# 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

### A diagram of a software system Description automatically generatedUML Use Case Diagram

[LucidChart Link To UML Use Case Diagram](https://lucid.app/lucidchart/0a9a310f-ba63-42eb-81f9-4d6822608eb0/edit?invitationId=inv_f210b964-340a-4a59-9e77-bf1de062e951)

### A diagram of a payment process Description automatically generatedA diagram of a process Description automatically generatedUML Activity Diagrams Making a Reservation Activity Diagram Creating an Account Activity Diagram

[LucidChart Link to Reservation Activity Diagram](https://lucid.app/lucidchart/6bbab6c4-0893-45af-809d-59d10361d307/edit?invitationId=inv_77c4637f-6a0d-44e2-b296-ded01f611a72) [LucidChart Link to Account Creation Activity Diagram](https://lucid.app/lucidchart/3f797207-94e8-4234-a9dc-8c0b18b4dd08/edit?invitationId=inv_5d06df98-b238-49ca-98fd-640edeaa55d7)

### UML Sequence Diagram

Creating An Account Sequence Diagram

A diagram of a workflow

Description automatically generated

[LucidChart Link to UML Sequence Diagram](https://lucid.app/lucidchart/8abcc1e1-bb53-4bd0-8047-69c1a6a16e22/edit?viewport_loc=-10223%2C-3653%2C18523%2C8295%2C0_0&invitationId=inv_84086eb4-3f93-4bad-bd1e-c7087afc5af5)

### UML Class Diagram

A diagram of a software company

Description automatically generated with medium confidence

[LucidChart Link to UML Class Diagram](https://lucid.app/lucidchart/335a41db-a3ad-42fb-a419-925d3a00adbc/edit?invitationId=inv_56d27941-d952-43c3-976b-4fe29ed25413)

## Technical Requirements

*The DriverPass System necessitates an infrastructure that is capable of robustness and scalability to meet the functional and non-functional requirements. It should, therefore, be hosted on high-performance servers that would handle peak usage with ease while the web and application servers manage user requests and business logic. In addition, there is the need for a scalable database system, such as MySQL, PostgreSQL, or MongoDB, to be used for the storage of user information, reservations, payments, and progress reports. Besides, load balancers shall be implemented to evenly distribute the traffic to prevent degradation of performance. The cloud infrastructure solution-AWS, Google Cloud, or Microsoft Azure-is to be used for elastic compute resources, managed databases, and secure storage with high availability and scalability.*

*On the software side, the system requires a modern web application framework, with frontend technologies like React or Angular to build intuitive and responsive user interfaces, and backend frameworks like Node.js or Django to manage business logic and API communication. Security is paramount; hence, the system should implement SSL/TLS encryption to securely transmit data, use OAuth 2.0 or JWT for user authentication, and password hashing, such as bcrypt, to store credentials. Payment processing should be secured through third-party gateways like Stripe or PayPal. This requires encrypted communication of real-time validations to make sure that transactions are processed accordingly.*

*The system should also support a full monitoring and reporting framework, utilizing tools such as Prometheus, Datadog, or the ELK stack for tracking system health, detecting suspicious activity, and ensuring smooth operation. Automated testing with Jest and Mocha will ensure the reliability of the system, while a CI/CD pipeline-e.g., Jenkins or GitHub Actions-will automate deployment and reduce any downtime during updates. To support these increasing user loads and volumes of transactions, horizontal scaling solutions and caching mechanisms like Redis will be critical to maintaining fast response times and performance during peak usage.*

*Security measures will be implemented in the form of WAF and DDoS protection. Regular backup and disaster recovery plans will be in place to ensure data integrity and recoverability, while the overall system architecture should allow for upgrades and maintenance with minimal service disruption. Clients will access the system through an internet connection; therefore, performance optimizations must consider variable networking conditions to provide a seamless user experience even in low network connectivity scenarios.*

*In summary, the DriverPass System will rely on an infrastructure that is secure, scalable, and responsive with the integration of cloud-based services, modern web technologies, and robust security features. It will ensure a flawless experience for all user roles involved and make operations run smoothly without interference to data.*