two-factor authentication, firewalls, and intrusion detection systems should also be implemented.
* The system should temporarily lock the user's account after a specified number of failed logins attempts to prevent "brute force" hacking attempts and protect the account from unauthorized access.
* The system should provide an option for password reset if the user forgets their password. The password reset process should be done through secure methods such as email verification or security questions. The system should also have mechanisms in place to prevent unauthorized access to the user's account during the password reset process.

Performance Requirements

• The DriverPass system shall respond to user requests within 3 seconds of receiving the request, with a maximum response time of 5 seconds for any request.

• The system shall be able to handle at least 1000 simultaneous users without any lag or latency issues.

• The system shall be updated within 24 hours of any new DMV rules, policies, or sample questions being released.

Platform Constraints

• The DriverPass system shall be compatible with the latest version of Windows, Unix, and Linux operating systems and the latest versions of popular browsers such as Chrome, Firefox, and Safari.

• The back end of the system shall be designed to use a MySQL database to support the application and store user information, appointment details, training materials, and other relevant data.

Accuracy and Precision

• The system shall assign unique user IDs or usernames to each user to distinguish between different users. Different levels of access and permissions shall also be used to differentiate between different users based on their roles and responsibilities.

• The system shall be designed to be case-sensitive for input to prevent attackers from guessing usernames and passwords.

• The system shall inform the admin of any issues within 30 seconds of detection, such as errors in the reservation process or security breaches. Mechanisms shall also be in place to monitor for potential issues and alert the admin as needed.

Adaptability

• The system shall allow for changes to user accounts, including the ability to add, remove, or modify them, without requiring changes to the code.

• The system shall be tested for compatibility with new platform updates, and any necessary adjustments shall be made to ensure continued functionality.

• The system shall use modular and scalable architecture to ensure that it can adapt to platform updates or changes in the business requirements.

• The IT admin shall have full access to the system's back end, including the ability to manage user accounts, update system configurations, and monitor system performance. Access control mechanisms shall be in place to ensure that only authorized personnel have access to sensitive data or system settings.

Security

• The system shall require users to provide a unique identifier or username and a password with at least 8 characters, including uppercase and lowercase letters, numbers, and symbols, to log in.

• The system shall use encryption and multi-factor authentication to protect the connection and data exchange between the client and the server.

• The system shall use SSL encryption to encrypt data in transit to secure the connection or data exchange between the client and the server. Two-factor authentication, firewalls, and intrusion detection systems shall also be implemented.

• The system shall temporarily lock the user's account after 3 failed logins attempts to prevent "brute force" hacking attempts and protect the account from unauthorized access.

• The system shall provide an option for password reset through email verification or security questions. The system shall also have mechanisms in place to prevent unauthorized access to the user's account during the password reset process.

### Functional Requirements

* The system shall provide online classes and practice tests for customers.
* The system shall be accessible online and offline from any computer or mobile device.
* The system shall allow reports and data to be downloadable, preferably in Excel format for Liam.
* The system shall provide clear tracking of reservations, cancellations, and modifications, including information on who made the changes and when, to facilitate accountability and troubleshooting. It shall also provide an activity report that enables Liam to identify who is responsible for any issues or errors that occur.
* The system shall enable customers to make reservations for two-hour driving lessons by specifying the desired date and time, either online through their account or by calling or visiting the office to schedule an appointment with the secretary.
* The system shall identify the driver assigned to each customer and car for admin.
* The system shall enable users to make, cancel, and modify appointments online at their discretion.
* The system shall accommodate different packages and be customizable.
* The system shall allow disabling of packages.
* The system shall collect customer information, including credit card information and pickup/drop-off locations.
* The system shall allow customers to reset their passwords automatically.
* The system shall be connected to the DMV to stay current with policies and regulations.
* The system shall run off the web and be cloud-based.
* The system shall have backup and security taken care of, with minimal technical problems.
* The system shall be designed to allow for the introduction of new features.
* The system shall display the customer's progress for online tests taken through the app, including the name of the test, the time taken, the score, and the status (which can be "not taken," "in progress," "failed," or "passed").
* The system shall display any comments left by the driver in the driver notes section, along with the times for each lesson.

• The system shall provide online classes and practice tests for customers.

The system must load online classes and practice tests in less than 5 seconds on any device.

• The system shall be accessible online and offline from any computer or mobile device.

• The system shall allow reports and data to be downloadable, preferably in Excel format for Liam.

The system must allow users to download reports and data within 10 seconds.

