Daniel Mitchell

CS-255 System Analysis and Design

Project Two: System Design Document

## UML Diagrams

### UML Use Case Diagram

A diagram of a company

AI-generated content may be incorrect.

### UML Activity Diagrams

A diagram of a workflow

AI-generated content may be incorrect.

A diagram of a process

AI-generated content may be incorrect.

### UML Sequence Diagram

A diagram of a lesson

AI-generated content may be incorrect.

### UML Class Diagram

A diagram of a user flow

AI-generated content may be incorrect.

## Technical Requirements

Software:

* Web Front End: React.js or equivalent modern framework.
* Backend API: Node.js/Express or Python/FastAPI.
* Database: PostgreSQL with full audit logging.
* Authentication: Role-based access control (RBAC), hashed passwords, and account lockout after three failed attempts.
* Reporting: Built-in PDF/CSV generation for activity and progress.

Hardware/Infrastructure:

* Hosted in AWS/Azure/GCP environment.
* Load balancing and autoscaling for 500+ concurrent users.
* Continuous data backups and restore validation.

Security:

* HTTPS/TLS encryption for all web traffic.
* PCI-compliant payment gateway.
* Daily backup and regular penetration testing.
* Audit log of all create, update, and delete actions.

Integration:

* DMV data synchronization via API or secure upload.
* Third-party payment processor integration.

Performance:

* Page load: < 5 seconds.
* Report generation: < 1 minute.
* API response: < 10 seconds.

Limitations:

* No offline access (read-only mode for downloaded data only).
* Dependent on DMV for update accuracy.
* Non-developer users cannot add new course modules.