



**AlgoTron**

**MAST**

**SYSTEM D**

**IN JUST 2**

## Day 1

# Introduction to System Design

- ◆ Understand the importance of system design for building reliable systems.
- ◆ Explore the key components of system design: load balancing, caching, and databases.

## Day 2

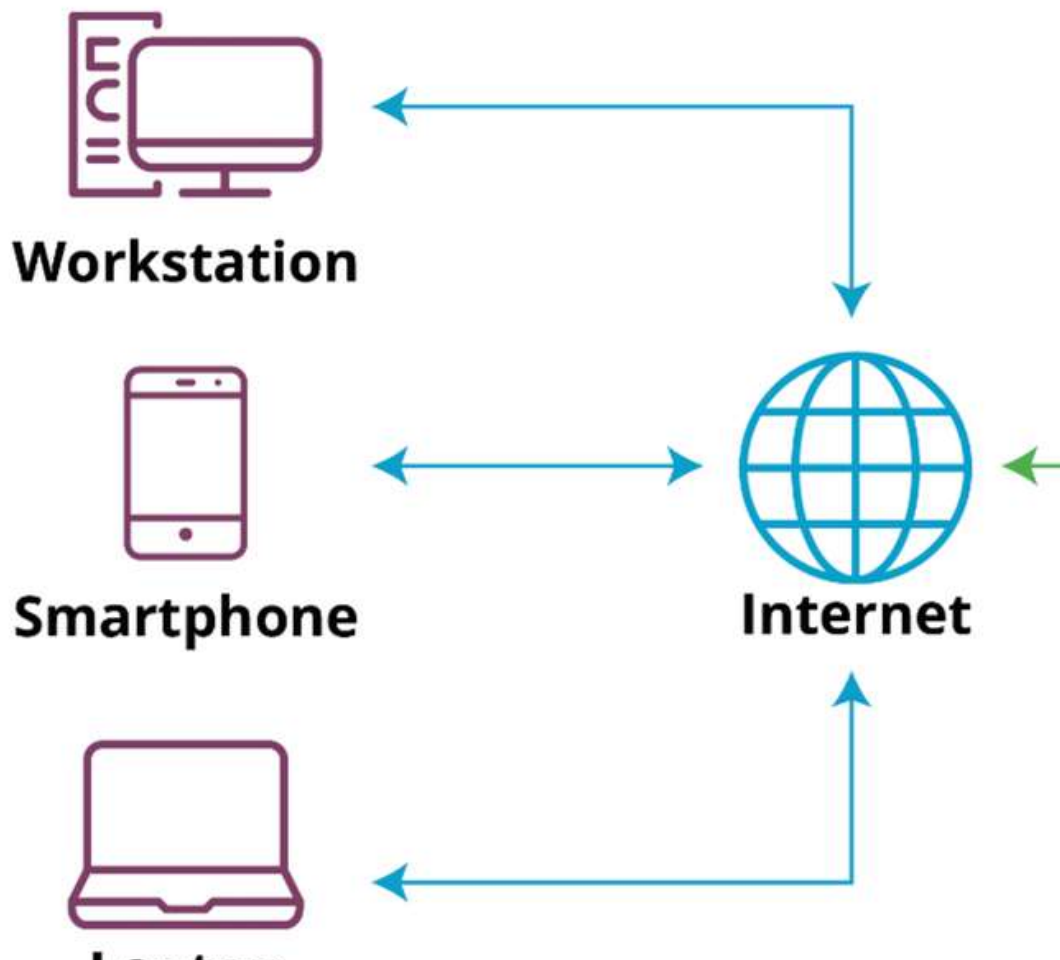
# Networking Basics

- ◆ Dive into networking concepts, including IP addressing, DNS, and routing.

## Day 3

# Understanding Client-Server

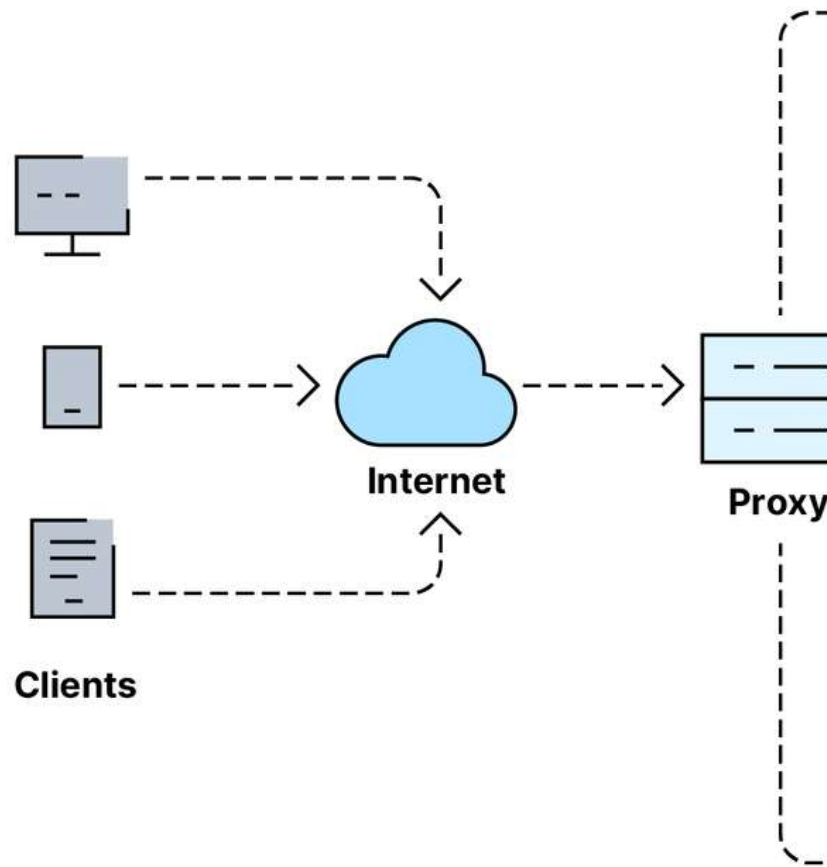
- ◆ Explore the client-server model and its components.
- ◆ Study the roles and responsibilities of a distributed system.



## Day 5

# Load Balancing

- ◆ Learn about load balancers and their role
- ◆ Study load balancing algorithms and their implementation



## Day 6

# Caching

## Day 8

# Distributed Systems Basics

- ◆ Dive into the fundamentals of distributed systems
- ◆ Learn about distributed computing models

## Day 9

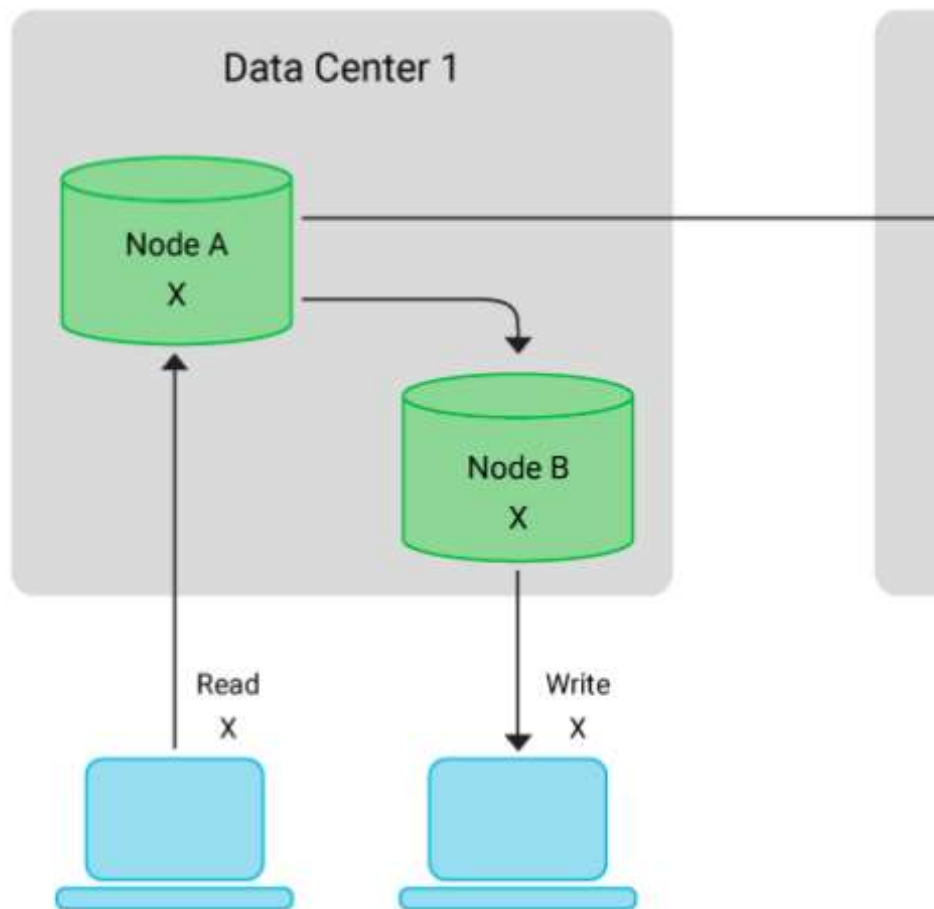
# CAP Theorem

- ◆ Study the CAP theorem and its implications
- ◆ Understand the trade-offs between Consistency, Availability, and Partition Tolerance.

