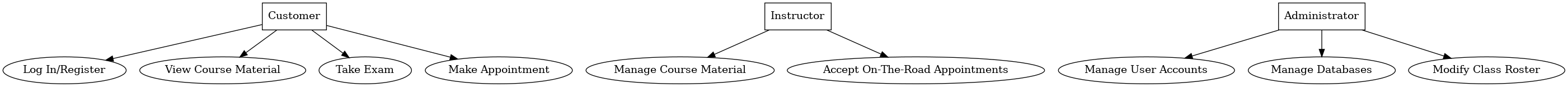
# CS 255 System Design Document

Jordan Walker

This template lays out all the different sections that you need to complete for Project Two. Each section has guidance to prompt your thinking. 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 what your client’s needs are. Remove this note when you are finished, and replace all bracketed text with the relevant information.

## UML Diagrams

### UML Use Case Diagram



### UML Activity Diagrams

A diagram of a computer program

Description automatically generated

### A diagram of a flowchart Description automatically generated

### UML Sequence Diagram

A diagram of a driver pass site

Description automatically generated

### UML Class Diagram

A diagram of a user information

Description automatically generated

## Technical Requirements

#### **Hardware Requirements**

1. **Servers:**
   * A dedicated server or cloud-based infrastructure with the following specifications:
     + **CPU:** 8 cores or higher.
     + **RAM:** 16 GB minimum.
     + **Storage:** 500 GB SSD storage for database and application deployment.
2. **Network Infrastructure:**
   * High-speed internet connection to ensure reliable access for all users.
3. **Client Devices:**
   * Supported devices: Windows/macOS/Linux computers or mobile devices (smartphones and tablets).
   * Minimum requirement: Dual-core processor, 4GB RAM, and modern web browsers (Google Chrome, Firefox, Safari).

#### **Software Requirements**

1. **Operating System:**
   * Servers: Windows Server 2019 or Linux (Ubuntu 20.04 LTS).
2. **Database Management System (DBMS):**
   * **SQL Server** or **PostgreSQL** for managing user data, exam records, and schedules.
3. **Application Stack:**
   * **Front-End:** React.js or Angular for dynamic, user-friendly interface design.
   * **Back-End:** Node.js with Express.js to handle server requests and business logic.
4. **Security Software:**
   * Firewalls and antivirus software to protect against malware and unauthorized access.
   * SSL/TLS encryption for secure data transmission.
5. **Backup and Recovery:**
   * Regular automated backups using cloud storage (e.g., AWS S3 or Azure Storage).

#### **Tools**

1. **UML Design Tool:**
   * Lucidchart for creating and maintaining UML diagrams.
2. **Development Tools:**
   * Visual Studio Code (or other IDEs).
   * GitHub or GitLab for version control and collaboration.
3. **Testing Tools:**
   * Selenium for automated UI testing.
   * Postman for API testing.
4. **Deployment Tools:**
   * Docker for containerization and deployment.
   * AWS Elastic Beanstalk or Azure App Service for scaling the application.

#### **Infrastructure**

1. **Cloud Infrastructure:**
   * AWS or Azure for hosting the application and database.
2. **Load Balancers:**
   * To distribute user requests and ensure high availability.
3. **Authentication Service:**
   * Integration with **OAuth 2.0** or multi-factor authentication (MFA) for securing user logins.