# CS 255 System Design Document Template

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

The use case diagram below illustrates the interactions between different actors (Customer, Secretary, Driving Instructor, Owner, IT Officer, and DMV System) and the various functions they can perform within the DriverPass system. This diagram captures all the key functionalities identified in the business requirements, including user authentication, package registration, lesson scheduling, online learning, and administrative functions.

*A diagram of a company structure

AI-generated content may be incorrect.*

### UML Activity Diagrams

The following activity diagrams detail the step-by-step flow for two critical use cases: "Schedule Lesson" and "Take Practice Tests". These diagrams show the decision points, parallel processes, and the complete workflow from start to finish for each use case.

A diagram of a flowchart

AI-generated content may be incorrect.A diagram of a process

AI-generated content may be incorrect.

### UML Sequence Diagram

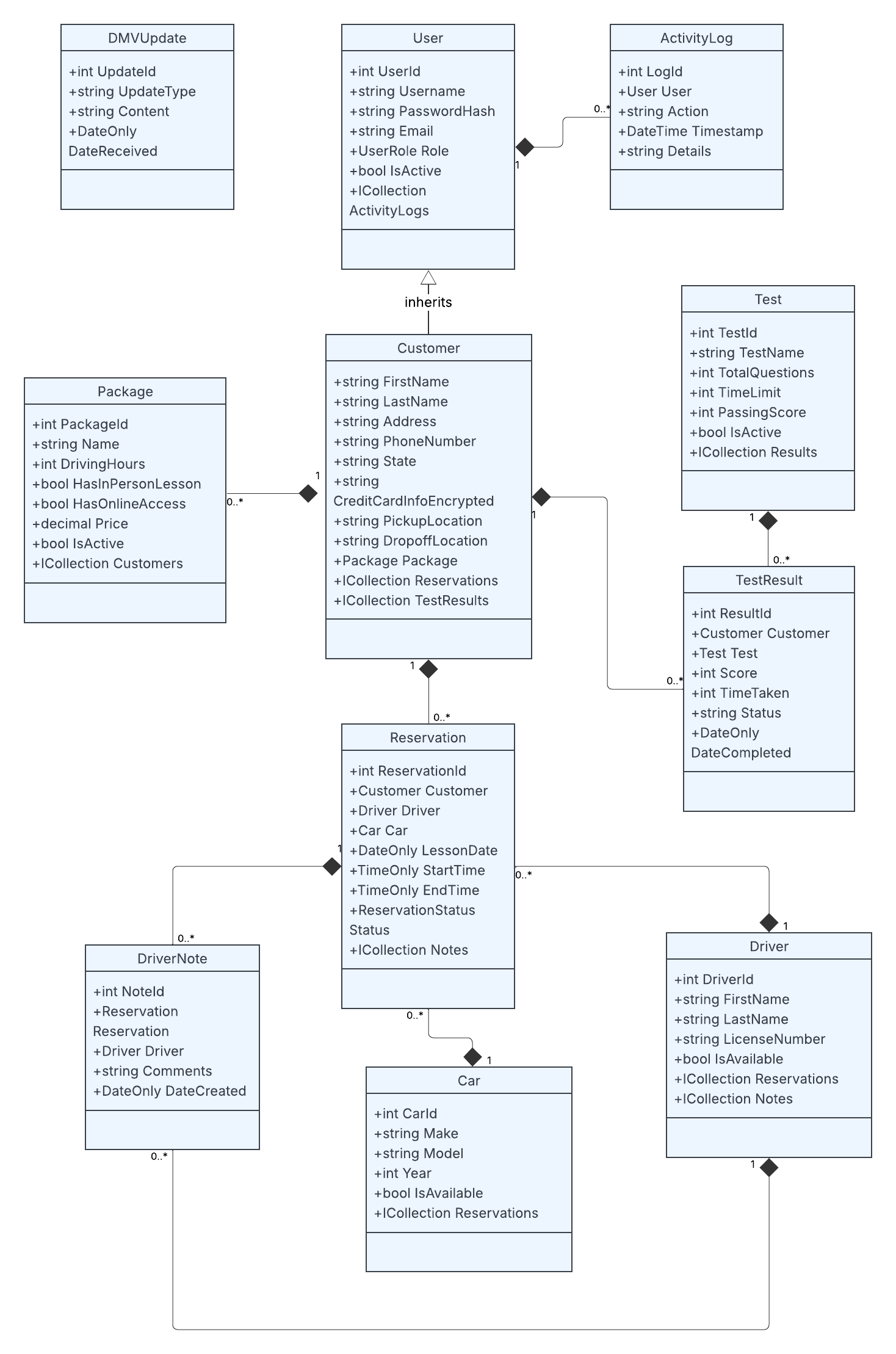
This sequence diagram shows the interaction between different system components for the "Schedule Lesson" use case, illustrating how objects communicate with each other over time to complete this critical business function.