## Day 10

# Eventual Consistency

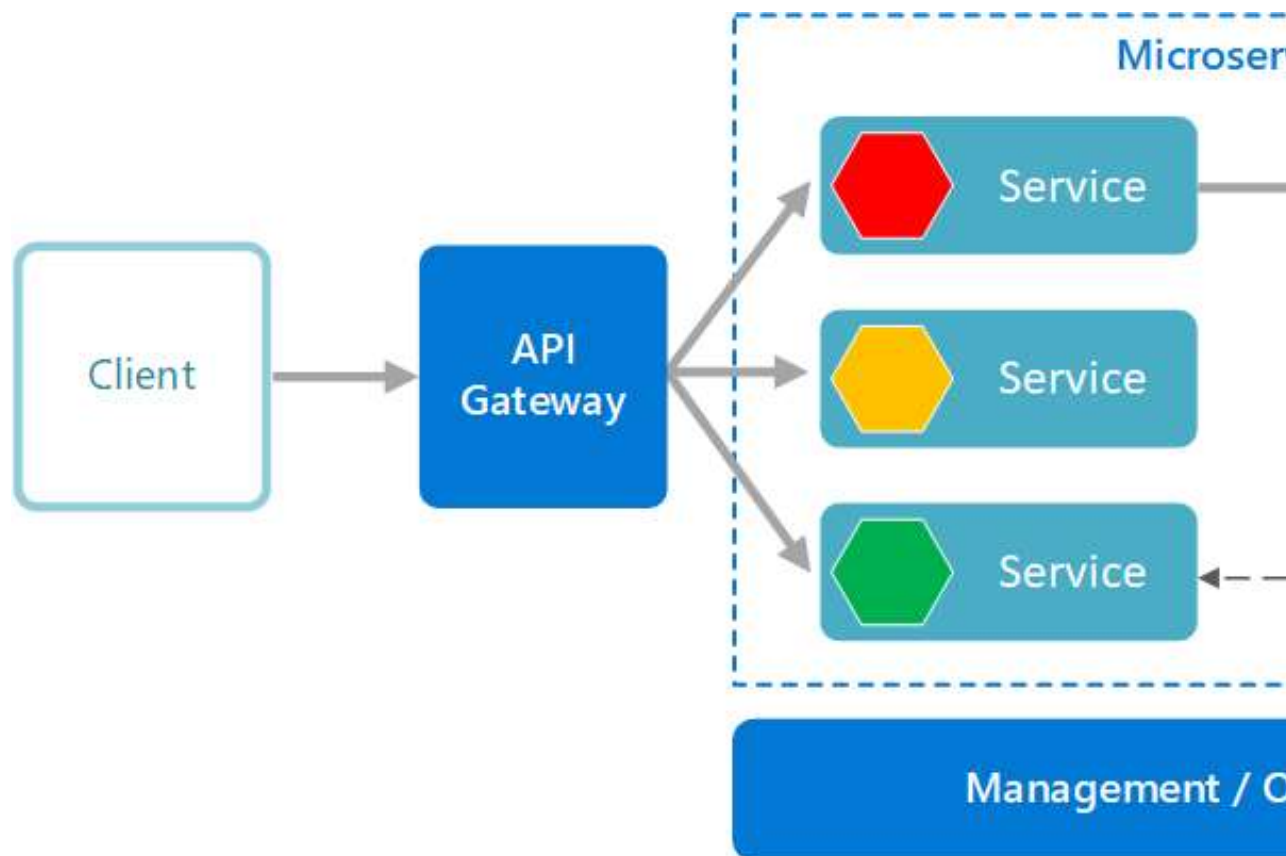
- ◆ Explore the concept of eventual consistency
- ◆ Study how systems achieve consistency



## Day 12

# Microservices Architecture

- ◆ Explore microservices architecture and
- ◆ Learn about service discovery, comm



## Day 13

## Day 15

# Case Studies

- ◆ Analyze real-world case studies of systems
- ◆ Learn from successful system design

## Day 16

# Cloud Computing

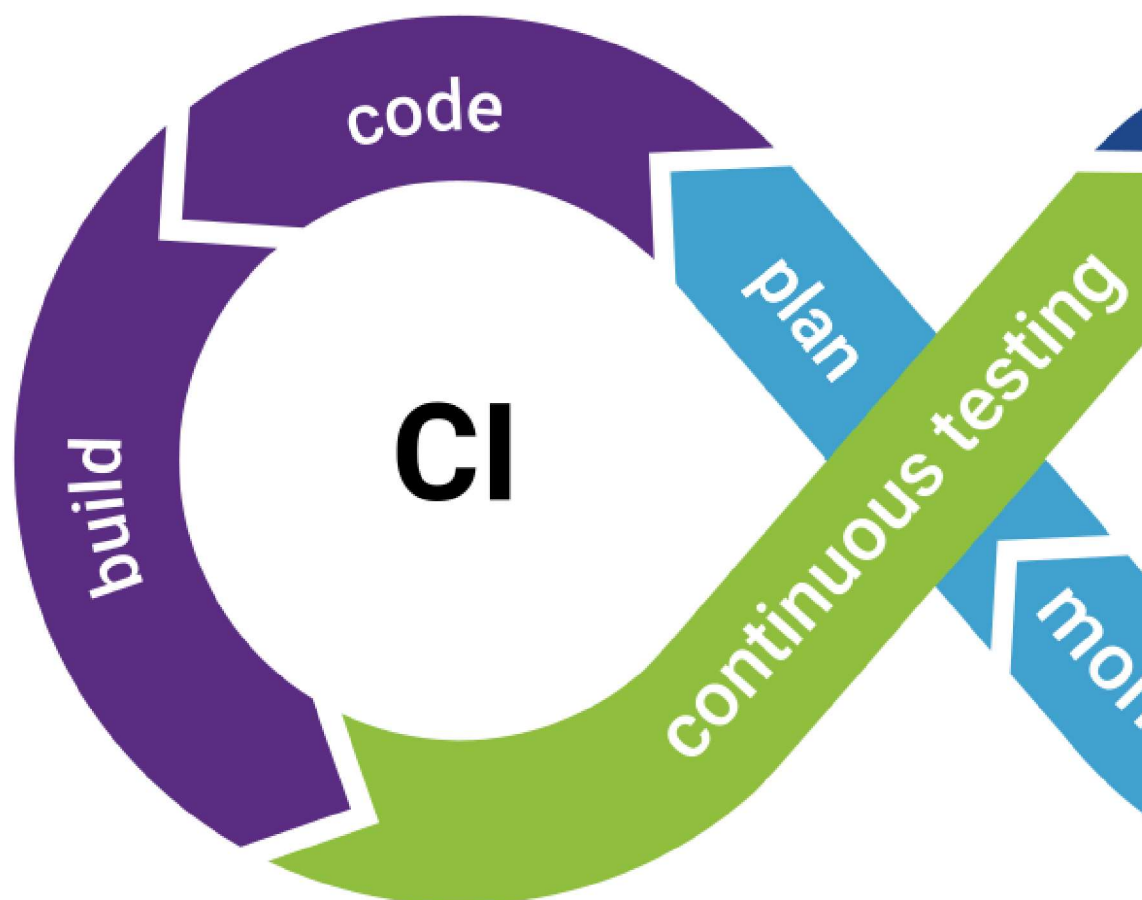
- ◆ Explore cloud services from providers
- ◆ Study how to design and deploy systems



## Day 17

# DevOps and Continuous Integration/Deployment (CI/CD)

- ◆ Learn about DevOps practices and CI/CD
- ◆ Study how they are integrated into systems



## Day 19

# Performance Optimization

- ◆ Learn about performance monitoring
- ◆ Study profiling tools and techniques.

## Day 20

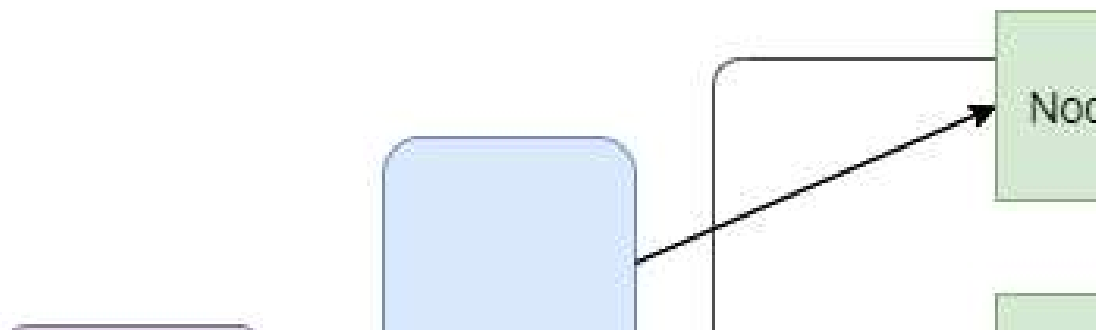
# Review and Practice

- ◆ Review key concepts from the past 20
- ◆ Work on design exercises and case stu

# Important System Interview Questions

## 1. Design a URL shortening service

- Design a service to shorten long URLs
- Key Components: URL shortening algorithm, database storage, analytics.
- Additional: Customizable short URLs



## 2. Design a Authentication S

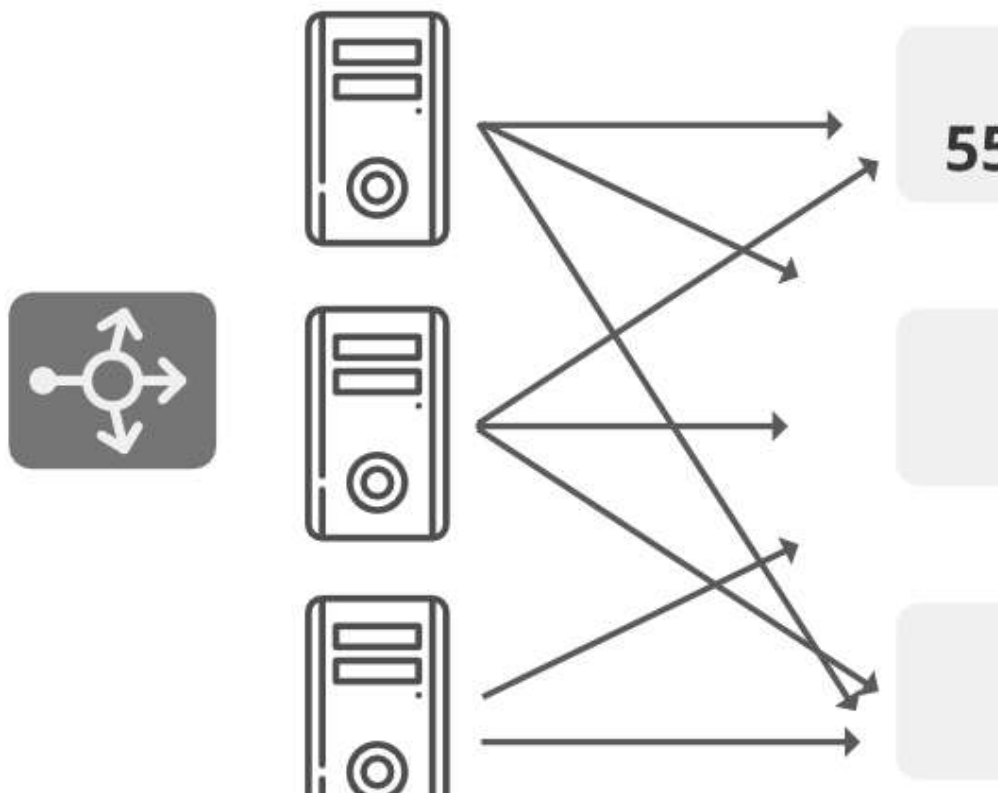
- Design a centralized authentication access.
- Key Components: User authentication encryption.
- Additional: Multi-factor authentication



