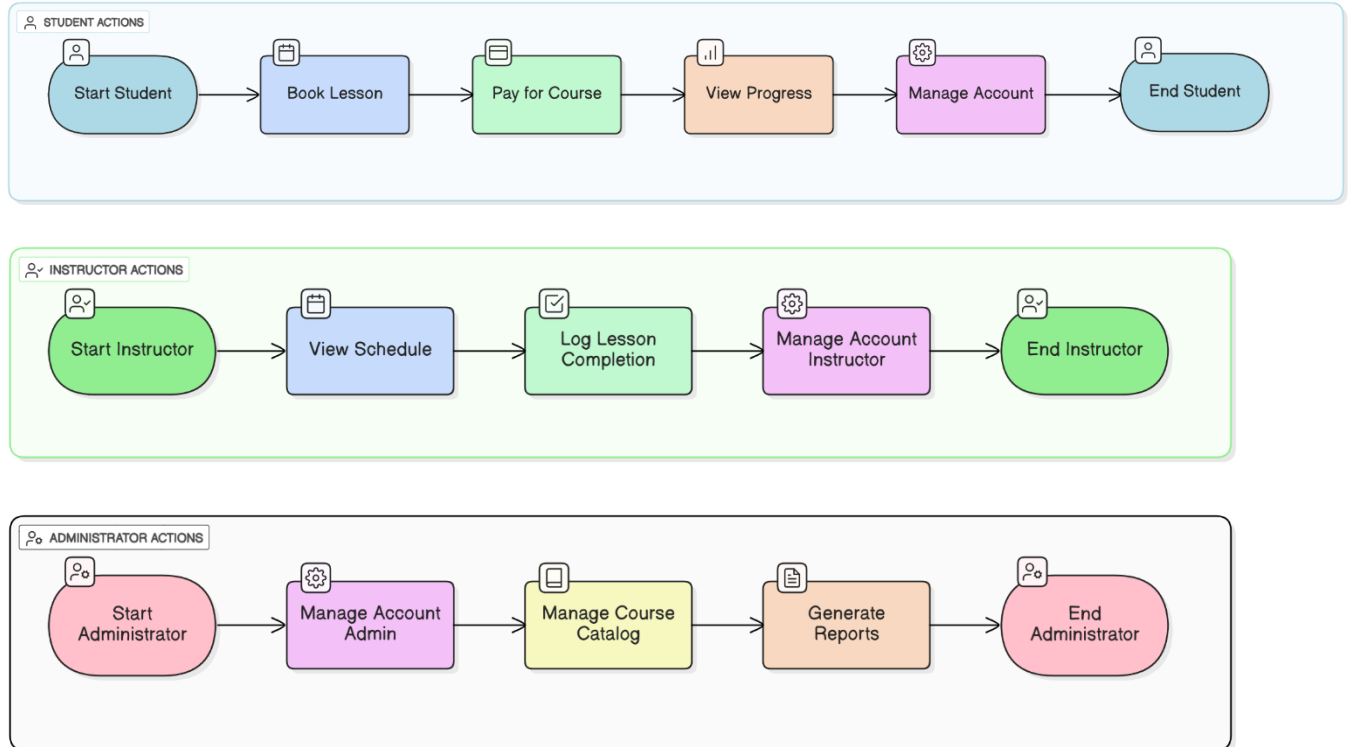# CS 255 System Design Document
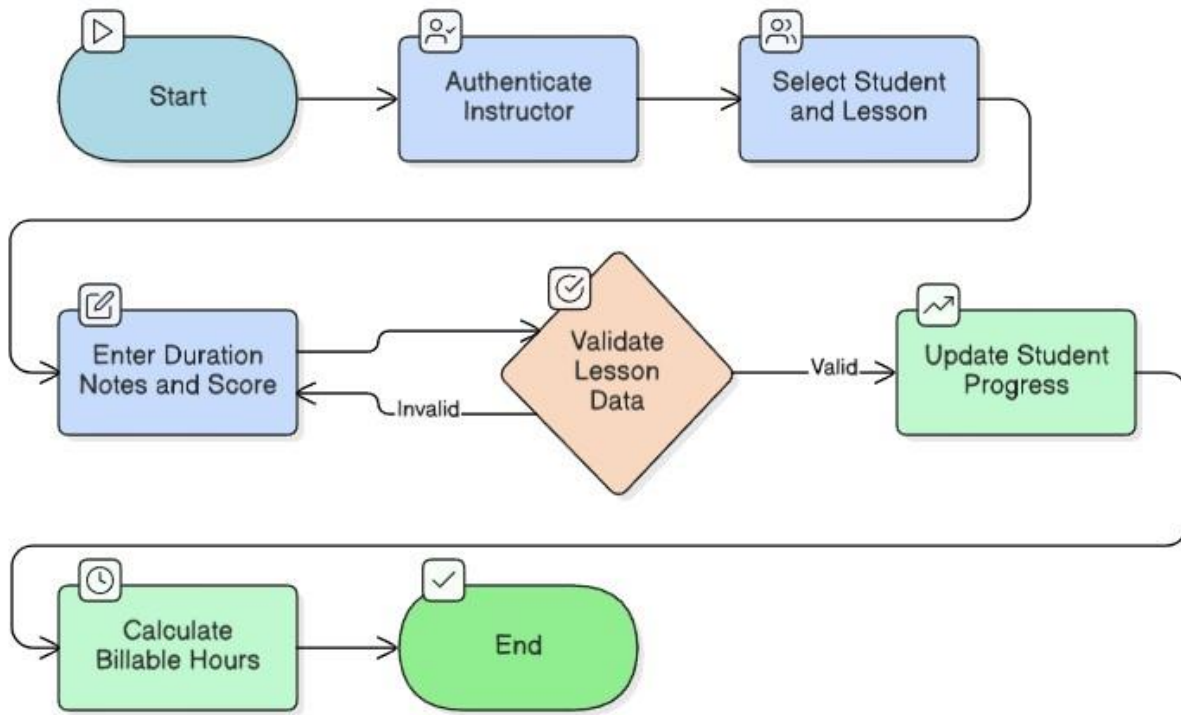
UML Diagrams

