EVIDENCE FOR EVERYDAY DECISIONS

The Tuesday Test

Choose an AI tool using the work you actually do.



Run a small, repeatable trial. Count corrections, handling time and total cost before choosing a tool.

01 / PRACTICAL FIELDNOTE

Start with an ordinary Tuesday

Two-minute summary: define the job, test it, count the repairs.

INPUT

Approved order notes

OUTPUT

Checkable draft production brief

BOUNDARY

No order placed automatically

A furniture workshop needs to turn rough order notes into a production brief. A brilliant essay or dramatic generated video tells the owner little about whether a tool can preserve dimensions, flag a missing finish and avoid inventing a delivery date.

Write the job in a sentence: “Create a draft production brief from these approved order notes; flag anything missing and do not place an order.” Then define a usable result. The brief must preserve every supplied measurement, separate confirmed choices from questions and be easy for a colleague to check.

Use this issue to design a small trial before making a commitment. The proposed trial is an editorial method, not a validated industry benchmark. It can reveal obvious mismatches; it cannot prove reliability on work you did not test.

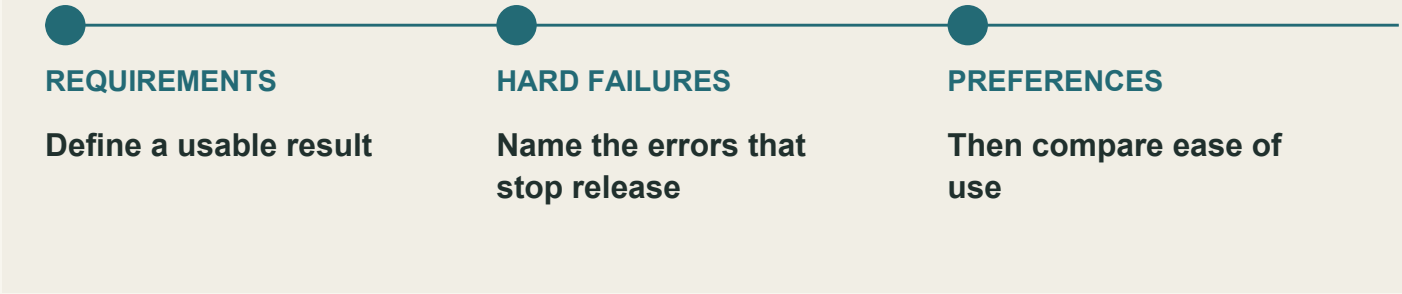
Judge the output you need, with the constraints you actually have.

Fictional workshop. Documented data and source scope are identified in the text.

02 / PRACTICAL FIELDNOTE

Write the acceptance rule first

Decide what failure means before seeing the output.



For the invented production brief, a changed dimension is a hard failure. Attractive formatting cannot compensate. Missing an explicitly supplied finish is also a failure. A correct brief with clumsy headings may be repairable.

Make these distinctions before testing. Otherwise a charming answer quietly changes the criteria. Write a few non-negotiables, then a small set of preferences such as readable structure, correction time and export convenience.

A tool that does not meet a privacy or permission requirement should not proceed to a trial with sensitive material. Start with synthetic order notes. Check current vendor terms for the specific plan and feature; consumer chat, API and business plans can differ. Recheck prices when making the decision; they change.

Use hard failures for material errors; preferences for convenience.

Fictional workshop. Documented data and source scope are identified in the text.

03 / PRACTICAL FIELDNOTE

Build a small test pack

Include the awkward cases.

NORMAL
Complete and consistent

MISSING
Ask for the absent fact

CONFLICT
Flag the contradiction

EMPTY
Invent nothing

Prepare a normal order, a missing measurement, a contradictory delivery note and an empty input. Keep one correct reference brief for each case. The empty case should produce a request for information, not an invented order.

Freeze the input text and instructions so competing tools receive the same task. Record the product, version where visible, settings, date and whether web access or extra tools were enabled. If the interface does not expose a model version, record that limitation.

Keep a few cases aside until after you adjust the instructions. These held-out cases help reveal whether the prompt has become a clever fit to the examples already seen. A handful of cases screens out obvious mismatches. Establishing broader reliability takes substantially more evidence.