Auth Service

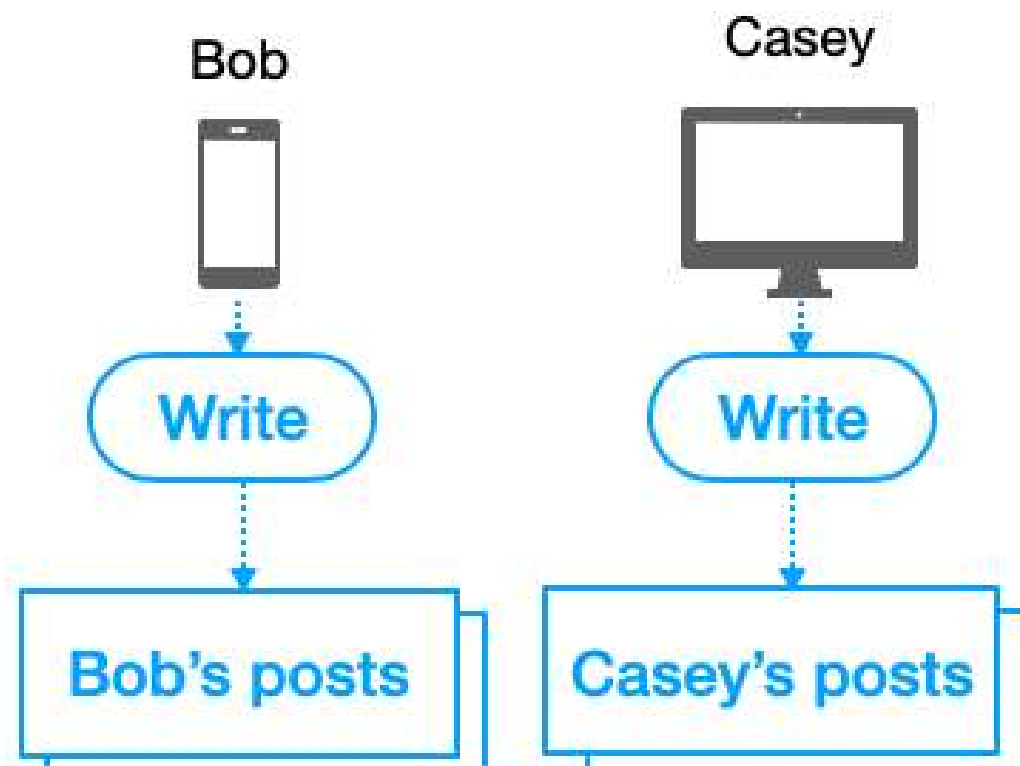
### 3. Design a Cache System:

- Create a caching system for frequent
- Key Components: Cache eviction p  
distributed caching.
- Additional: Support for cache inval



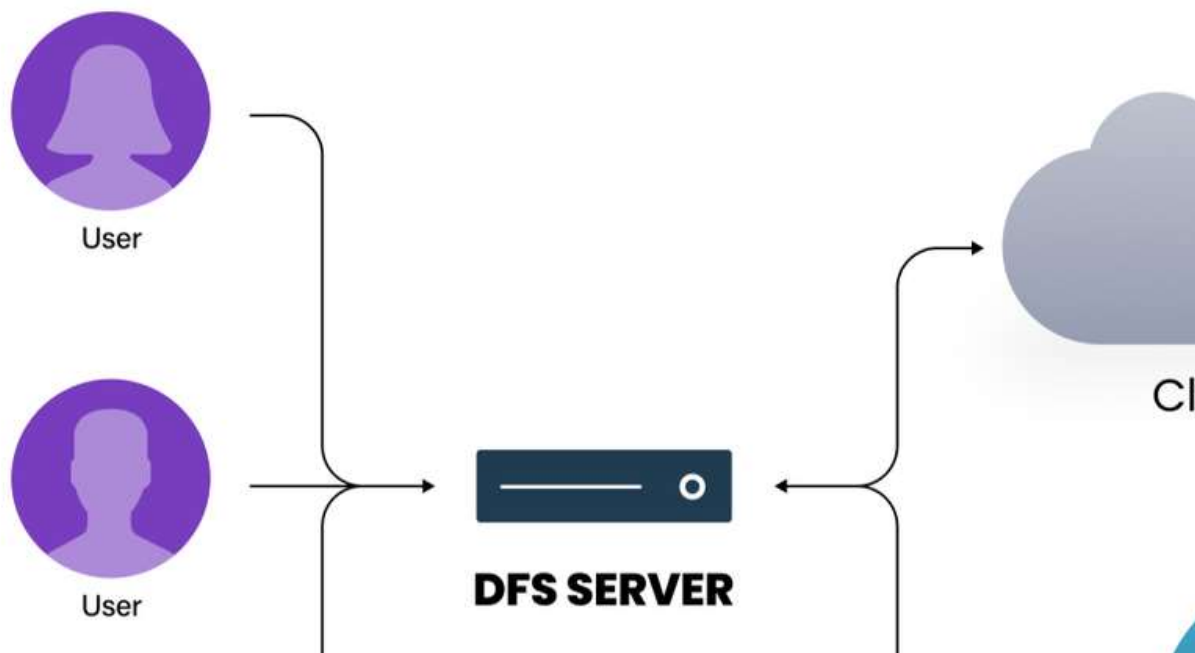
## 4. Design a Social Media Feed

- Design a scalable social media feed
- Key Components: Feed generation and delivery.
- Additional: Real-time trending topics



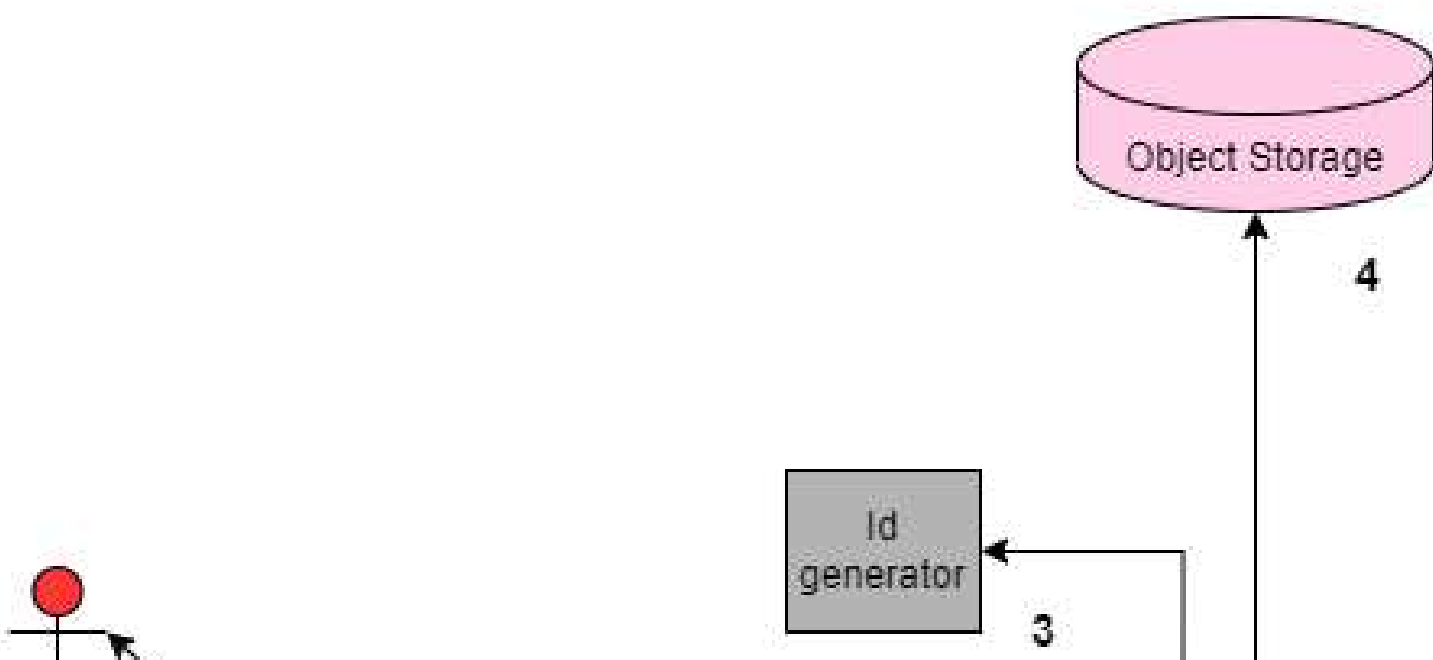
## 5. Design a Distributed File S

- Design a distributed file system like
- Key Components: NameNode, Data
- tolerance.
- Additional: Support for large file s



## 6. Design a Chat Application

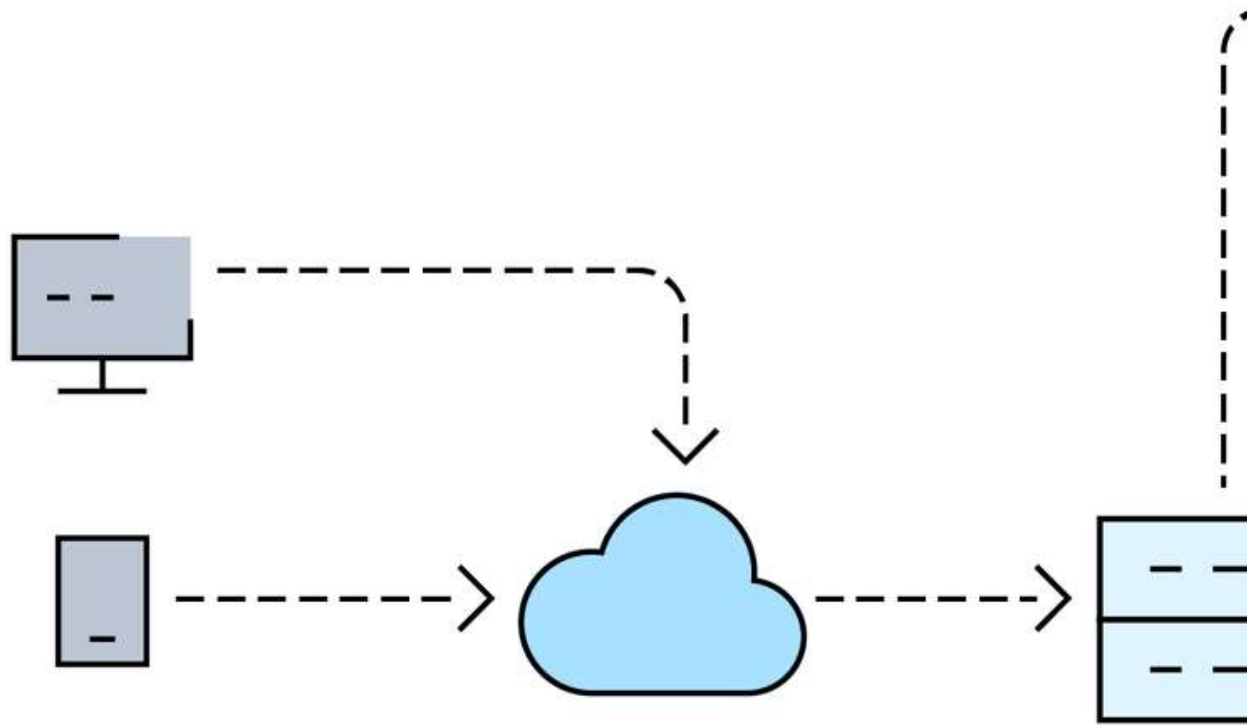
- Design a chat application for millions of users.
- Key Components: Messaging protocol for 1:1 and group chats.
- Additional: End-to-end encryption.





