Product Architect — Self-Practice Technical Interview Guide

1■■ Architecture & System Design

Q1. How would you design a scalable microservice architecture for 1B+ records/day?
■ Hint: Use event-driven microservices, Kafka/RabbitMQ, horizontal scaling, and eventual consistency. Separate ingestion, processing, and storage layers.

Q2. What's the difference between monolithic and microservice architecture?
■ Hint: Monolith = tightly coupled, single deployable unit; Microservice = independent deployables with API contracts.

Q3. Explain API Gateway & Composite API.
■ Hint: Gateway manages authentication/routing; Composite API aggregates responses from multiple microservices for a single client request.

2■■ Cloud & DevOps

Q1. How do you design Kubernetes deployment for cost and scale?
■ Hint: Use node autoscaling, resource requests/limits, HPA, and efficient container images.

Q2. How would you integrate observability using Grafana & OpenTelemetry?
■ Hint: Collect metrics, traces, and logs; visualize in Grafana dashboards with Prometheus back-end.

Q3. How do you manage secrets in production?
■ Hint: Use HashiCorp Vault or K8s secrets; never hardcode credentials or store in Git.

3■■ Performance Optimization

Q1. How do you find bottlenecks in high-throughput systems?
■ Hint: Use profiling tools, distributed tracing, async I/O, and caching with Redis.

Q2. What are best practices for optimizing .NET microservices?
■ Hint: Pool DB connections, use async/await, reduce GC pressure, and enable metrics for latency.

Q3. How would you benchmark performance?
■ Hint: Use BenchmarkDotNet or k6 for API testing; monitor throughput, latency, and CPU utilization.

4■■ Data Architecture

Q1. SQL vs NoSQL — when to choose which?
■ Hint: SQL = consistency & transactions; NoSQL = scalability & flexibility.

Q2. How to handle 80M+ records in MySQL efficiently?
■ Hint: Use partitioning, indexing, and read replicas.

Q3. How to compress or archive large data volumes?
■ Hint: Use gzip, columnar storage, or time-based retention policies.

5■■ Security & Governance

Q1. How would you implement SSO across multiple clients?
■ Hint: Use OAuth2/OpenID Connect with IdentityServer; issue tokens validated by APIs.

Q2. What is RBAC and how to enforce it?
■ Hint: Define roles, permissions, and claim-based access checks in API layer.

Q3. How do you integrate SAST in CI/CD?
■ Hint: Automate scanning with SonarQube/Kiuwan before deployment.

6■■ Leadership & Product Thinking

Q1. How do you balance technical debt vs product delivery?
■ Hint: Maintain architecture runway; prioritize refactoring aligned with business milestones.

Q2. How do you communicate architecture to non-technical stakeholders?
■ Hint: Use diagrams, metrics, and business outcomes instead of technical jargon.

Q3. How do you mentor teams on architecture decisions?
■ Hint: Run architecture reviews, code walkthroughs, and share design patterns.

7■■ Scenario-Based Questions

Q1. Throughput of mediation service drops by 40% — how to troubleshoot?
■ Hint: Check queue backlog, DB latency, Redis health, and thread starvation in services.

Q2. You need to enable multi-tenant support — what changes are required?
■ Hint: Add tenant_id in schema, isolate cache keys, and ensure tenant-based routing & config.

Q3. A client requests real-time fraud detection — how to add it?
■ Hint: Stream data to ML model API; use event streaming or async microservice for detection.