CoderPad

CODERPAD SCREEN | AI PROFICIENCY SCORECARD — JUNIOR / MID

CODERPAD SCREEN · AI-AWARE ASSESSMENT

Score AI proficiency, not just AI usage.

Score each criterion 1–5. Total range: 5–25. Ownership (C) ≤ 2 = no hire regardless of other scores. The best candidates don't use AI more — they use it better. This scorecard separates high-leverage AI use from dependency.

SCORING CRITERIA

CRITERIA	SCORE /5	SCORE 1–2 · WEAK	SCORE 3–4 · STRONG	SCORE 5 · ADVANCED	KEY QUESTION
AI Strategy & Intent *Why and when do they use AI?*	☐	*Uses AI reflexively, no plan. AI is the default problem-solver. No stated rationale for when to prompt.*	Uses AI with clear, stated intent. "I'll get a baseline and sanity-check it." Uses AI deliberately to accelerate.	**Uses AI to critique own thinking. Anticipates when AI will mislead. Fluidly shifts between modes.**	*"What did you want AI to do here — and was that the right call?"*
Prompting & Framing *How do they frame the request?*	☐	*"Solve this" — minimal context. No constraints or expectations given. Accepts first output without iteration.*	Provides constraints + acceptance criteria. Iterates prompts to improve output. Focused on control over the output.	**Designs prompts strategically. Optimizes for token efficiency. Creates reusable prompt patterns.**	*"Walk me through how you decided what to ask the AI."*
⚠ Most Important **Ownership & Explanation** *Can they explain the solution?*	☐	*Cannot explain the generated code. Reads output line-by-line. No synthesis; hesitates on "why does this work?"*	Explains trade-offs and design reasoning. Answers "why" questions confidently. Identifies inefficiencies unprompted.	**Critiques and improves the approach. Spots risks the AI introduced. Explains as if teaching someone else.**	*"Explain this in your own words. What's the biggest assumption here?"*
Critical Evaluation *Do they trust blindly?*	☐	*Copy–paste and pray. Notices issues only when they break. Doesn't test before submitting.*	Proactively adjusts or rewrites weak parts. Tests and fixes errors methodically. Rejects suggestions with clear reasoning.	**Rewrites parts of solution confidently. Catches subtle edge cases unprompted. "I don't like how it handles this — I'll refactor."**	*"Where could this break?" / "Would you change anything?"*
Integration into Workflow *Does AI accelerate or derail?*	☐	*Gets stuck debugging AI output. AI adds confusion, slows progress. Can't recover when AI is wrong.*	AI clearly speeds up progress. Recovers quickly from bad AI output. Confident, fluid use throughout.	**Seamlessly embedded in workflow. Visible progress within minutes of AI use. Knows exactly when to stop using AI.**	*"If you couldn't use AI, how would your approach differ?"*
TOTAL	☐ /25	**Ownership ≤ 2 = no hire regardless of total · Advanced (score 5) is rare at this level**			

SCORING BANDS · HIRE DECISION · INTERVIEWER REFERENCE

SCORE	LEVEL	INTERPRETATION
5–10	Basic	AI as answer engine, limited independent reasoning
11–17	Effective	Strong engineer, AI used competently as an assistant
18–25	Advanced	High-leverage AI use with strong independent judgment

SIGNAL	CRITERIA
Strong Hire	Effective or above (≥11) AND Ownership ≥ 4
Weak Signal	Ownership ≤ 2 — no hire regardless of total
Advanced	Rare at this level; indicates high AI leverage
Escalate	"What would you change if performance mattered?" / "What would you have tried without AI?"

PROBING QUESTIONS TO USE MID-SCREEN

- "What do you think of the AI's output here?"
- "Can you explain what this code does and why it works?"
- "Why did you accept — or reject — that suggestion?"
- "Which model did you use, and why that one?"
- "What would you do differently with more time?"
- "Walk me through how payment validation works across these files."

Task Types & What They Reveal

- **Debug** — Juniors jump to fixing; seniors understand root cause first
- **Build / extend** — Watch how they break the problem down with AI
- **Optimize / refactor** — Reveals strategic thinking and trade-off awareness
- **Code review** — Separates those who understand from those just following AI
- **Explain** — Ask mid-task to check comprehension, not just execution

WHAT THE DATA TELLS US

- Candidates who prompt more (multi-turn) pass at **10%+ higher rates** per 1,000 interviews
- Selecting a **specific model provider** correlates strongly with better performance
- Only **30% of candidates** set collaboration terms unprompted — it's a top differentiator
- The best candidates use both **augmentative** (real-time) and **delegative** (async) modes
- Final code alone doesn't tell the story — **the process matters as much as the output**