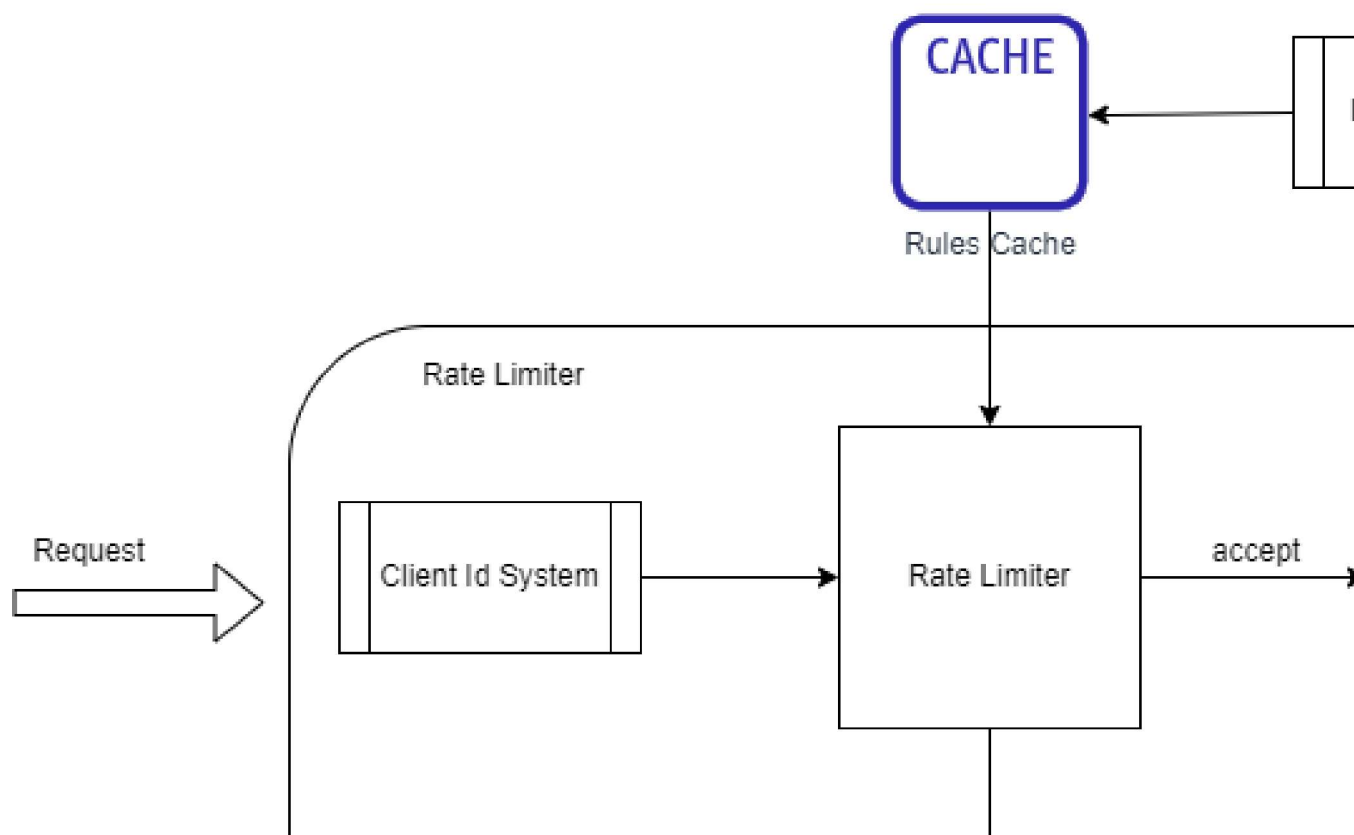
## 7. Design a Load Balancer:

- Design a load balancing system for
- Key Components: Load balancing  
server scaling.
- Additional: Global load balancing



## 8. Design a Rate Limiter:

- Design a rate limiter to prevent service from being overwhelmed by too many requests.
- Key Components: Token bucket algorithm for rate limiting.
- Additional: Dynamic rate limiting based on client behavior.



## 9. Design a Key-Value Store:

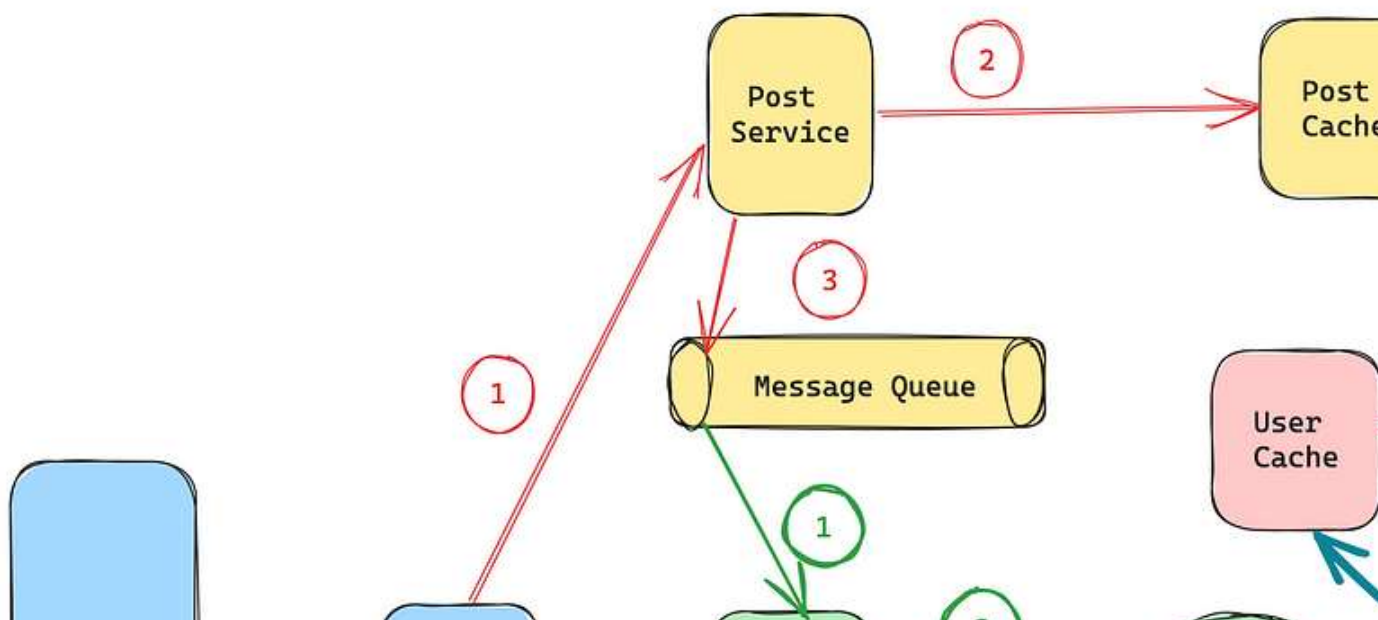
- Design a distributed key-value store
- Key Components: Partitioning, replication, consistency, availability, and fault tolerance.
- Additional: Support for secondary

key	value
123	123 Main St.
126	(805) 477-39



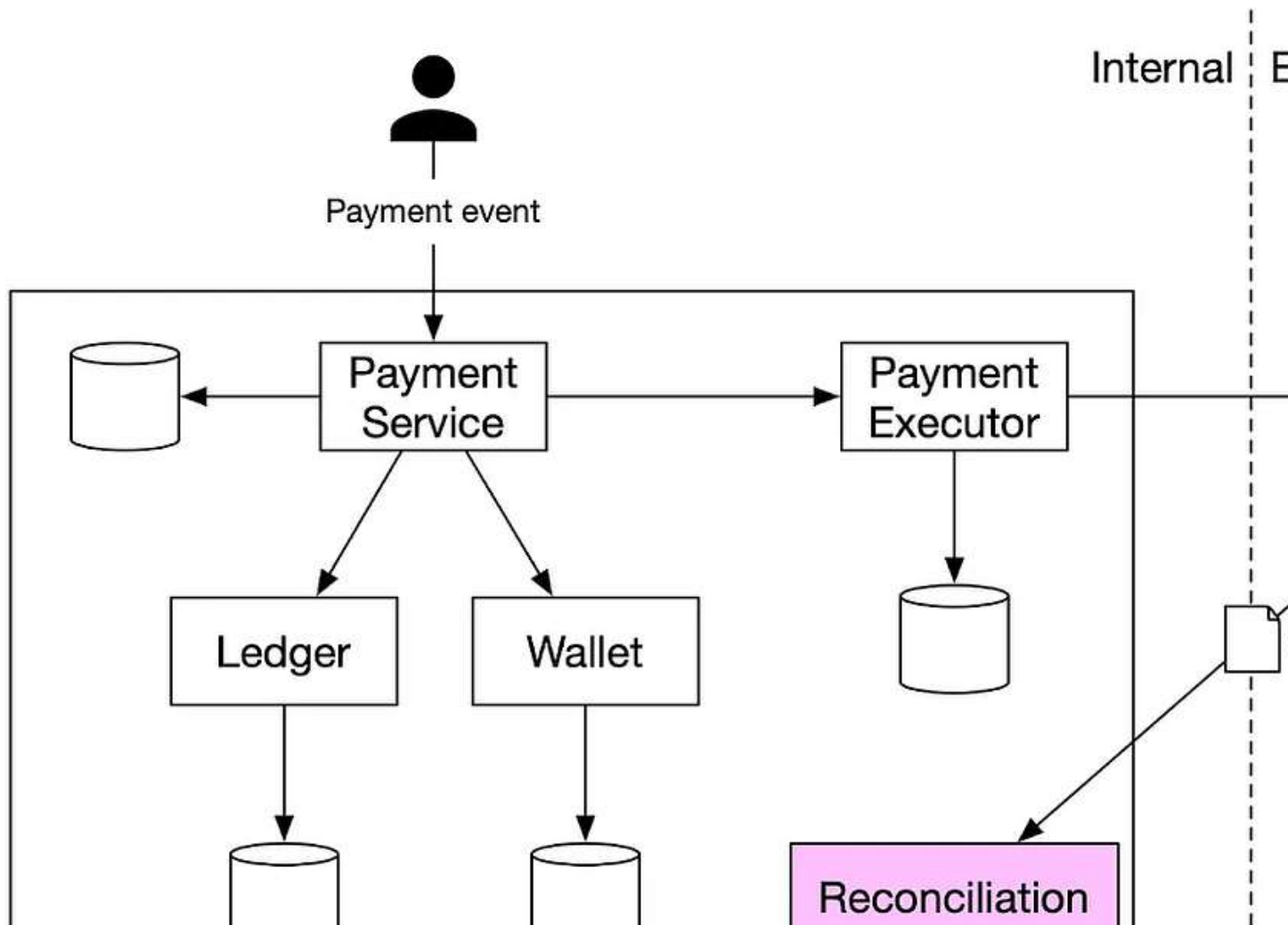
## 10. Design a scalable news feed (Facebook's news feed):

- Design a system for generating personalized news feed.
- Key Components: Content aggregation, ranking algorithm.
- Additional: Personalized notifications.



