

The brief and the marking sheet

A take-home earns its place only with 5 rules written into it, a conversation after it, and 3 checks a quarter. Most teams run it with none of those. Here is the version that is defensible.

1. The brief the candidate receives

[Role] take-home · [Company]

Time: 3 hours. We mean it. We have timed this task with our own engineers and we look at what you built in 3 hours, not at what you could build in 10. Please stop at 3 and tell us what you skipped.

The task: [A small service with one real requirement, or an existing small codebase with one subtle bug. One paragraph. Something a person can scope, not a spec.]

What we look at: how the code reads (structure, naming, tests), and what you chose to leave out and why. Scope judgement is scored.

Tools: anything you use at work, including AI assistants. We will talk through your decisions with you afterwards.

What happens next: a 30-minute conversation with [name], within [3] working days of your submission, about the code. Then a decision, in writing, within 2 days of that.

This task is not used by us for anything. It does not resemble our roadmap.

2. The 5 rules, and how the brief meets them

RULE	IN THE BRIEF	SKIP IT AND
Cap it at 3 hours and mean it	The time line, and "tell us what you skipped"	A task you believe is 3 hours is usually 6, and you select for free evenings.
Make it look like the job	The task paragraph	A toy problem measures toy-problem skill.
Never use the output	The last line	Free work gets discussed publicly, with your name on it.
Always follow it with a conversation	"What happens next"	You cannot verify authorship, and the signal is in the conversation anyway.
Put it after a human has spoken to them	Sent only after the screen	A stranger sent a link will not spend the 3 hours.

3. The 30-minute conversation

1. "Walk me through the structure. Why this shape?" (5 min)

2. **"What did you leave out, and how did you decide?"** The most useful signal in the exercise. (5 min)
3. **"Pick a decision on line [N]. Why that?"** Someone who did not make the decision runs out inside 2 follow-ups. (10 min)
4. **"What did the tool suggest that you rejected?"** (5 min)
5. **"With 3 more hours, what first?"** (5 min)

4. The marking sheet

Two markers, blind, before the conversation. Same 4 lines for every candidate at the level.

LINE	STRONG	MIXED	WEAK	MARKER A	MARKER B
Reads well	Structure explains itself; names say what things are; a stranger could change it	Works, needs the author to explain it	One file, one function, or names that hide what happens		
Tests	Tests the edge case the candidate named themselves	Tests the happy path	No tests, or tests that assert nothing		
Scope judgement	Cut the right things and said so; the 3 hours went to the requirement	Built the requirement plus decoration	Ran out of time on the part that mattered		
Decisions defended (after the conversation)	Explains every decision asked about, including the ones they would now change	Explains most; vague on one	Cannot explain code they submitted		

5. The 3 checks, once a quarter

- ☐ **Completion rate by stage.** Sent: _____ Submitted: _____
If a meaningful share go quiet, you are losing people and you do not know which ones.
- ☐ **Correlation with the outcome.** For the last 10 hires, did the take-home score predict anything about month 6? If not, cut the stage this quarter.
- ☐ **Marking spread.** Two markers, 3 submissions, blind. A wide spread means you are measuring the marker. Almost nobody runs this check.

What usually beats it: a 60-minute pair on a small prepared codebase, all tools allowed. One hour for the candidate instead of 6, one hour for you, no marking, and authorship answered for free. Keep the take-home for the one case where writing alone, over hours, is central to the job.