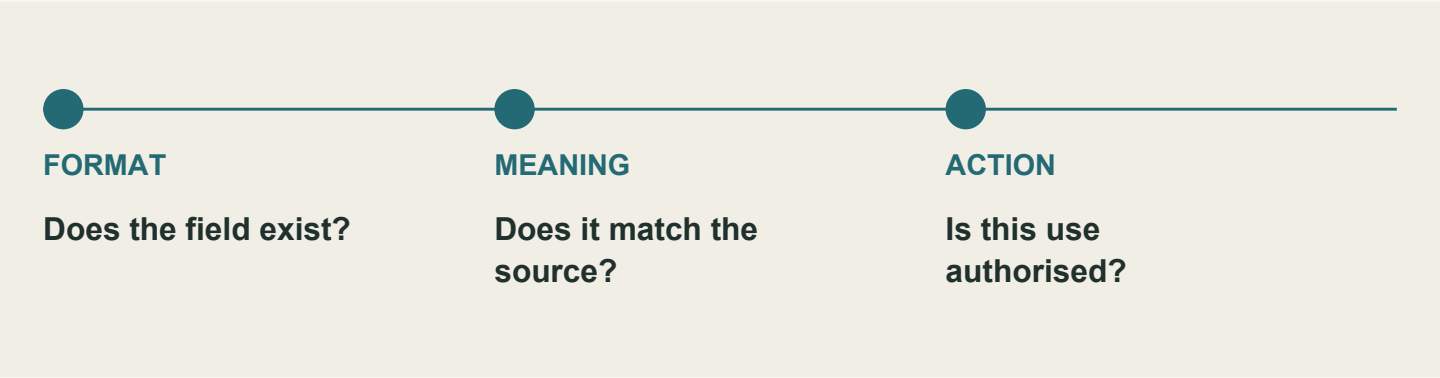
Normal, missing, contradictory and empty inputs all belong in the trial.

Fictional workshop. Documented data and source scope are identified in the text.

04 / PRACTICAL FIELDNOTE

A tidy structure can still be wrong

Schema checks and fact checks answer different questions.



A production brief can contain every required field and still change 120 cm to 120 mm. A format checker can establish that a field exists or contains a number. It cannot, by itself, establish that the number came from the order.

Some tools can constrain output to a fixed set of fields, often called a schema. Ollama's documentation is one technical example, cited for the mechanism rather than as a product recommendation. A completed brief still needs checking against its input.

Check meaning separately: every source measurement preserved, no invented deadline, unresolved details labelled, no action taken without authority. Treat a well-formed file as one stage of acceptance. The colleague using it should not have to discover the difference with a saw.

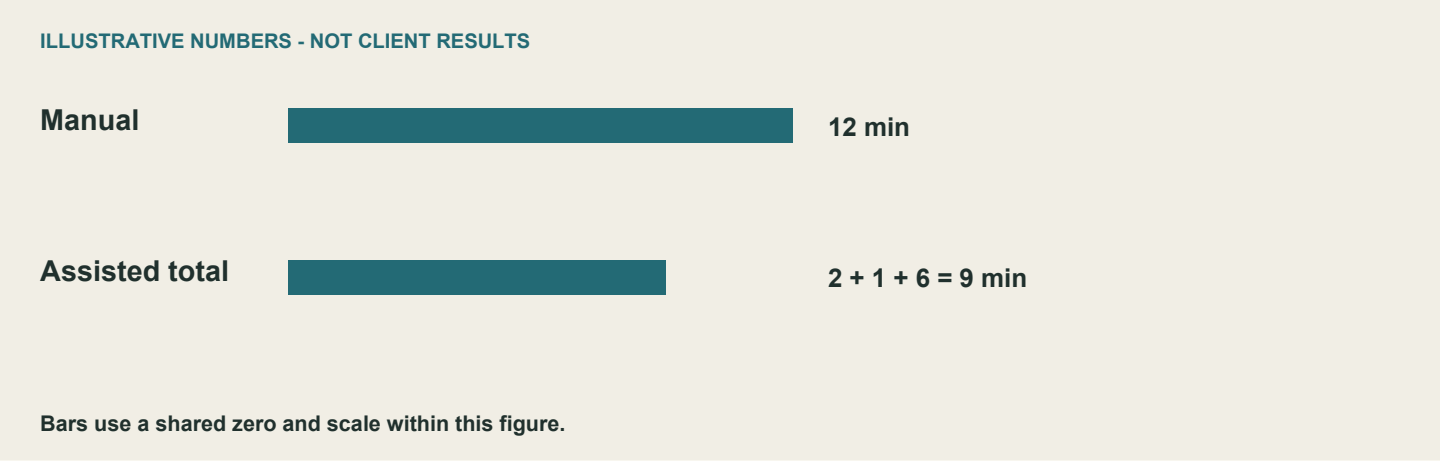
Valid structure is necessary for some workflows; it is not factual accuracy.

