**Frontend Developer Interview Scorecard**

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

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

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

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

|  |  |  |
| --- | --- | --- |
| **Dimension** | **Guidance** | **Score (1–5)** |
| HTML & CSS Fundamentals | 1-2: Produces incorrect or non-semantic markup and frequently breaks layout.
3: Creates functional, semantic markup and resolves common layout issues with guidance.
4: Builds responsive, maintainable CSS and avoids layout regressions independently.
5: Designs reusable styling patterns and improves team CSS practices proactively. |  |
| JavaScript Fundamentals | 1-2: Struggles with basic language constructs and common DOM interaction patterns.
3: Understands ES6+ syntax and implements straightforward logic reliably.
4: Writes clear, idiomatic JS and handles async flows and edge cases independently.
5: Optimizes logic for performance and mentors peers on JavaScript best practices. |  |
| Framework & Component Experience | 1-2: Cannot create or reason about components in the team's framework.
3: Builds and composes components following basic patterns with some guidance.
4: Implements well-structured components, state management, and props flows independently.
5: Introduces improved component patterns and helps reduce complexity across the codebase. |  |
| Problem Solving & Debugging | 1-2: Gets stuck frequently and cannot isolate root causes of bugs.
3: Diagnoses common issues using console, devtools, and error messages.
4: Systematically isolates problems, writes reproducible steps, and proposes fixes.
5: Identifies underlying patterns in bugs and prevents recurrence proactively. |  |
| Code Quality & Testing | 1-2: Produces untested code with inconsistent style and unclear intent.
3: Writes readable code and adds basic unit or integration tests when prompted.
4: Delivers well-structured code with reliable tests and follows linting/formatting rules.
5: Improves test coverage, suggests meaningful test cases, and enforces quality standards. |  |
| UX & Accessibility | 1-2: Ignores basic accessibility and produces unclear user interactions.
3: Implements common UX patterns and basic ARIA/keyboard support when guided.
4: Delivers accessible components and considers error states and focus management.
5: Advocates accessibility improvements and designs interactions that reduce user friction. |  |
| Collaboration & Communication | 1-2: Communicates poorly, misses context, and requires frequent follow-up.
3: Asks relevant questions, documents work, and responds to feedback constructively.
4: Proactively shares progress, aligns with teammates, and incorporates feedback quickly.
5: Facilitates small design or code discussions and helps coordinate across roles. |  |
| Learning & Growth Mindset | 1-2: Resists feedback and shows little progress after coaching.
3: Accepts feedback and applies it to improve tasks over time.
4: Seeks new challenges, learns tools quickly, and applies knowledge independently.
5: Drives personal improvement plans and shares learnings to uplift peers. |  |

 **Overall Evaluation**

**Strengths Observed:**

**Concerns / Weaknesses:**

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

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