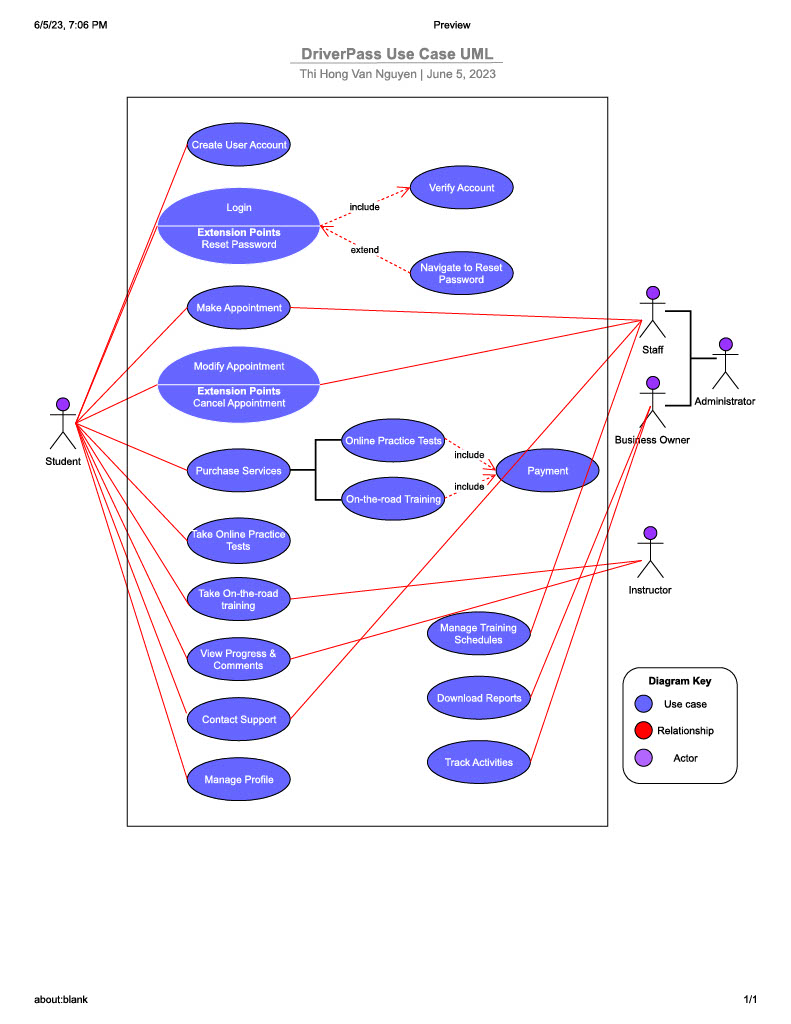
# CS 255 System Design Document

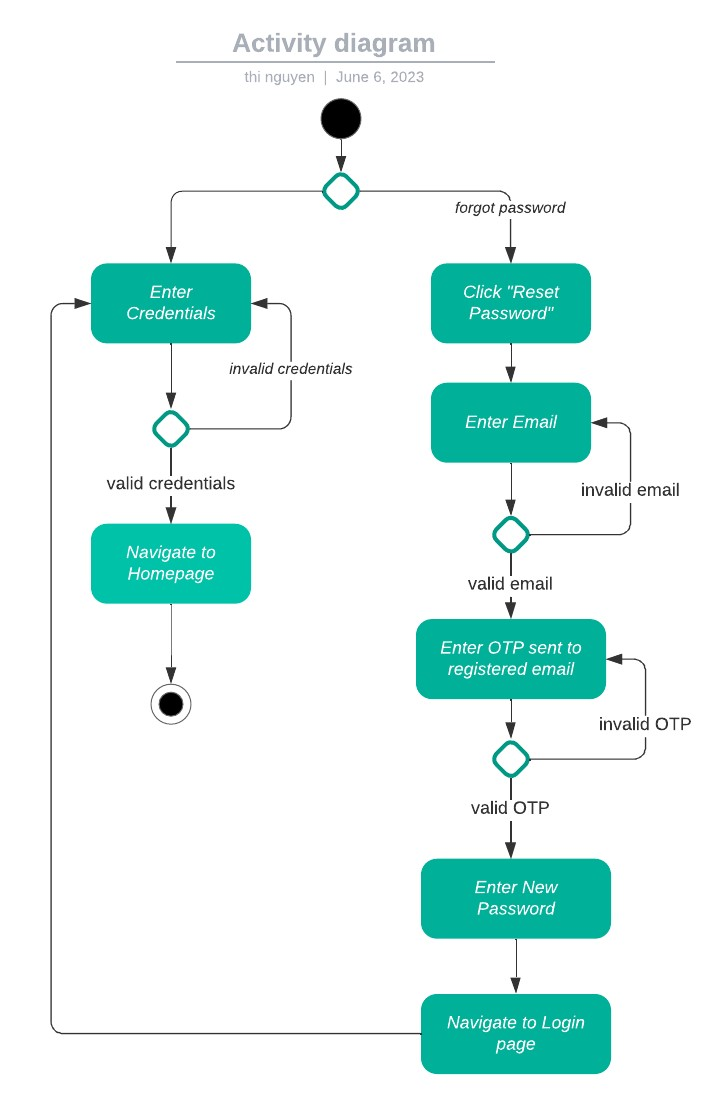
## UML Diagrams

### UML Use Case Diagram

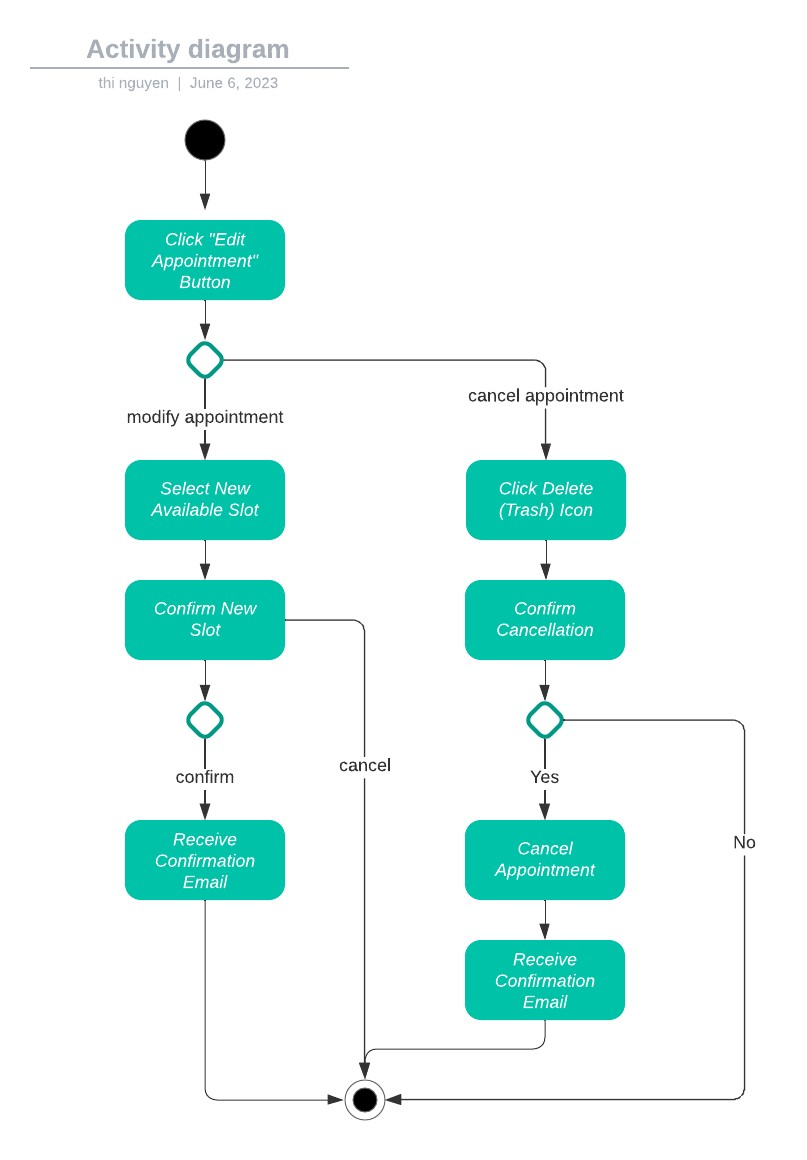


### UML Activity Diagrams

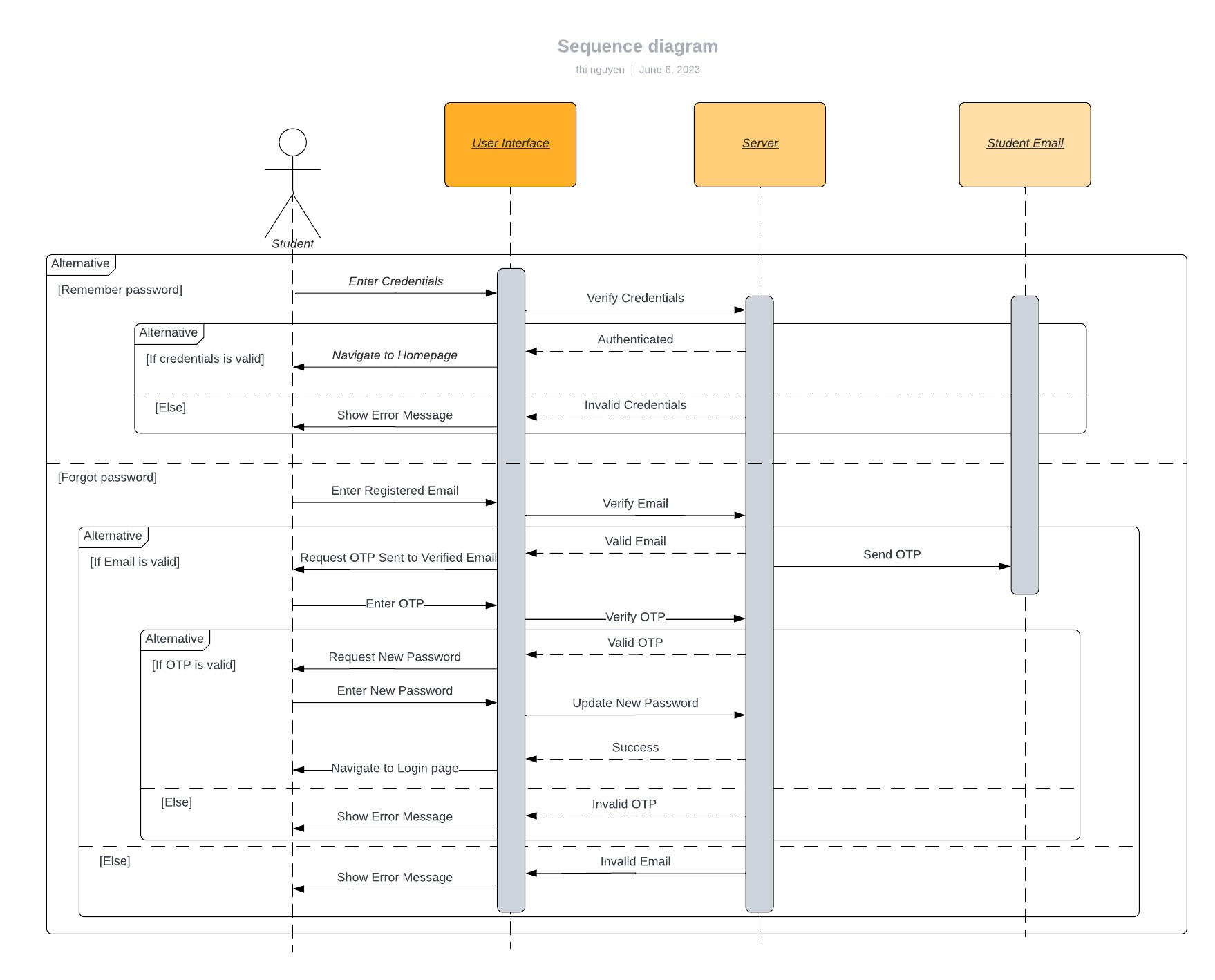
***Use Case “Login”***



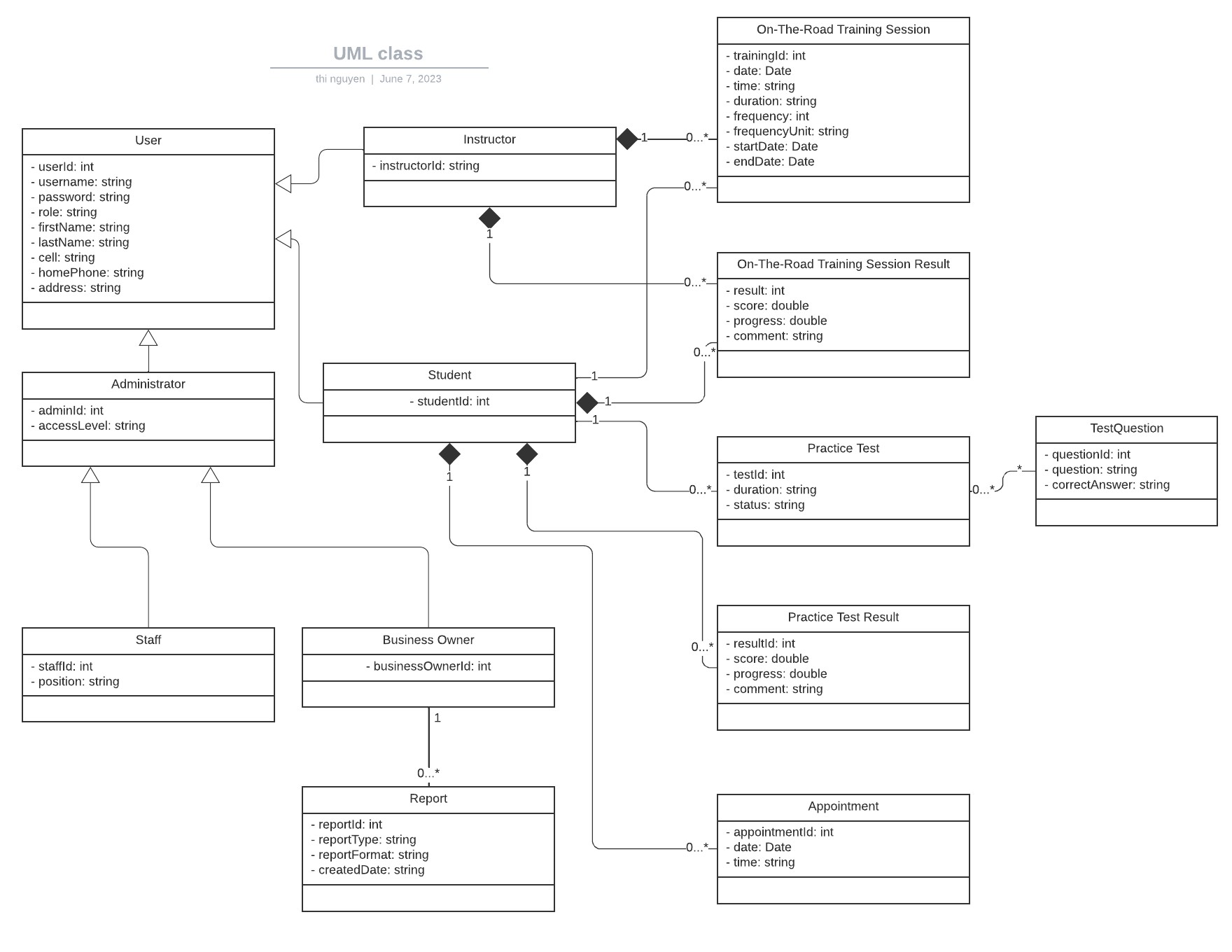
***Use Case “Modify Appointment”***



### UML Sequence Diagram



### UML Class Diagram



## Technical Requirements

**Hardware Requirements**

The DriverPass system design necessitates a robust server infrastructure, which could consist of high-performance servers with sufficient processing power, memory, and storage capacity to handle the application's data storage and processing needs effectively. Examples of server hardware could include enterprise-grade rack servers, blade servers, or cloud-based virtual servers.

