# Eskinder Kassahun

# CS 255 System Design Document Template

## UML Diagrams

### UML Use Case Diagram

*This use case diagram effectively outlines the primary actors and their interactions with the system. Each actor—Customer, Secretary, IT Officer, and Owner—is clearly identified, with associated use cases that align with their respective roles. The inclusion of functionalities like Schedule Lesson, Access Practice Tests, and Generate Report covers essential system operations. The diagram provides a straightforward view of system interactions, allowing developers to understand user roles and system functionality requirements.A diagram of a diagram

Description automatically generated*

### UML Activity Diagrams

***Activity Diagram for Schedule Lesson*** *This diagram illustrates the process for a Customer to schedule a lesson, including date/time selection, availability check, and reservation confirmation.*

*A diagram of a diagram

Description automatically generated*

***Activity Diagram for Take Practice Test*** *This diagram demonstrates the flow of a Customer accessing, completing, and receiving feedback on an online practice test.*

*A white sheet with black text

Description automatically generated*

### UML Sequence Diagram

*This sequence diagram for Schedule Lesson captures the interaction between the Customer, System, and Secretary (for assisted bookings). The system checks availability, confirms reservations, and provides real-time feedback to the customer. The diagram also reflects cases where time slots may be unavailable, looping the customer back to select a different slot. This sequence diagram is beneficial for understanding the communication flow and ensuring the system meets user expectations for real-time booking.*

*A diagram of a machine

Description automatically generated*

### UML Class Diagram

*The class diagram accurately represents the core classes required for the DriverPass system: Customer, Appointment, TrainingPackage, and TestProgress. Each class contains relevant attributes that support system functionality, such as customerID for Customer and testID for TestProgress. Relationships between classes are clearly defined, with Customer associated with multiple Appointment and TestProgress entries, representing a customer’s multiple interactions with the system. This structured representation helps clarify the database and object-oriented structure needed for implementation.A diagram of a customer application

Description automatically generated*

## Technical Requirements

*The DriverPass system’s technical requirements need to secure, scalable, and accessible functionality for the diversity of its users in various roles and devices. Hardware-wise, the system will be using cloud-backed servers (AWS or Azure) to save data that can easily scale and provide high availability of access securely. This involves supporting client devices (PCs, tablets, and smartphones) so that external users such as customers, secretaries, or other IT staff can remotely access the system. In addition, fallible network equipment (and from the point of view of remote users and parties transmitting data real sea trouble). The backend will largely be based on a Linux environment which can scale up more easily (and of course, the reliability and security features). To manage structured data like customer, appointments, test results, and packages a relational Data-Base will be used (MySQL / PostgreSQL). To each backend development, Java or Python will be used to process the main operations such as scheduling services and account management (from "profile creation" stage onwards) & Security protocols. You are developing frontend, HTML, CSS, and JavaScript (React or Vue.js) will be used to develop the responsive web interface across all of our devices.*

*Visual Studio Code or Eclipse will be used for coding, Lucidchart to create design diagrams, and GitHub or GitLab for checking in code changes. Security is also significant, with SSL/TLS encryption to secure data transmissions, role-based access control (RBAC) to restrict access by user roles, and firewalls to prevent unauthorized server-side communications. Data loss can be easily recovered as the system will frequently auto-backup information. Lastly, it will interface with DMV databases to keep practice tests and course information current whenever regulations change. DriverPass will be strong, safe, and easy to use for all users, thanks to these technical features.*