UML Use Case Diagram
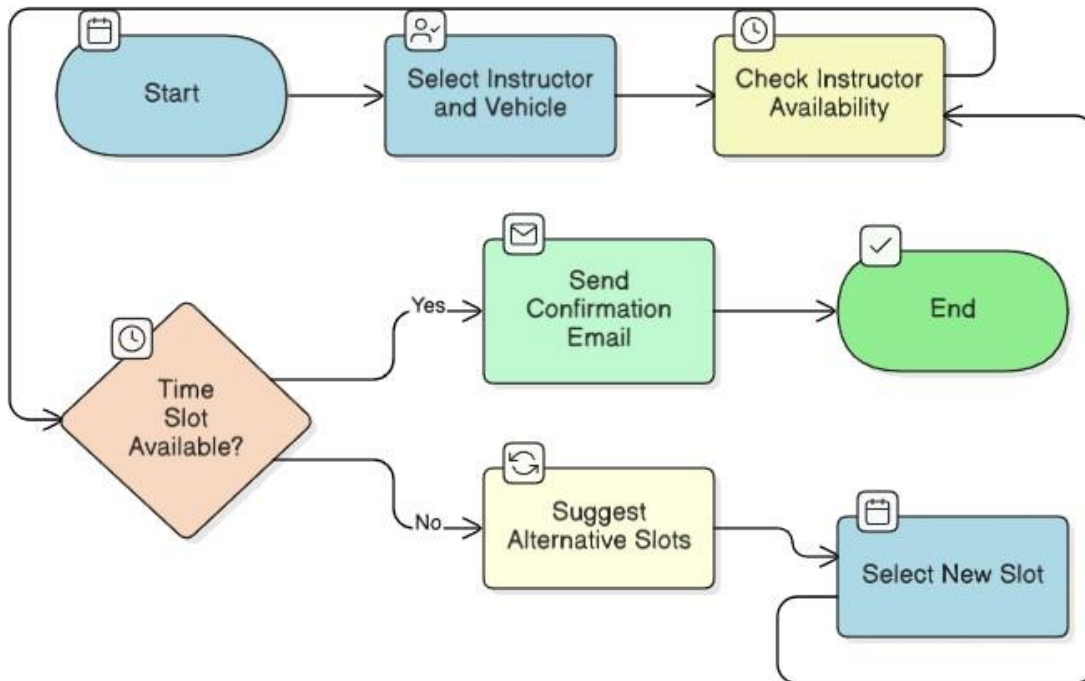
UML Activity Diagrams
