# CS 255 System Design Document Template

## UML Diagrams

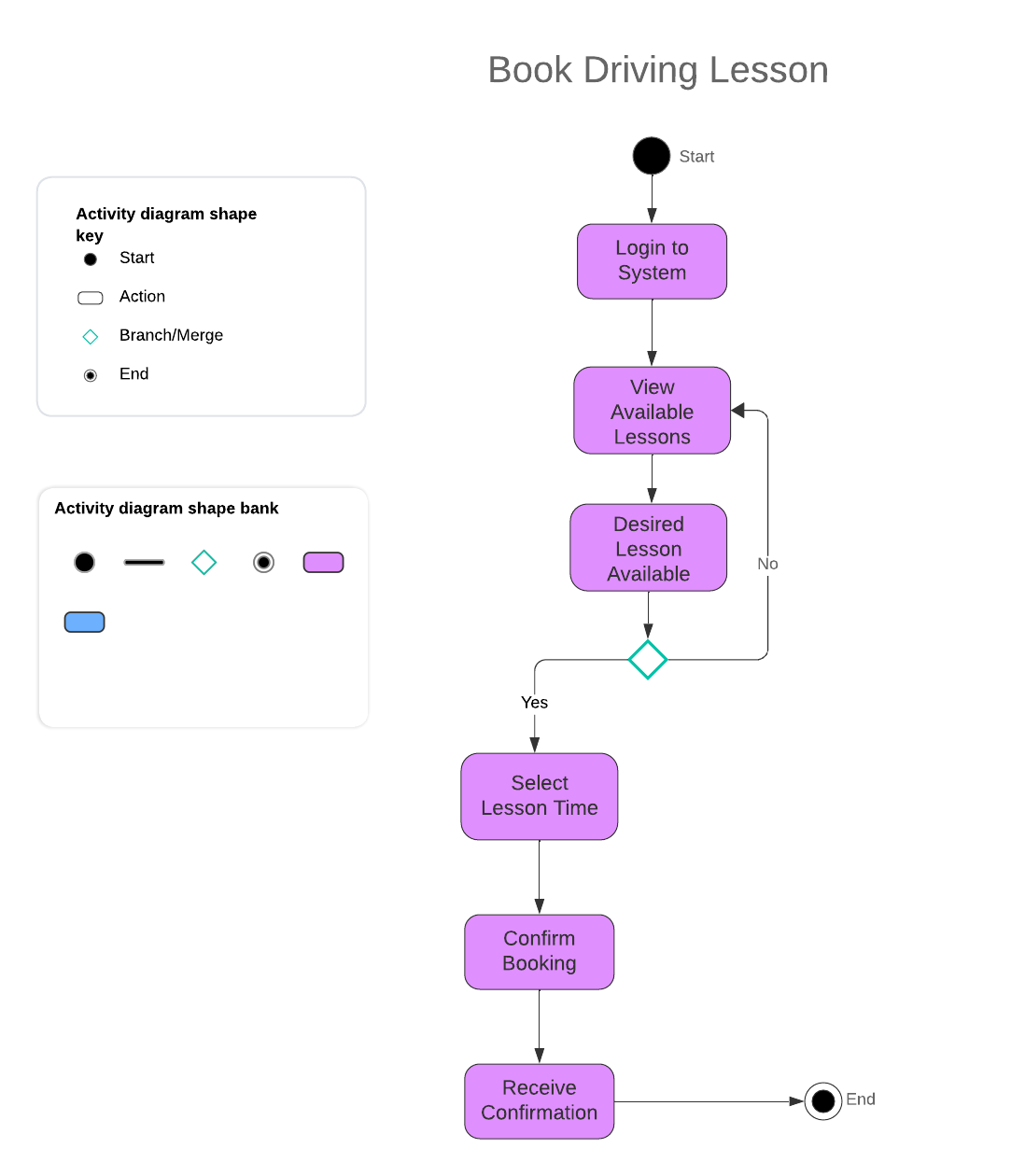
**Cody VanGosen**

### UML Use Case Diagram

A screenshot of a diagram

Description automatically generated

### UML Activity Diagrams



A diagram of a work flow

Description automatically generated

### UML Sequence Diagram

A computer screen shot of a diagram

Description automatically generated

### UML Class Diagram

A diagram of a data flow

Description automatically generated

## Technical Requirements

**Technical Requirements**

**Hardware Requirements:**

1. **Servers**:
   * **Database Server**: The system will require a dedicated server to host the **Test Database** and other data related to user accounts, test results, and lesson schedules. The database server must handle queries from the system to ensure real-time data retrieval and storage.
     + **Required Specifications**: 64-bit server architecture, 16GB RAM or higher, multi-core processor (minimum 4 cores), SSD for faster access to database queries.
   * **Application Server**: A second server is necessary to run the **DriverPass System**, handling interactions between the customers, administrators, and the system. This server will host the web application and mobile API.
     + **Required Specifications**: 64-bit server, 16GB RAM, multi-core processor, 1TB storage (depending on the scale of the user base).
2. **End-User Devices**:
   * **Workstations** and **mobile devices** (smartphones, tablets) will be used by **customers**, **office staff**, and **administrators** to access the system. These devices need to be capable of running web-based applications smoothly and accessing the mobile application.
   * **Client-Side Devices** should meet the following:
     + For desktop/laptop: 2GHz processor, 8GB RAM, 500GB storage.
     + For mobile: Android or iOS devices capable of running the latest version of their respective operating systems (Android 10+ or iOS 12+).

**Software Requirements:**

1. **Operating Systems**:
   * **Server OS**: The servers should run Linux-based systems (e.g., Ubuntu Server or CentOS) for better compatibility with most cloud services and reduced licensing costs.
   * **Client OS**: The system must be compatible with modern operating systems such as:
     + Windows 10 or later.
     + macOS.
     + Mobile support for Android and iOS.
2. **Database Management System (DBMS)**:
   * **MySQL** or **MongoDB** is required to handle data storage and retrieval for tests, user progress, lesson schedules, and other system data. MySQL will offer structured relational databases, while MongoDB can handle unstructured data more flexibly.
3. **Web Application Framework**:
   * **React.js** or **Angular.js** for the front-end of the web application, ensuring responsive design and interactive elements for users (customers and admins).
   * **Node.js** or **Django** for the back-end, managing server requests and ensuring smooth data flow between the user interface and the database.
4. **Mobile Application**:
   * For customers accessing the system via mobile devices, a **native mobile app** will need to be developed for both **Android** (using Kotlin/Java) and **iOS** (using Swift).
