Technical Phone Screen — DevOps/SRE

Phone Screen | DevOps/SRE

Evaluate infrastructure skills: CI/CD, containers, monitoring, and incident response.

Evaluation Criteria

CI/CD Pipeline Design

Evaluates experience designing and maintaining continuous integration and deployment pipelines.

Rating: □ 1 □ 2 □ 3 □ 4 □ 5

Sample Questions:

- Walk me through a CI/CD pipeline you've built — what stages does it include and why?
- How do you handle deployment rollbacks?
- What's your approach to managing environment-specific configuration across dev, staging, and prod?

✓ STRONG SIGNAL

Candidate describes pipelines with build, test, security scanning, and deployment stages. They discuss blue-green or canary deployments, rollback strategies, and environment promotion patterns.

✕ WEAK SIGNAL

Candidate has only used basic push-to-deploy setups, cannot explain how to roll back a bad deployment, or has no strategy for environment management.

Containers & Orchestration

Assesses knowledge of containerization concepts and orchestration platforms like Kubernetes.

Rating: □ 1 □ 2 □ 3 □ 4 □ 5

Sample Questions:

- How do you write a Dockerfile for a production service — what best practices do you follow?
- Explain how Kubernetes manages scaling and self-healing.
- When would you choose containers vs. serverless, and what are the trade-offs?

✓ STRONG SIGNAL

Candidate discusses multi-stage builds, minimal base images, resource limits, health checks, and liveness/readiness probes. They understand pod scheduling, horizontal autoscaling, and service mesh concepts.

✕ WEAK SIGNAL

Candidate runs everything as root in containers, doesn't understand namespaces or resource limits, or cannot explain basic Kubernetes concepts despite claiming experience.

Monitoring, Alerting & Observability

Evaluates the candidate's approach to building observable systems and meaningful alerting.

Rating: □ 1 □ 2 □ 3 □ 4 □ 5

Sample Questions:

- What metrics do you consider essential for monitoring a production service?
- How do you distinguish between meaningful alerts and noise?

- Describe your ideal observability stack and why you'd choose those tools.

✓ STRONG SIGNAL

Candidate references the four golden signals (latency, traffic, errors, saturation) or RED/USE methods. They discuss SLOs/SLIs, alert fatigue prevention, and correlating metrics with logs and traces.

✕ WEAK SIGNAL

Candidate monitors only CPU and memory, has no alerting strategy, or cannot explain the difference between monitoring, logging, and tracing.

Incident Response & Reliability

Assesses experience with production incidents, on-call responsibilities, and post-incident processes.

Rating: □ 1 □ 2 □ 3 □ 4 □ 5

Sample Questions:

- Walk me through how you'd respond to a production outage from first alert to resolution.
- How do you write an effective postmortem?
- What is your approach to setting and managing SLOs?

✓ STRONG SIGNAL

Candidate describes a structured incident response process: triage, communicate, mitigate, root cause, postmortem. They emphasize blameless culture and action items that prevent recurrence.

✕ WEAK SIGNAL

Candidate has never been on-call, views incidents as someone else's problem, or focuses only on blame rather than systemic improvement.

Infrastructure as Code & Automation

Evaluates experience with IaC tools, automation patterns, and managing infrastructure at scale.

Rating: □ 1 □ 2 □ 3 □ 4 □ 5

Sample Questions:

- What IaC tools have you used, and how do you manage state and drift?
- How do you handle secrets management in your infrastructure code?
- Describe a complex piece of infrastructure you automated.

✓ STRONG SIGNAL

Candidate has hands-on experience with Terraform, Pulumi, or CloudFormation. They discuss state management, module design, code review for infra changes, and drift detection strategies.

✕ WEAK SIGNAL

Candidate makes infrastructure changes manually via cloud consoles, doesn't version control infrastructure, or has only used IaC for trivial setups.

Red Flags

- Makes production changes without version control or peer review
- No concept of infrastructure as code — manages everything via cloud console clicks
- Cannot explain basic networking concepts like DNS, load balancing, or TLS

Notes & Overall Recommendation

□ Strong Hire □ Hire □ No Hire □ Strong No Hire

Notes: _____