**Frontend Developer Interview Scorecard**

Candidate Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Role Interviewed For: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Interviewer: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| **Dimension** | **Guidance** | **Score (1–5)** |
| System Design | 1-2: Cannot decompose systems; ignores trade-offs; produces brittle designs.
3: Creates reasonable designs for moderate scope and recognizes common trade-offs.
4: Designs scalable, modular systems and anticipates failure modes and scaling needs.
5: Defines long-term architecture, drives cross-team design decisions, and balances extensibility, cost, and performance. |  |
| Backend Implementation | 1-2: Produces buggy or inefficient server code; misses edge cases and error handling.
3: Implements reliable, maintainable backend features with appropriate abstractions.
4: Delivers efficient, well-tested services with clear APIs and robust error handling.
5: Optimizes throughput and latency, mentors others on backend patterns, and leads complex refactors. |  |
| Frontend Implementation | 1-2: UI is non-functional or inaccessible and misuses framework patterns.
3: Builds responsive, maintainable UI following component patterns and basic accessibility.
4: Creates performant, testable components with clear state management and UX polish.
5: Shapes frontend architecture, improves developer experience, and drives performance and accessibility standards. |  |
| Testing and Quality | 1-2: No automated tests; relies on manual testing and introduces regressions.
3: Delivers unit and integration tests covering key paths and uses CI.
4: Maintains high test coverage, effective mocks, and deterministic tests while reducing flakiness.
5: Establishes testing strategy, fosters team test ownership, and improves CI reliability. |  |
| DevOps & Reliability | 1-2: Deploys manually with frequent failures and lacks monitoring knowledge.
3: Uses CI/CD, can deploy and rollback, and monitors basic health metrics.
4: Implements robust deployment strategies, automation, alerts, and recovery steps.
5: Designs SLOs/SLIs, runs incident postmortems, and automates runbooks and capacity planning. |  |
| Collaboration & Communication | 1-2: Fails to communicate status, is unresponsive to feedback, and blocks the team.
3: Communicates clearly with engineers and product and participates in planning and reviews.
4: Influences technical decisions, negotiates trade-offs, and aligns stakeholders.
5: Leads cross-team initiatives, mentors peers, and communicates complex trade-offs to leadership. |  |
| Ownership & Mentorship | 1-2: Avoids ownership, repeats issues, and gives minimal code review feedback.
3: Owns features end-to-end and provides helpful review feedback.
4: Proactively improves the codebase, mentors juniors, and reduces technical debt.
5: Defines team technical direction, scales processes, and grows others into senior roles. |  |

 **Overall Evaluation**

**Strengths Observed:**

**Concerns / Weaknesses:**

**Recommendation (Yes / No / With Reservations):**

**Final Score (Avg / Weighted):**