   * The mobile app will need to interact with the **DriverPass System** for features such as lesson scheduling, test-taking, and viewing progress.

**Tools:**

1. **UML Modeling Tool**:
   * **Lucidchart** was used to create the system’s UML diagrams, including the **Use Case**, **Activity**, **Sequence**, and **Class Diagrams**. This tool is essential for modeling the system architecture and behavior.
2. **Version Control**:
   * **Git** and **GitHub** or **GitLab** for version control of the software. This allows the development team to track changes, manage branches, and handle multiple versions of the system during development.
3. **IDE (Integrated Development Environment)**:
   * **IntelliJ IDEA**, **Eclipse**, or **Visual Studio Code** will be used to develop and maintain the system, with support for Java, Python, or JavaScript for back-end logic.
4. **Security Tools**:
   * **SSL/TLS Certificates** for ensuring secure communication between the user’s device and the system (for login, test submission, and other sensitive operations).
   * **Brute-force Protection Tools** to lock user accounts after several failed login attempts, as indicated in the security requirements.

**Infrastructure Requirements:**

1. **Cloud Hosting**:
   * The system should be hosted in the cloud for scalability and reliability. Services such as **Amazon Web Services (AWS)** or **Google Cloud Platform (GCP)** can provide the necessary infrastructure to host the application servers, database, and storage.
     + **Elastic Load Balancing**: To distribute traffic across multiple servers, ensuring high availability during peak usage times (e.g., test-taking periods).
     + **Cloud Storage**: For backing up user data, including test results, lesson schedules, and system logs.
2. **Network and Connectivity**:
   * **High-bandwidth Internet**: Required for seamless data transmission between the system, database, and end users. Both server-side and user-side bandwidth must support quick data retrieval, especially when submitting tests or checking lesson availability.
   * **Real-Time Synchronization**: The system will need real-time data synchronization for updates to lesson schedules and test results. This ensures no conflicts arise from simultaneous user activity (e.g., two users booking the same lesson time).
3. **Security Infrastructure**:
   * **Role-based Access Control (RBAC)**: Necessary to manage permissions for different user roles (Customer, Administrator, Office Staff). This ensures that only authorized personnel can modify lesson schedules or access sensitive customer data.
   * **Data Encryption (SSL/TLS)**: All sensitive data transfers (such as user login credentials or test results) should be encrypted in transit using SSL/TLS to prevent interception.
4. **Backup and Recovery**:
   * **Daily Backups**: Regular backups of the database and critical application files must be scheduled to prevent data loss in case of system failure.
   * **Disaster Recovery Plan**: Cloud services should include disaster recovery options such as automatic failover to backup servers.