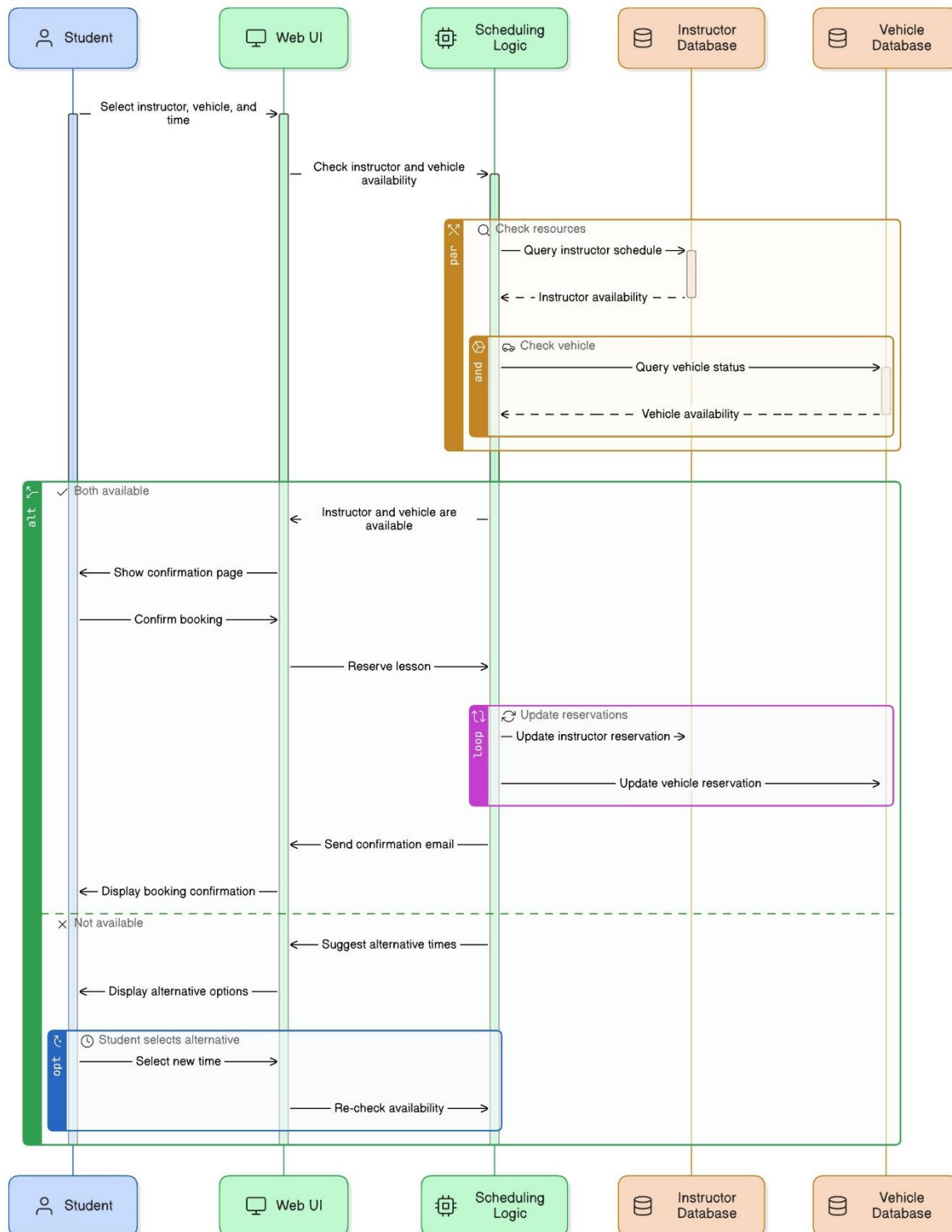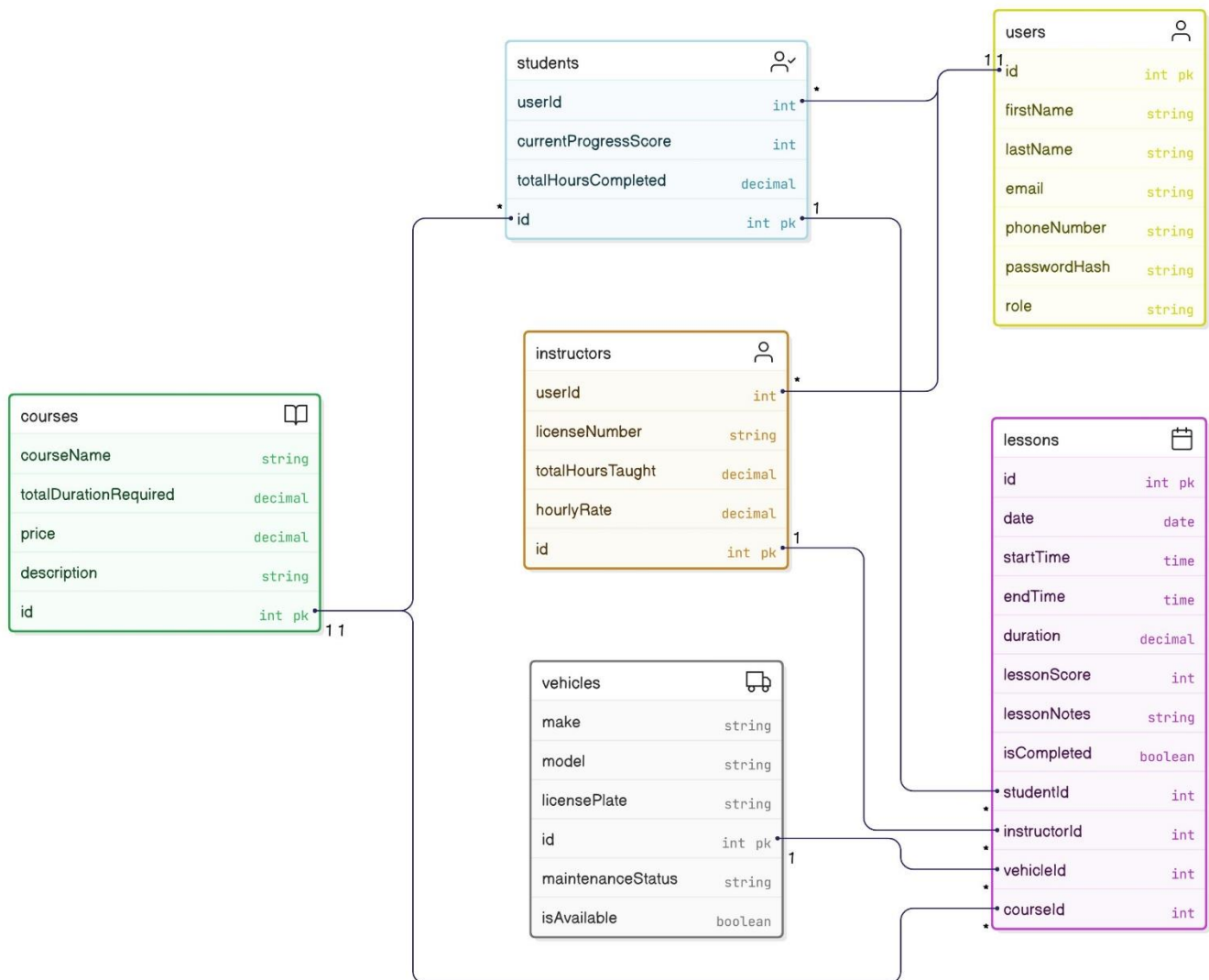
Lesson Scheduling