**Software Requirements**

The DriverPass system has several software requirements that need to be considered. Firstly, a stable and secure operating system should be selected for the servers to ensure compatibility with the required software components and handle the expected workload. Additionally, a suitable Database Management System should be chosen to efficiently store and manage the system's data, taking into account factors like data volume, performance, and scalability. A web server software is necessary to handle incoming HTTP requests and serve the web-based components of the system. Depending on the chosen programming language and technology stack, relevant development frameworks and libraries should be utilized to facilitate development and provide necessary functionalities. Finally, client applications such as mobile apps or desktop software may be required to offer an optimal user experience and offline access to certain system features. Examples of software components could include Linux or Windows Server as the operating system, MySQL or PostgreSQL as the DBMS, Apache or Nginx as the web server, frameworks like Django or Ruby on Rails, and client applications developed for iOS and Android platforms.

**Infrastructure Requirements**

Infrastructure Requirements for the DriverPass system include a stable and high-speed internet connection for uninterrupted access to online features. Robust security measures, such as encryption and authentication, should be implemented to protect user data. Regular data backups and a disaster recovery plan are essential to prevent data loss and ensure business continuity. The system should also be designed with scalability and performance in mind, utilizing techniques like scalable server architecture, load balancing, and optimization strategies. Examples of infrastructure components could include a high-speed broadband internet connection, firewall and encryption technologies for security, cloud storage or backup solutions, and scalable server infrastructure using technologies like virtualization or containerization.

**Tools and Development Environment**

The Tools and Development Environment for the DriverPass system include an integrated development environment (IDE) for efficient code writing, debugging, and testing. A version control system should be used to manage source code versions and facilitate collaboration. Testing frameworks and tools are needed to ensure the system's quality, and project management tools can help track progress and facilitate communication. Examples of tools and technologies are IDEs like Visual Studio Code or JetBrains IntelliJ IDEA, Git for version control, frameworks like JUnit for testing, and project management tools such as Jira or Trello. The selection of specific tools and technologies will depend on the project's requirements and the development team's preferences and expertise.