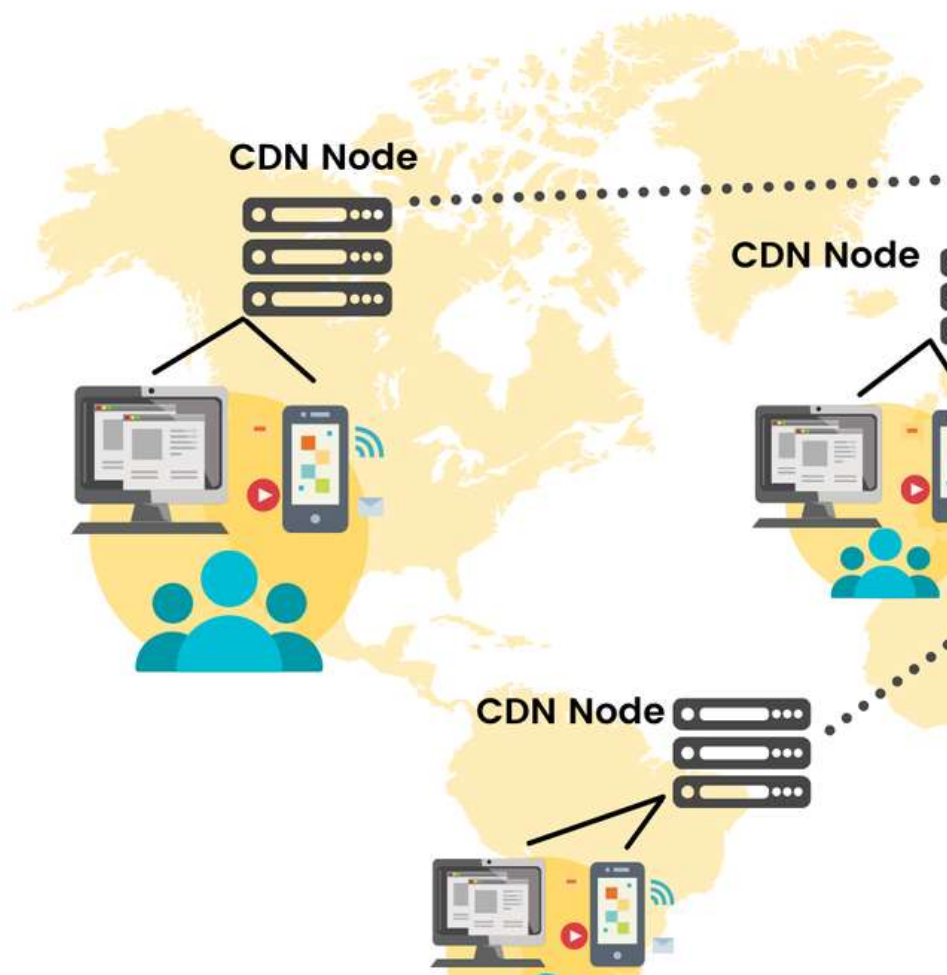
# 11. Design a Payment Gateway

- Design a payment gateway for online payments.
- Key Components: Payment processing, fraud detection.
- Additional: Two-factor authentication.



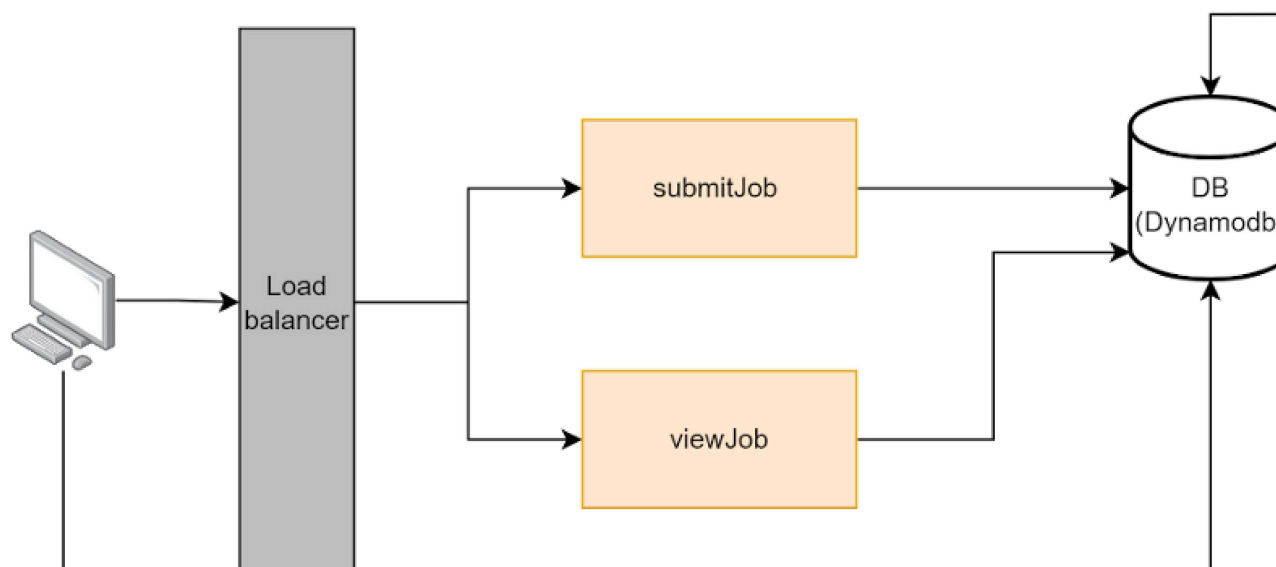
## 12. Design a Content Delivery

- Design a CDN for efficient content
- Key Components: Edge servers, ca
- Additional: Dynamic content cach  
content.



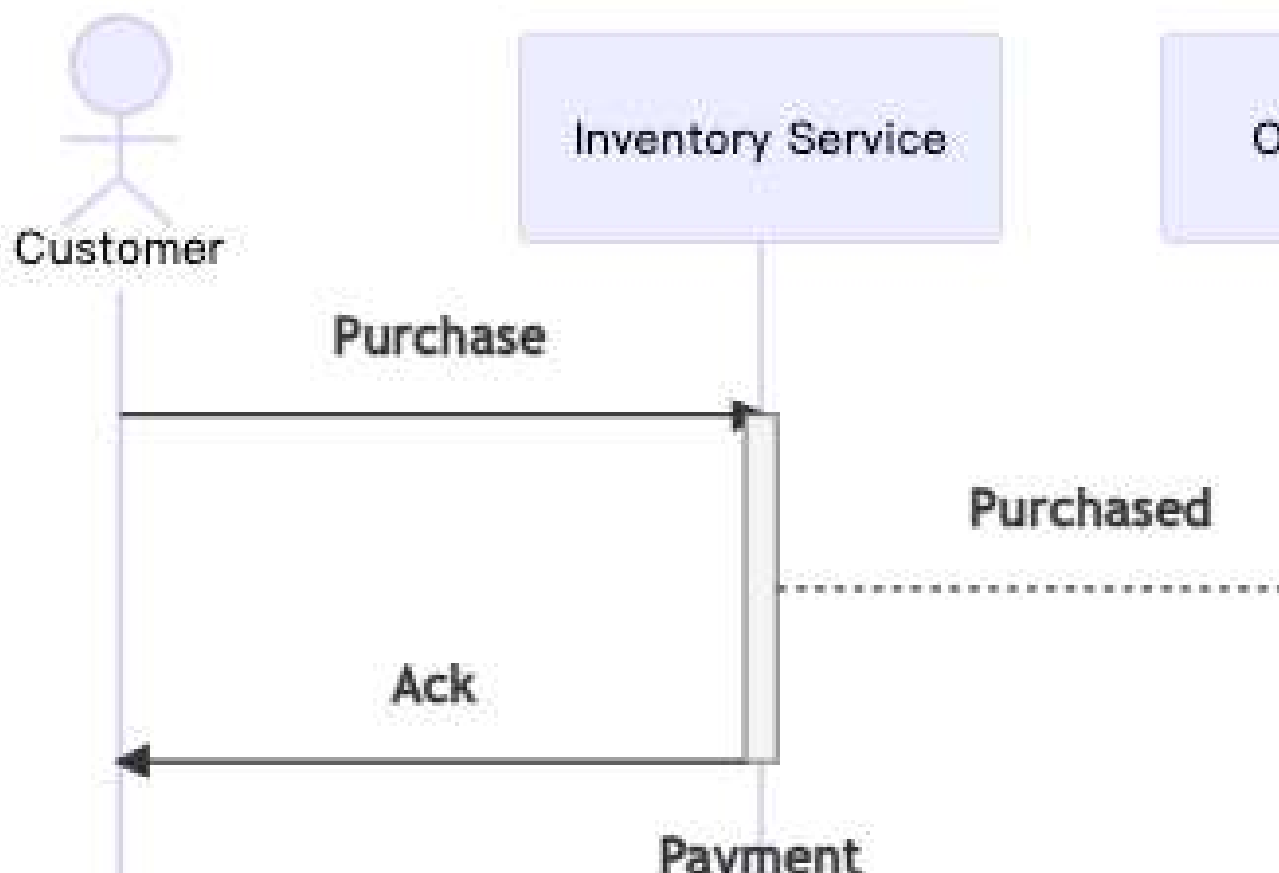
## 13. Design a Job Scheduler:

- Design a job scheduler for a distributed environment.
- Key Components: Job queue, scheduler, fault tolerance.
- Additional: Prioritization of critical jobs.



## 14. Design a Online Marketplace

- Design an online marketplace for
- Key Components: Listings, search, processing.
- Additional: Integrated customer s