Lesson Logging

UML Sequence Diagram

UML Class Diagram

Technical Requirements

1. Hardware Requirements

Component	Requirement	System Support
Server Infrastructure	Cloud-based hosting with automatic scaling and high-speed SSD storage.	Ensures high availability to support 24/7 student booking and fast database access for real-time schedule checks.
Database Server	Dedicated relational database with multi-region backup capability.	Guarantees data integrity for all linked entities and protects critical billing/progress data.
Instructor Devices	Modern tablets or smartphones with reliable 4G/5G mobile data access.	Necessary for instructors to securely authenticate and Log Lesson Completion immediately after class (Activity Diagram 1.0).

2. Software & Tool Requirements

Component	Requirement	System Support
Database Management System	PostgreSQL	Provides strong transactional support, which is essential for safely updating schedules and logging scores without errors.
Frontend Framework	React	Allows for a responsive, mobile-first design, ensuring a good experience for students booking lessons and instructors logging data on small screens.
Backend Framework	Node.js or Python	Handles the complex Scheduling Logic shown in the Sequence Diagram, efficiently coordinating checks across Instructor and Vehicle databases.
Payment Gateway	Integration with a secure, PCI-compliant service	Required for the Pay for Course use case, ensuring secure financial data handling and reducing liability.

3. Infrastructure & Security Requirements

Component	Requirement	System Support
Data Security	SSL encryption for all data in transit, 256 encryption for all data at rest.	Protects sensitive student and instructor data from unauthorized access, supporting the Security requirements.
Access Control	Implementation of role based access control based on the role attribute in the User Class.	Ensures that students can only View Progress and instructors can only Log Lesson Completion, maintaining data separation.
Deployment	Continuous Integration pipelines using tools like GitHub Actions or GitLab.	Allows for rapid, non-disruptive updates and fixes, ensuring the system remains current and reliable.
Monitoring	Automated system monitoring to track server performance and response times.	Essential for proactively identifying slowdowns that could affect the real-time Book Lesson sequence.