

1. Architecture & Design Patterns

- Monolithic architecture
- SOA (Service-Oriented Architecture)
- Event-driven architecture
- Hexagonal / Clean architecture
- CQRS (Command Query Responsibility Segregation)
- Saga pattern
- Strangler pattern
- Sidecar pattern
- API Gateway pattern
- Backend for Frontend (BFF)

2. Scalability & Performance

- Auto-scaling
- Elasticity
- Throughput vs Latency
- Capacity planning
- Sharding / Partitioning
- Data replication
- Read replicas
- Hotspot handling
- Rate limiting

- Throttling
- Backpressure
- Load shedding

3. Caching & Data Access

- Caching strategies (Write-through, Write-back, Cache-aside)
- Cache eviction policies (LRU, LFU, FIFO)
- CDN (Content Delivery Network)
- In-memory databases (Redis, Memcached)
- Consistent hashing
- Query optimization
- Indexing strategies

4. Databases & Storage

- SQL vs NoSQL
- CAP theorem
- ACID vs BASE
- Eventual consistency
- Strong consistency
- Multi-region databases
- Time-series databases
- Object storage (S3-like)
- Data lake vs Data warehouse

- Schema design & normalization
- Denormalization

5. Distributed Systems Concepts

- Consensus (Raft, Paxos – high level)
- Leader election
- Clock synchronization
- Distributed locks
- Idempotency
- Exactly-once vs At-least-once delivery
- Message ordering
- Network partitions
- Split brain problem
- Gossip protocols

6. Messaging & Asynchronous Processing

- Message queues (Kafka, RabbitMQ, SQS)
- Pub-Sub model
- Event streaming
- Dead letter queues (DLQ)
- Retry mechanisms
- Idempotent consumers
- Batch processing
- Stream processing

7. Reliability & Availability

- High availability (HA)
- Failover
- Graceful degradation
- Circuit breaker
- Retry with exponential backoff
- Chaos engineering
- Health checks
- Heartbeats
- Disaster recovery (DR)
- RTO / RPO

8. Security & Compliance

- Authentication vs Authorization
- OAuth / JWT
- API security
- Encryption at rest & in transit
- Secrets management
- Role-based access control (RBAC)
- Rate-limit abuse protection
- Audit logs
- Compliance (GDPR, HIPAA basics)

9. Observability & Operations

- Monitoring
- Alerting
- Distributed tracing
- Log aggregation
- Metrics (RED / USE metrics)
- SLAs, SLOs, SLIs
- Blue-green deployment
- Canary releases
- Rollbacks

10. DevOps & Infrastructure

- Containers (Docker)
- Container orchestration (Kubernetes – high level)
- Infrastructure as Code (IaC)
- CI/CD pipelines
- Service discovery
- Configuration management
- Cloud vs On-prem
- Multi-cloud strategy

11. API & Communication

- REST vs GraphQL vs gRPC

- Synchronous vs Asynchronous calls
- API versioning
- Backward compatibility
- Contract testing
- Pagination & filtering
- Timeout handling

12. Trade-offs & Design Thinking (VERY IMPORTANT)

- Consistency vs Availability
- Latency vs Accuracy
- Cost vs Performance
- Simplicity vs Flexibility
- Strong vs Eventual consistency
- Centralized vs Decentralized
- Build vs Buy

13. Interview-Focused System Design Topics

- Design URL shortener
- Design WhatsApp / Chat system
- Design Netflix / YouTube
- Design Uber
- Design Notification system
- Design Rate limiter

- Design Search autocomplete
- Design Logging system
- Design Payment system

How to use this list

- **For students:** Teach by category, not randomly
- **For interviews:** Focus on *trade-offs + scalability + failure handling*
- **For projects:** Pick 5–6 topics and apply them deeply