• The system shall provide clear tracking of reservations, cancellations, and modifications, including information on who made the changes and when, to facilitate accountability and troubleshooting. It shall also provide an activity report that enables Liam to identify who is responsible for any issues or errors that occur.

The system must log reservation, cancellation, and modification activity and make it available for review by admins within 5 seconds.

• The system shall enable customers to make reservations for two-hour driving lessons by specifying the desired date and time, either online through their account or by calling or visiting the office to schedule an appointment with the secretary.

The system must allow customers to schedule appointments for a specific date and time within 15 seconds.

• The system shall identify the driver assigned to each customer and car for admin.

The system must display the assigned driver and car within 5 seconds of accessing the customer's profile.

• The system shall enable users to make, cancel, and modify appointments online at their discretion.

The system must allow users to make, cancel, and modify appointments within 10 seconds.

• The system shall accommodate different packages and be customizable.

The system must allow customization of packages within 5 seconds.

• The system shall allow disabling of packages.

The system must allow admins to disable packages within 5 seconds.

• The system shall collect customer information, including credit card information and pickup/drop-off locations.

The system must allow customers to input and store their personal and financial information within 15 seconds.

• The system shall allow customers to reset their passwords automatically.

The system must allow customers to reset their passwords within 10 seconds.

• The system shall be connected to the DMV to stay current with policies and regulations.

The system must update DMV policies and regulations within 1 hour of their release.

• The system shall run off the web and be cloud-based.

The system must be accessible from any web browser and run on the cloud within 5 seconds.

• The system shall have backup and security taken care of, with minimal technical problems.

The system must have a backup plan and be secure against potential threats, with no more than 1 hour of downtime per month.

• The system shall be designed to allow for the introduction of new features.

The system must allow for the introduction of new features within 1 week of development completion.

• The system shall display the customer's progress for online tests taken through the app, including the name of the test, the time taken, the score, and the status (which can be "not taken," "in progress," "failed," or "passed").

### User Interface

* The interface should display test progress information for customers, including completed tests and those in progress, along with test scores and status (not taken, in progress, failed, or passed).
* The driver notes section of the interface should display lesson times, start, and end hours, driver information, and comments left by the driver.
* The interface should include fields for users to input their personal information, including first name, last name, address, city, state, zip, phone number, and email address. It should also include a section for users to specify any special needs they may have.
* The interface should display driver and student photos for easy identification.
* The different users of this interface include customers, drivers, administrative staff, and Liam. Customers will use the interface to book and manage appointments, view their test progress, and update their personal information. Drivers will use the interface to view their schedules, leave comments, and modify lesson times. Administrative staff will use the interface to manage appointments, monitor test progress, and perform other administrative tasks. Liam will use the interface to monitor the performance of the business, access customer information, manage appointments, and perform other administrative tasks.

Customer User Interface

• The customer interface should allow users to register for courses, view their progress, and manage their appointments, including rescheduling or canceling.

• The interface should display test progress information for customers, including completed tests and those in progress, along with test scores and status (not taken, in progress, failed, or passed).

• The interface should include fields for users to input their personal information, including first name, last name, address, city, state, zip, phone number, and email address. It should also include a section for users to specify any special needs they may have.

• The interface should display driver and student photos for easy identification.

Secretary User Interface

• The secretary interface should be similar to the customer interface but should have additional privileges to supervise all users and perform administrative tasks on behalf of users.

• The interface should allow secretaries to make appointments on behalf of users and assist with the registration process.

• The interface should provide basic reports and views of schedules, and allow secretaries to modify schedules as needed.

Administrator User Interface

• The administrator interface should have access to all records, user information, accounts, and logs, and should be able to track activity across the system.

• The interface should allow administrators to view, edit, and delete records as needed.

• The interface should provide tools for managing user roles and permissions, including adding and removing users and modifying user access levels.

• The interface should provide detailed reports on system performance, usage statistics, and other relevant data.

### Assumptions

* Users have basic knowledge of how to use a web browser and access the internet.
* Users will provide accurate and valid personal and credit card information during the registration process.
* IT personnel have the necessary technical expertise to manage and troubleshoot the system.
* The system will be hosted on a reliable and secure web hosting platform.
* The system's performance and compatibility will be regularly monitored and updated to ensure optimal user experience.

### 

### Limitations

* The system will stop functioning once the number of concurrent users exceeds its capacity, leading to slow performance or system downtime during peak periods.
* The system will stop functioning once the database is corrupted or fails to operate as expected.
* The system will stop functioning once the database is corrupted or fails to operate as expected.

Top of Form

Bottom of Form

### Gantt Chart