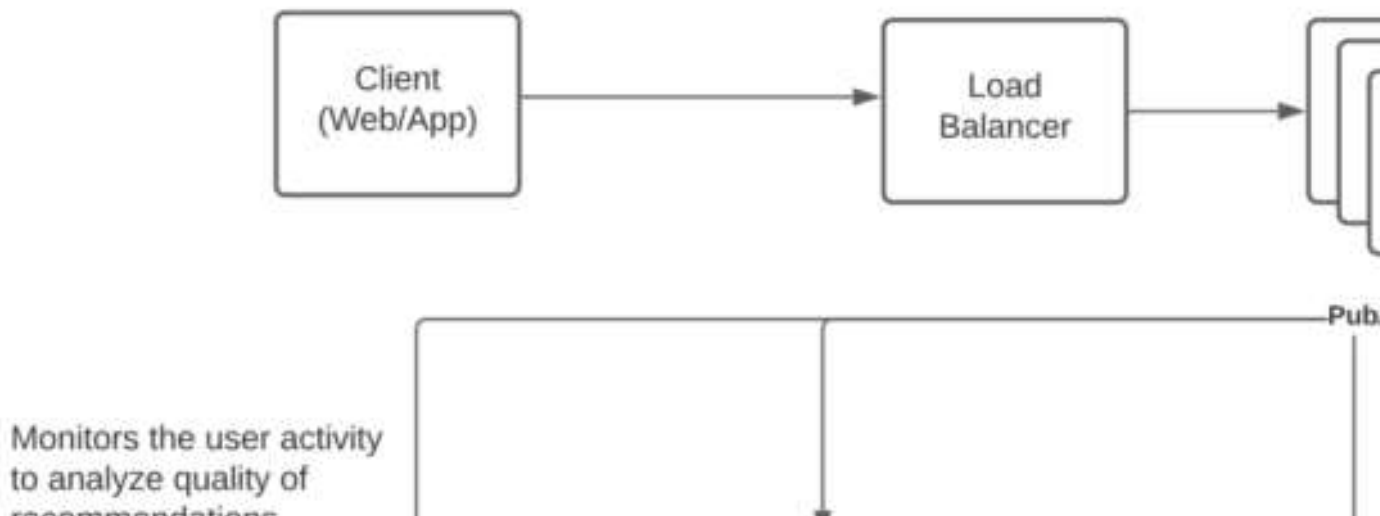




# 15. Design a Recommendation System

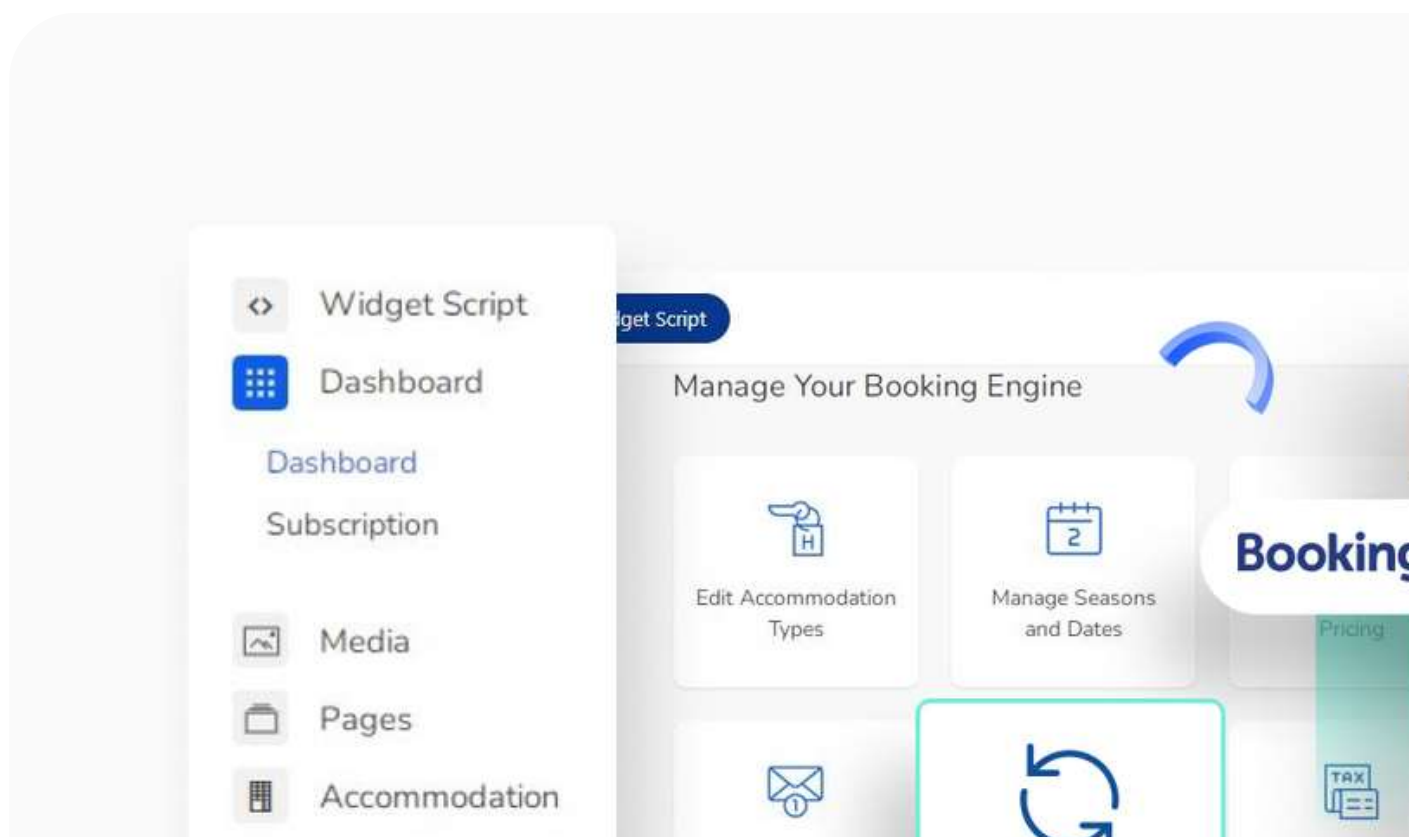
- Design a recommendation system
- Key Components: Collaborative filtering, personalization.
- Additional: Continuous learning for user preferences.

## Recommendation System Architecture



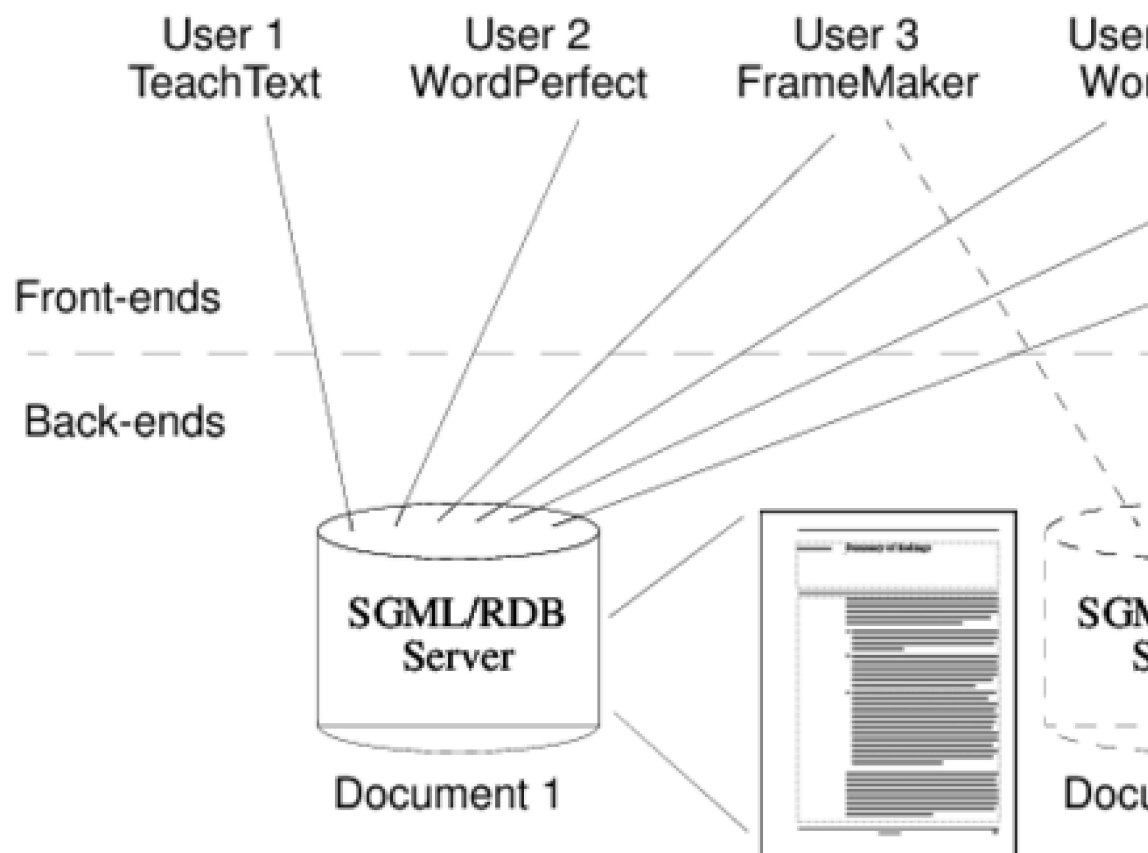
## 16. Design a Hotel Reservation

- Design a system for hotel room re
- Key Components: Booking engine, reservation system.
- Additional: Integration with external



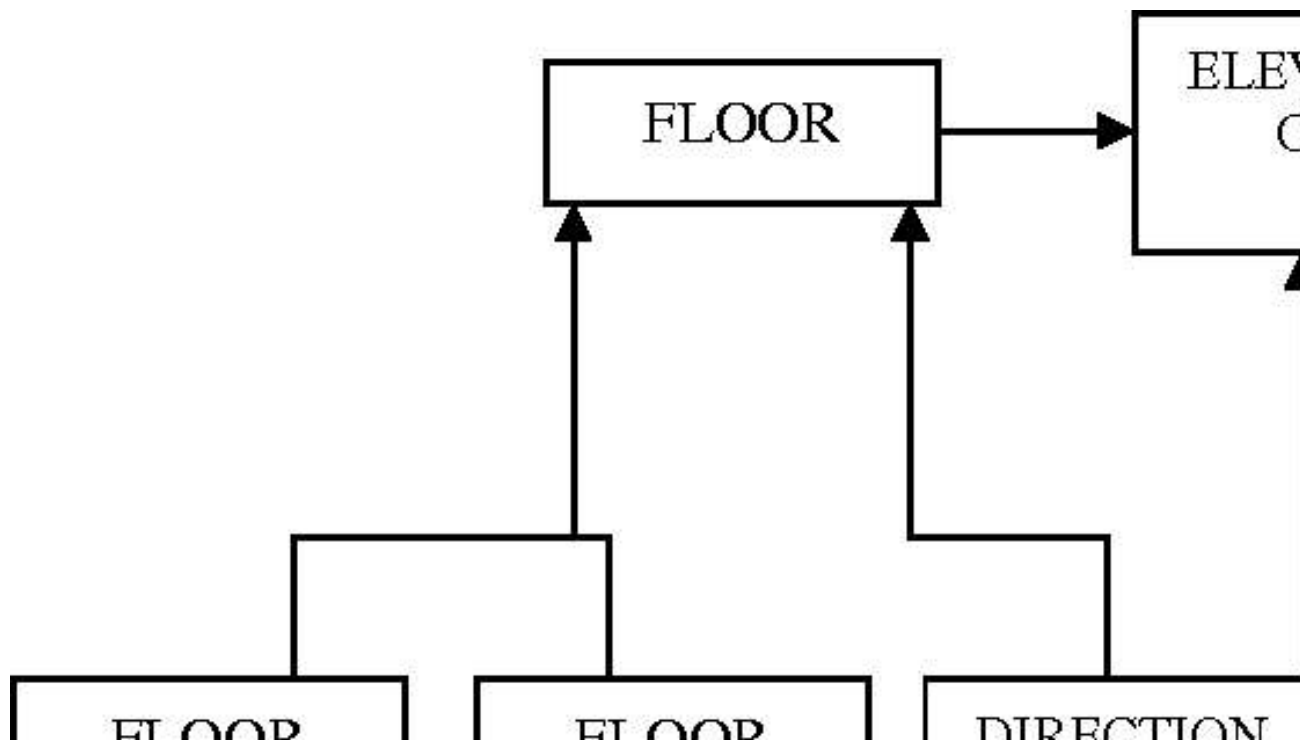
## 17. Design a Collaborative E

- Design a collaborative text editing
- Key Components: Operational tra  
synchronization.
- Additional: Version control for col



