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?*

* Our company has been hired by DriverPass, a company that offers online driver education courses, to create a web-based application that allows users to book driving lessons online and track their progress.
* There is a need for better driving training.
* Many people fail the driving test at the local DMV.
* DriverPass will train students with on-the-road driver training.
* There is an option to enroll in online classes and test practice tests as well.

### 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?*

* The problem that DriverPass wants to fix is the low success rate of people who take driving exams for the first time. Just a quick search shows that around 50% of people fail the driving portion the first time, and 40% fail the written portion.
* The solution that DriverPass wants to offer is an online platform that provides comprehensive and personalized driver education. Their system would allow users to access practice tests, study materials, and schedule in-person lessons with certified instructors. Their goal is to help users prepare for the driving test and become safe and confident drivers.
* The current system, from what I understand, is customers calling in and a secretary taking appointments by hand. A automated system is really needed here if the business is going to be able to expand.
* The system needs to be accessible from any computer or mobile device.
* Keep track of user information.
* Keep track of tests that the user has and hasn’t taken.
* Show notes from the driving instructor.
* Have a way of communicating between DriverPass and the user.
* Keep track of user driving appointments.

### 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?*

* Download reports and information for use offline.
* Have different levels of security: owner, IT officer, secretary, and user.
* Have the customer or secretary make reservations for driving training.
* Identify a driver for every 2-hour block of driving time.
* Track changes made to reservations: who made it, who cancelled it, and who last modified.
* Include several packages, and the ability to add and delete packages in the future, and the ability to disable packages immediately.
* Register from secretary or user. Input user first and last name, address, phone number, state, credit card number, expiration date, security code, and location of pickup and drop off.
* Automatic password reset for user.
* Connect to DMV and receive notification when there is an update to rules, policy, or sample questions.
* Cloud based with backup and security handled.
* Test progress: Show in progress, pass, fail, or not taken. Also test name, time taken, and score.
* Driver notes: Comments of driver and times of lessons.
* A page for communication between user and DriverPass.

## 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 in a web-based environment that supports multiple browsers such as Chrome and Edge and possibly on apps developed specifically designed for mobile devices.
* The system should run fast enough to provide a satisfactory user experience (reasonable web browser load times) and meet the business needs (reasonable load times from the database).
* The system should be updated regularly to ensure security, reliability and functionality, and also when there is a change to our system or the DMV 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?*

* The system should run on Windows, Unix, and MacOS platforms, as well as Android and iOS systems.
* The back end of the system should use a relational database to store and manage the driver and passenger data, such as SQL or Oracle.
* The back end of the system should also support RESTful API to communicate with the front end of the system for security, efficiency, and reliability.

#### 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?*

* To distinguish between different users, the system will use a combination of username and password authentication.
* The input is case-sensitive for the password, but not for the username. The system will automatically convert the username to lowercase before checking its validity.
* The system should inform the admin of a problem when he user enters an incorrect username or password more than three times in a row or the system detects any unauthorized or suspicious activity on the user's account.

#### 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?*

* Anyone should be able to make changes to their own profile except for the security level.
* The secretary should be able to make changes to customers’ profiles if they call.
* The IT admin needs full access to the system, including the database, the web server, and the backend. The IT admin can also backup and restore data and troubleshoot any issues.

#### 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?*

* To log in, the user should provide a valid username and password.
* To secure the connection between the client and the server we should encrypt the data, and use RESTful API.
* If there is a “brute force” hacking attempt, the account should be locked out after a certain number of failed login attempts. The user would then have to reset their password.
* If the user forgets their password, they should be able to request a password reset link that is sent to their registered email address.

### 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:

* Download reports and information for use offline.
* Have different levels of security: owner, IT officer, secretary, and user.
* Allow the customer or secretary to make reservations for driving training.
* Identify a driver for every 2-hour block of driving time.
* Track changes made to reservations: who made it, who cancelled it, and who last modified it.
* Include several packages, and provide the ability to add and delete packages in the future, and the ability to disable packages immediately.
* Register customers from secretary or user accounts. Require user first and last name, address, phone number, state, credit card number, expiration date, security code, and location of pickup and drop off.
* Enable automatic password reset for user accounts.
* Connect to DMV and receive notification when there is an update to rules, policy, or sample questions.
* Be cloud based with backup and security handled.
* Display test progress: Show in progress, pass, fail, or not taken. Also show test name, time taken, and score.
* Store driver notes: Comments of driver and times of lessons.
* Have a page for communication between user and DriverPass.

### 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.)?*

* There are three different interfaces: customer, secretary, and IT admin
* Customer is the basic interface shown in the interview
* Secretary is the same but with a additional window for looking up different customers accounts
* IT admin will have access to all the reports and all information in the system.

### 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?*

* Reliable and secure communication between client and server
* The client and server are compatible.
* The client and server have enough resources.
* The client is secure and has antivirus / firewall protection.

### 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 system might not respond well to a malicious attack, even with the security measures in place.
* Although modular, adding features might be more difficult as the system gets larger.
* As technology advances, the system will need updates for newer operating systems.

### 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.*

*A picture containing diagram, plot, line, number

Description automatically generated*

**UML use case diagram**

A computer screen shot of a diagram

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with medium confidenceUML activity diagram login and authenticate.