A diagram with text and numbers

AI-generated content may be incorrect.

### UML Class Diagram

The class diagram represents the static structure of the DriverPass system, showing the main classes, their attributes, and relationships. This diagram provides the foundation for the database design and system architecture.



## Technical Requirements

Based on the functional and nonfunctional requirements identified in the business requirements document, the DriverPass system requires the following technical infrastructure and components:

**Hardware Requirements**

**Server Infrastructure:**

* Cloud-based hosting platform (AWS, Azure, or Google Cloud)
* Scalable compute instances to handle multiple concurrent users
* Load balancers for distributing traffic during peak usage periods
* Sufficient storage capacity for user data, test content, and activity logs

**Client Requirements:**

* Compatible with desktop computers (Windows, macOS, Linux)
* Mobile device support (iOS and Android tablets/smartphones)
* Minimum 1GB RAM and stable internet connection for optimal performance

**Software Requirements**

**Web Server Technology:**

* Apache or Nginx web server
* SSL/TLS encryption for secure HTTPS connections
* Content Delivery Network (CDN) for fast content delivery

**Database System:**

* Relational database management system (MySQL, PostgreSQL, or SQL Server)
* Database clustering for high availability
* Automated backup and disaster recovery solutions
* Data encryption at rest and in transit

**Development Framework:**

* Web-based application using modern frameworks (React, Angular, or Vue.js for frontend)
* Server-side technology (Node.js, Python Django, or Java Spring)
* RESTful API architecture for system integration
* Responsive web design for cross-platform compatibility

**Security Infrastructure**

**Authentication and Authorization:**

* Multi-factor authentication capabilities
* Role-based access control (RBAC) system
* Password encryption using industry-standard algorithms
* Session management and timeout controls
* Brute force protection with account lockout mechanisms

**Data Protection:**

* PCI DSS compliance for credit card processing
* Data encryption using AES-256 standards
* Secure API endpoints with rate limiting
* Regular security audits and vulnerability assessments

**Integration Requirements**

**DMV System Integration:**

* API endpoints for receiving DMV updates
* Webhook support for real-time notifications
* Data synchronization protocols
* Error handling and retry mechanisms

**Payment Processing:**

* Integration with secure payment gateways (Stripe, PayPal, or Square)
* Tokenization of sensitive payment information
* Automated billing and invoice generation
* Refund and chargeback handling capabilities

**Performance and Scalability**

**System Performance:**

* Page load times under 3 seconds
* Support for 500+ concurrent users
* 99.9% uptime availability
* Automatic scaling based on traffic demands

**Monitoring and Analytics:**

* Real-time system monitoring and alerting
* Performance metrics tracking
* User activity analytics
* Error logging and debugging tools

**Backup and Recovery**

**Data Backup:**

* Automated daily backups with 30-day retention
* Geographic redundancy for disaster recovery
* Point-in-time recovery capabilities
* Regular backup testing and validation

**Business Continuity:**

* Failover mechanisms for system components
* Recovery Time Objective (RTO) of 4 hours
* Recovery Point Objective (RPO) of 1 hour
* Documented disaster recovery procedures

**Compliance and Standards**

**Regulatory Compliance:**

* GDPR compliance for data privacy
* ADA compliance for accessibility
* SOC 2 Type II certification for security controls
* Regular compliance audits and reporting

**Technical Standards:**

* W3C web standards compliance
* Mobile-first responsive design
* Cross-browser compatibility (Chrome, Firefox, Safari, Edge)