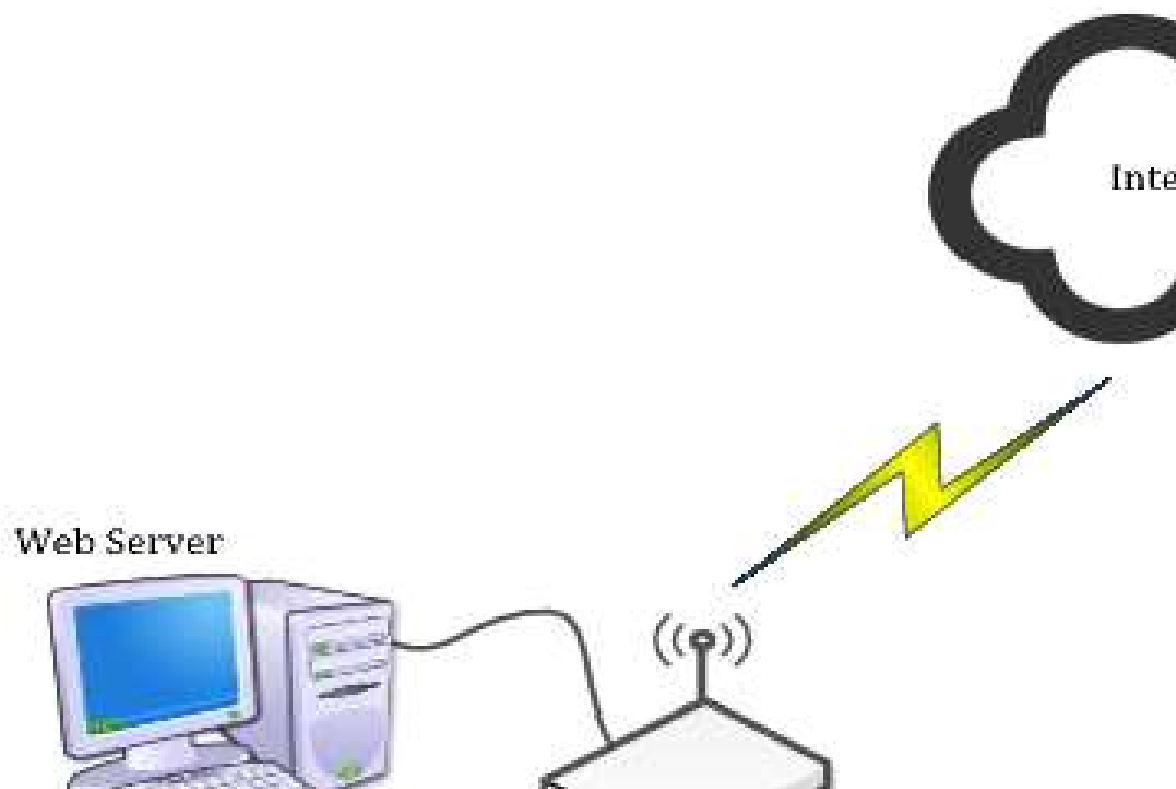
## 18. Design a Elevator System

- Design a control system for a multi-story building.
- Key Components: Scheduling algorithm, floor sensor, door control, and emergency stop.
- Additional: Emergency evacuation protocol.



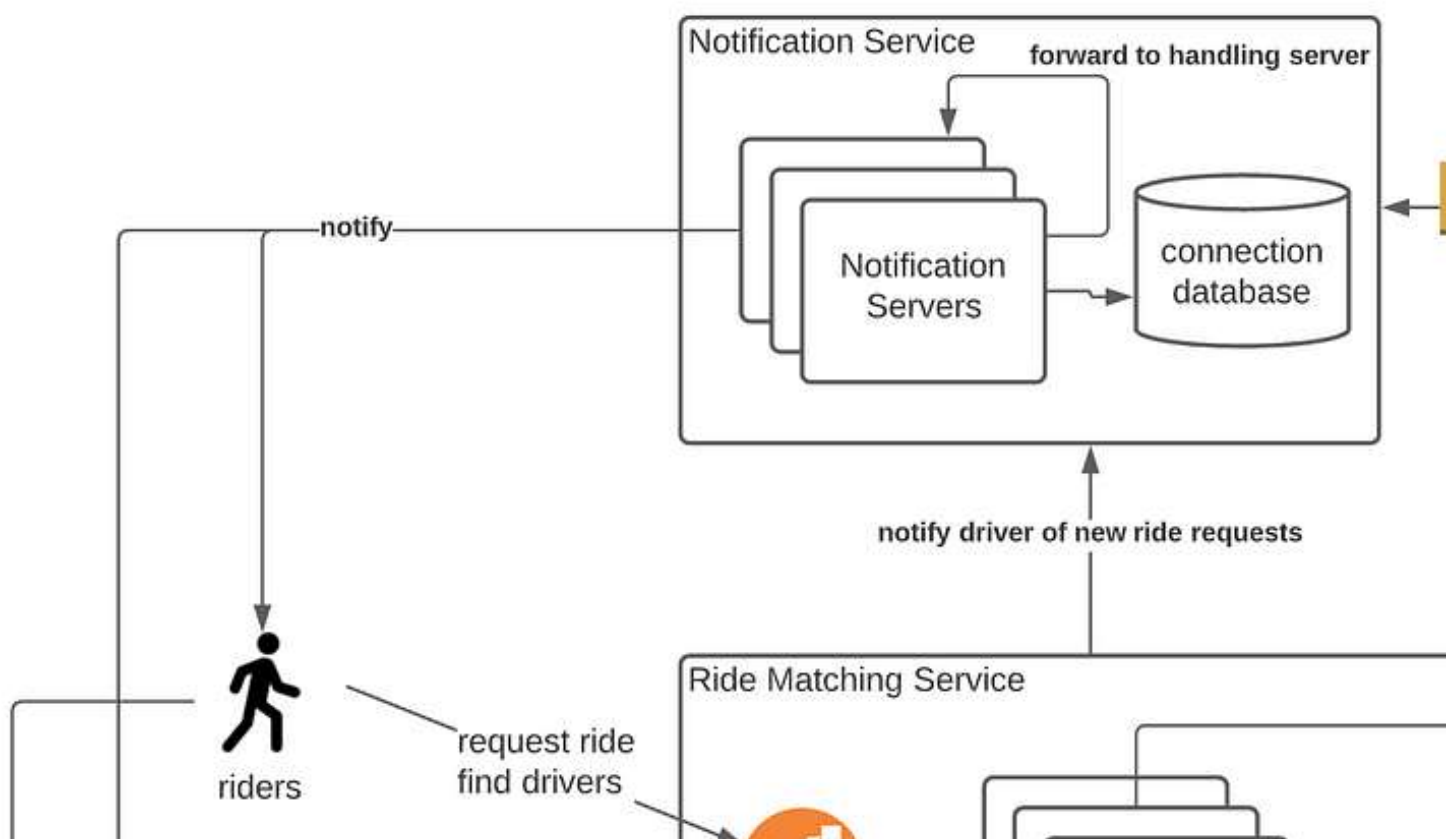
## 19. Design a Weather Service

- Design a system for providing weather updates.
- Key Components: Data acquisition
- Additional: Historical weather data



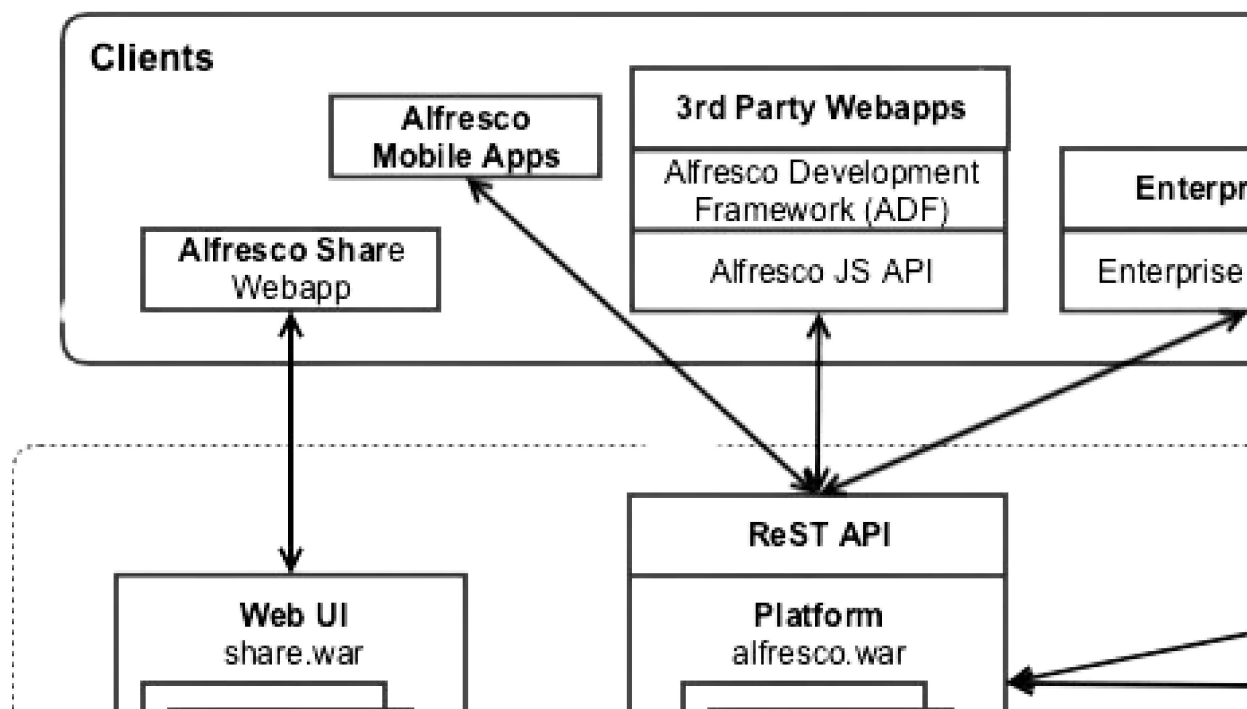
## 20. Design a Ride-Sharing System

- Design a system for a ride-sharing
- Key Components: Matching algorithm
- Additional: Dynamic pricing based



