

Interview scorecard

A one-page interview scorecard with anchored 1–5 ratings, a column for the candidate's own words, and worked examples for three roles.

- Keep the three to six competencies that matter most for the role and delete the rest.
- Score each competency straight after the interview, before you discuss the candidate with anyone.
- A 2 or a 4 means the answer fell between two anchors. No evidence, no score.
- Write your overall recommendation last, separately from the scores.

Competency	What good looks like	1 – Weak	3 – Solid	5 – Strong	Score (1–5)	Evidence (their words)
First-round screen (any role)						
Communication	Answers the question asked, clearly and concisely, and gives specifics without being pushed.	Drifts off the question or talks in generalities; you can't tell what they actually did.	Clear answers with a point, but needs a prompt to get to specifics.	Structured, specific answers that get to the point and back it with an example, unprompted.		
Relevant experience	Has done the core tasks of this job, at a similar scale or level.	Experience is adjacent, or described only as job titles and duties.	Has done the core tasks; examples are real but light on scale or results.	Has done the core tasks at a similar or greater scale and describes concrete results: numbers, outcomes, what changed.		
Problem solving	Works out the cause of a problem and fixes it, rather than working around it.	Describes the problem but not how they worked it out, or the fix came from someone else.	A real example with a sensible approach and a working fix; little on the root cause.	Finds the root cause, weighs options, fixes it, checks the fix held, and says what they would do differently.		
Motivation for the role	Wants this job for reasons the job can actually satisfy.	Reasons that would fit any job; knows little about this one.	Clear reasons tied to the work or the role, and some knowledge of the company.	Specific reasons tied to the work, the team or their next career step, backed by real homework on the company.		
Ownership	Treats results as theirs: follows through, flags problems early, fixes what they broke.	Talks only about what 'we' did, or blames others for what went wrong.	An example of seeing a task through and raising a problem when it came up.	Took on a problem nobody owned, saw it through and can point to the result; owns mistakes plainly.		
Example: customer service representative						
De-escalation	Calms an upset customer and still solves the problem.	No real example, or the story ends with a transfer, a raised voice or blaming the customer.	A real example: stayed calm, acknowledged the problem and fixed it within policy.	A specific example with what the customer said and what they said back, a fair fix within policy, the outcome, and what they changed afterward.		
Resolving the issue	Finds the real problem behind a complaint and resolves it on the first contact where possible.	Follows the script without finding the cause, or escalates by default.	Asks questions to find the issue and resolves common problems without help.	Diagnoses an unusual problem from the account or order history, resolves it or hands it off with full notes, and follows up.		

Competency	What good looks like	1 – Weak	3 – Solid	5 – Strong	Score (1–5)	Evidence (their words)
Clear communication	Explains policies and next steps in plain words, by phone, chat or email.	Leans on jargon or policy language; the customer would still be unsure what happens next.	Explains clearly; written examples are accurate but generic.	Adjusts tone and detail to the customer, confirms they understood, and puts the next steps in writing.		
Accuracy	Gets account details, orders and notes right under time pressure.	Can't describe how they avoid errors, or treats mistakes as normal at high volume.	Has a routine for checking details, such as reading back an order or confirming an address.	Gives an example of catching an error others missed and explains the habit that caught it.		
Example: registered nurse						
Clinical judgment	Notices when a patient's condition changes and escalates it promptly.	Describes tasks rather than judgment; no example of noticing a change or escalating.	A real example of noticing a change in a patient and escalating to the charge nurse or provider.	A specific example: what they noticed and why it mattered, how they escalated (for example with SBAR), what they did while waiting, and the outcome.		
Medication safety	Follows the checks that prevent medication errors every time, including on busy shifts.	Can't describe their checks, or describes skipping them under time pressure as normal.	Describes standard checks (patient identity, order, dose, allergies) and uses them consistently.	Stopped an error before it reached the patient, and explains who they told, how it was reported and what changed.		
Communication and handoff	Gives clear handoffs and explains care to patients and families in plain language.	Describes handoff as reading out a list; no example of a patient or family conversation.	Uses a consistent handoff structure and explains care plainly to patients.	An example where a careful handoff or family conversation prevented a problem, including how they checked understanding.		
Prioritization under workload	Organizes care across several patients so the most urgent needs come first.	Says they just work faster; no method for deciding what comes first.	Prioritizes by acuity and time-critical tasks, and asks for help when needed.	A specific shift: how they triaged, what they delegated and to whom, how they kept the charge nurse informed, and how it ended.		
Example: software engineer						
Debugging	Finds the root cause of hard problems methodically.	A generic process, such as adding logging until it works; no real bug described.	A real bug: how they reproduced it, narrowed it down and fixed it.	A hard bug traced by forming and testing hypotheses, a root-cause fix, a test or alert so it can't quietly return, and what they shared with the team.		
Design trade-offs	Makes sensible design choices and can explain what they gave up.	Can't name a trade-off; answers with 'best practice'.	Describes a design choice and at least one alternative they considered.	Names the constraints, compares alternatives by their costs, explains the choice, and says what they would change with hindsight.		

Competency	What good looks like	1 – Weak	3 – Solid	5 – Strong	Score (1–5)	Evidence (their words)
Collaboration and code review	Gives and takes review well, and settles technical disagreements with evidence.	Describes review as a gate or a fight; no example of changing their mind.	Gives specific, respectful review comments and can describe a disagreement they resolved.	Settled a disagreement with data or a prototype, kept the working relationship, and improved a team practice as a result.		
Ownership of delivery	Ships working software and stays responsible for it after release.	Describes work as tickets closed; no sense of what happened after release.	Delivered a feature end to end, flagged risks early and fixed problems after launch.	Owned an outcome beyond their tickets, such as monitoring, a fix nobody owned or an incident follow-up, and can show the effect.		