UML activity diagram make reservationA picture containing text, screenshot, diagram, font

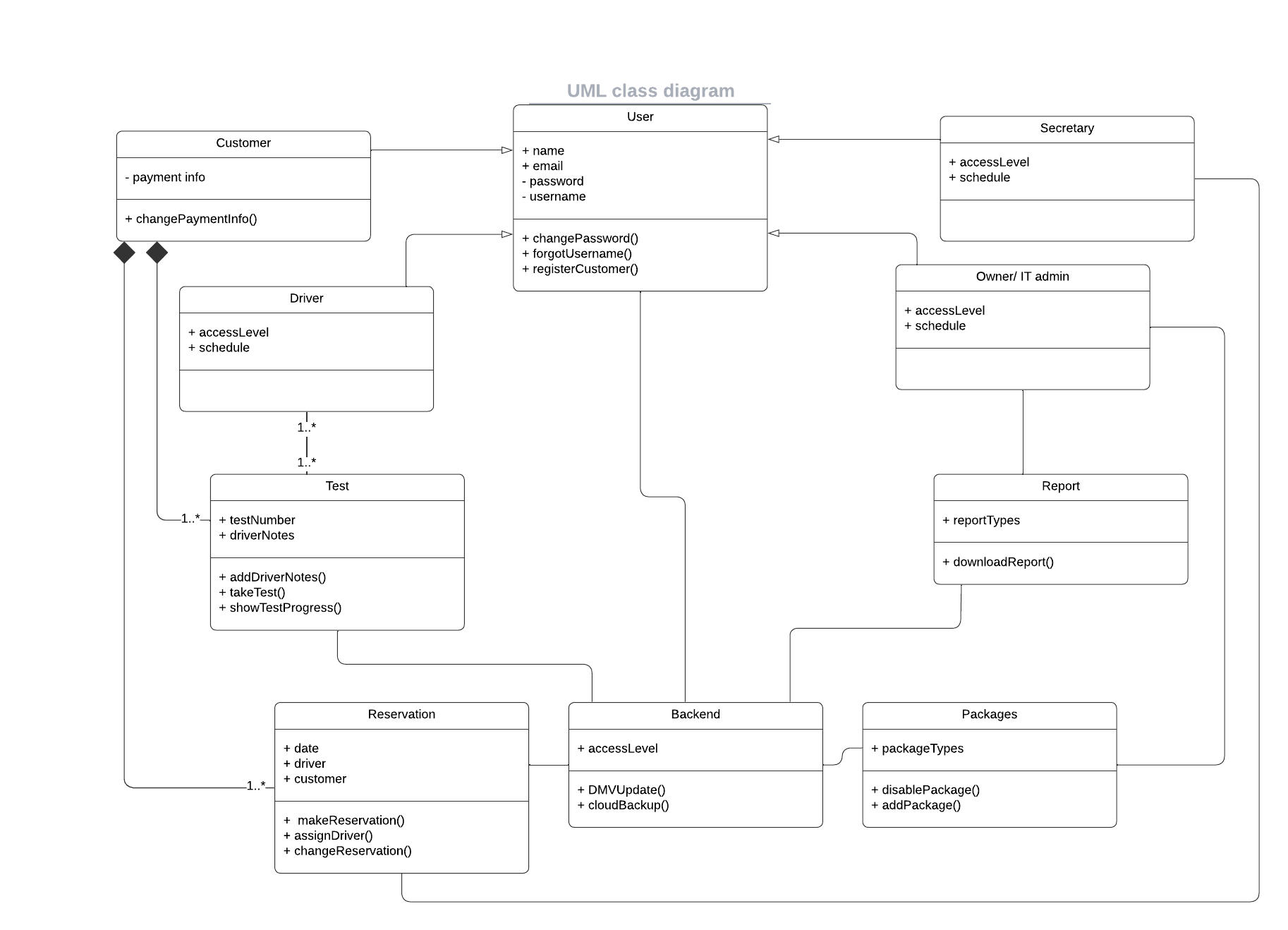
Description automatically generated.

UML sequence diagram login and authenticate

A picture containing text, diagram, parallel, plan

Description automatically generated

Uml class diagram



The technical requirements of our system are as follows:

- Hardware: The system will require a web server, a database server, and a firewall. The web server will host the front end of the system, which will run in a web environment supporting multiple browsers (e.g., Chrome, Edge) and mobile apps. The database server will host the relational database (e.g., SQL, Oracle) that will store and manage driver and passenger data. The firewall will protect the system from unauthorized access and attacks.

- Software: The system will use a web framework (e.g., Django, Flask) to develop the front end and the back end of the system. The front end will provide a user interface for drivers and passengers to access the system's features. The back end will provide the business logic and functionality of the system, such as username and password authentication, data encryption, RESTful API communication, etc. The system will also use an email service (e.g., Gmail, Outlook) to send password reset links to registered users who forget their passwords.

- Tools: The system will use various tools to ensure the quality, security, and performance of the system. For example, the system will use a code editor (e.g., Visual Studio Code, Sublime Text) to write and edit code, a version control system (e.g., Git, SVN) to manage code changes and updates, a testing tool (e.g., Selenium, PyTest) to test the functionality and usability of the system, a debugging tool (e.g., PyCharm, Eclipse) to identify and fix errors and bugs in the code, a code analysis tool (e.g., Pylint, SonarQube) to check the code quality and style, a security tool (e.g., Nmap, OWASP ZAP) to scan and assess the security vulnerabilities of the system, and a performance tool (e.g., JMeter, LoadRunner) to measure and improve the speed and efficiency of the system.

- Infrastructure: The system will run on Windows, Unix, MacOS, Android, and iOS platforms. The system will also use cloud services (e.g., AWS, Azure) to host and deploy the system online. The cloud services will provide scalability, reliability, and availability for the system. The system will also use a backup service (e.g., Dropbox, Google Drive) to backup and restore data in case of data loss or corruption. The system will also use a monitoring service (e.g., Nagios, Splunk) to monitor and report the status and performance of the system.