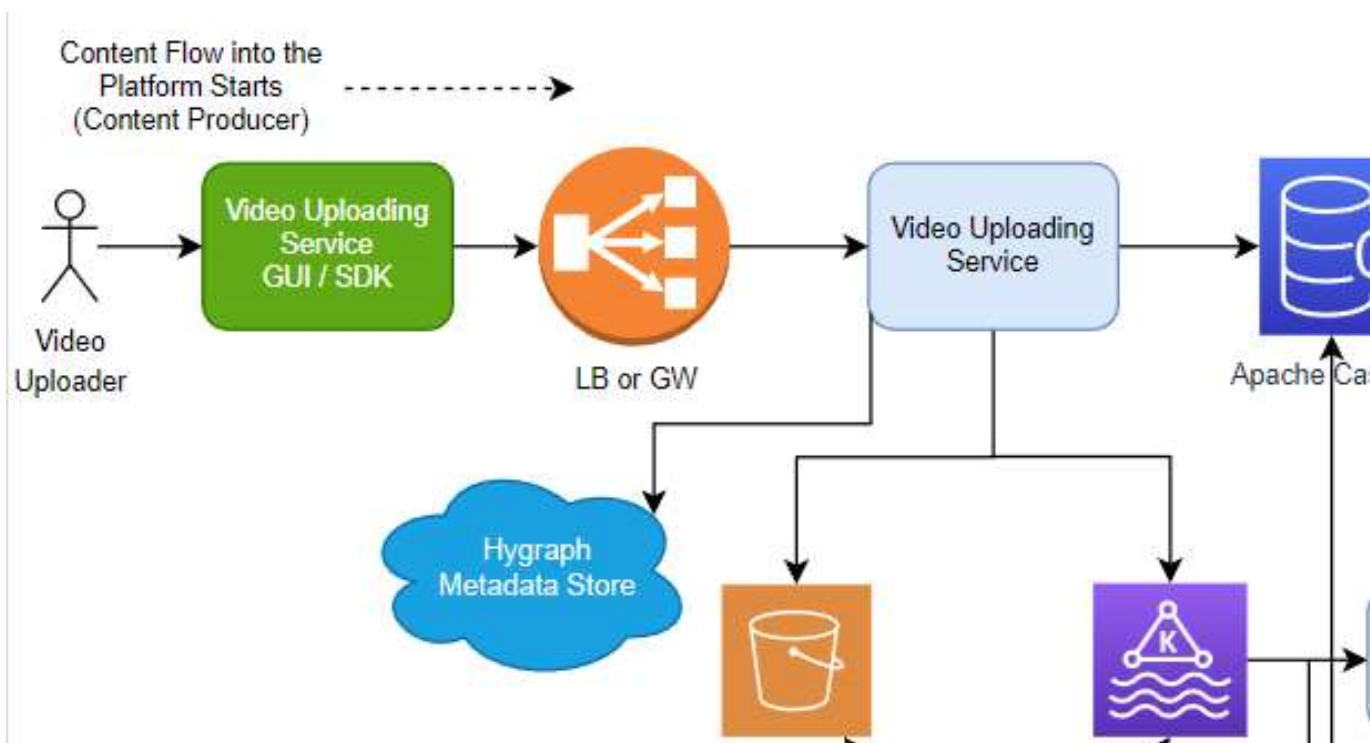
## 21. Design a Document Man

- Design a system for storing, retrieving documents.
- Key Components: Document storage
- Additional: Versioning for documents



## 22. Design a Video Streaming

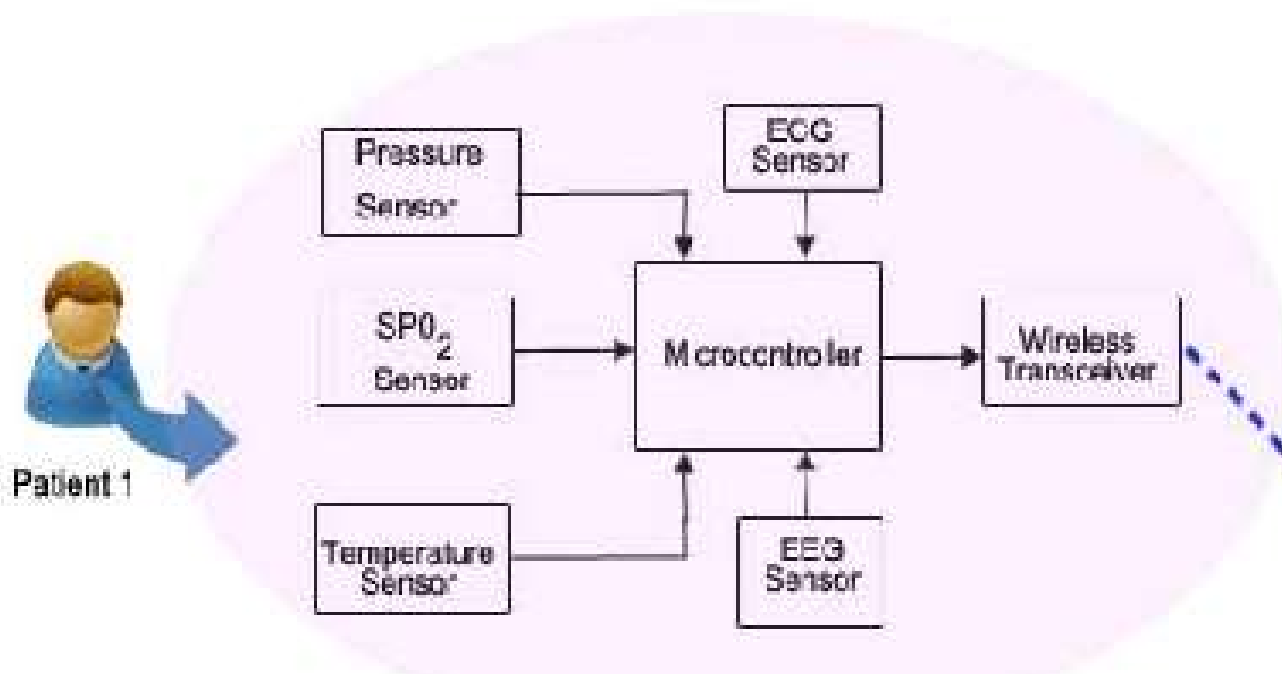
- Design a scalable video streaming content.
- Key Components: Content delivery
- Additional: Real-time streaming and





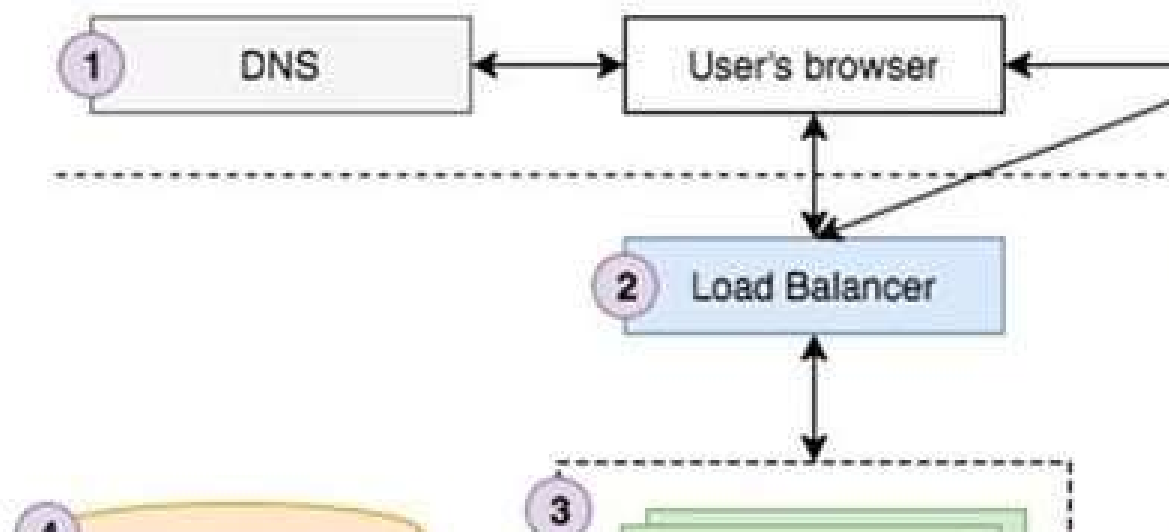
## 23. Design a Health Monitor

- Design a scalable video streaming content.
- Key Components: Content delivery
- Additional: Real-time streaming and



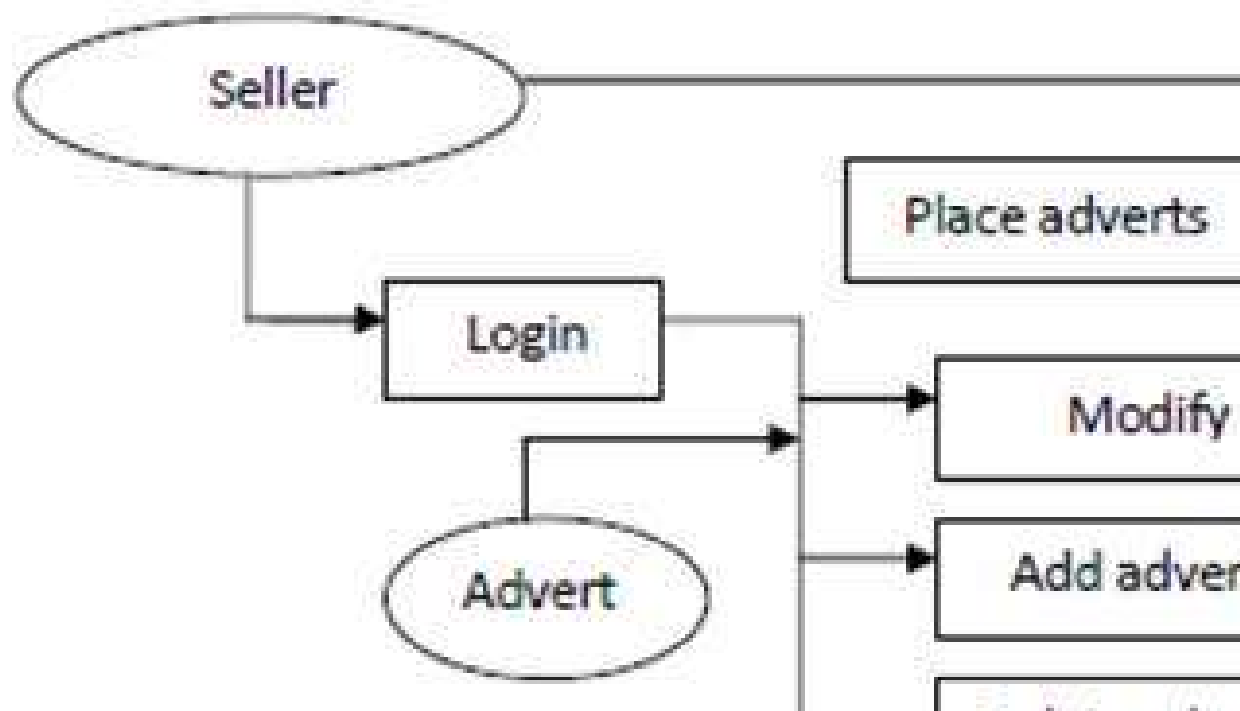
## 24. Design a Education Platform

- Design an online education platform with assessments.
- Key Components: Course content, grading system.
- Additional: Gamification elements



## 25. Design a Auction System

- Design an online auction system for
- Key Components: Bidding engine, payment processing.
- Additional: Anti-sniping measures bidding..





**AlgoT**

---

---

# WHY ALGO



**100% Placement Assistance**



**200+ Successful Alumni**