Fictional workshop. Documented data and source scope are identified in the text.

05 / PRACTICAL FIELDNOTE

Count the repair work

Fast generation can produce slow work.



Illustrative timing: a manual brief takes 12 minutes. An assisted draft takes 2 minutes to prepare, 1 minute to generate and 6 minutes to inspect and correct. Total: 9 minutes. The useful recurring saving is 3 minutes. Record one-off setup and test-pack preparation separately before estimating payback.

Repeat the measurement across ordinary and awkward cases. Include retries, transfers between tools and the time spent investigating a plausible error. Report failures rather than dropping them from the average.

Keep quality alongside time. A faster process that silently changes a dimension is unacceptable, however attractive its average. Do not multiply a tiny pilot’s time saving across a year as though every task, colleague and deadline behaves the same way. Use the trial to decide what deserves a longer test.

Total handling time = preparation + generation + checking + correction.

Fictional workshop. Documented data and source scope are identified in the text.

06 / PRACTICAL FIELDNOTE

Compare outputs without the badge

Preference can arrive before evidence.

HIDE
Remove tool names

SCORE
Apply the same rules

REVEAL
Compare the source tools

RESOLVE
Record disagreements

Where practical, remove tool names from the outputs and ask a second person to assess them against the acceptance rules. Keep the original inputs beside each output. The evaluator must be able to verify the content, not merely choose the nicest prose.

Record disagreements by case. One reviewer may prefer a shorter brief while another spots an omitted instruction. Resolve hard failures first; discuss style after the facts are secure.

If only one person is available, score the outputs before revealing their source where possible and revisit borderline cases after a break. This is not full blinding or a controlled scientific study. It is a modest way to make personal enthusiasm less influential. Keep a short explanation for each score so another person can understand it later.

Score against the task, and retain the reasons for disagreement.

Fictional workshop. Documented data and source scope are identified in the text.

07 / PRACTICAL FIELDNOTE

Build a shortlist you can finish

Start with tools you already have.

START

Tools already available

FILTER

Requirements and data rules

CAP

Two or three candidates

Check the tools you already pay for before looking for another subscription. Can one produce the required output, export it into your existing process and work within your data rules? Remove candidates that fail a mandatory requirement before uploading real material.

Keep the first shortlist to two or three candidates. Add a new tool only when it fills a specific gap, such as an essential file format or an approved offline workflow. A colleague's recommendation can suggest a candidate; your own test decides whether it fits.

For local versus hosted options, inspect the full data path and operating effort. Local work still uses hardware, electricity and maintenance; plugins can send data elsewhere. Hosted work brings service limits and external terms. Write down the constraint you need to test, then run the same task under that constraint.

Shortlist by requirements, then test the actual task.

Fictional workshop. Documented data and source scope are identified in the text.

08 / PRACTICAL FIELDNOTE

A two-week trial plan

Keep the experiment small enough to finish.

DAYS 1–2
Rules and test pack

DAYS 3–6
Runs and full output review

DAYS 7–10
Failures and held-out repeats

DAYS 11–14
Supervised work and decision

Suggested calendar: days 1–2, define the rules and test pack; days 3–6, run the same pack and review every output; days 7–10, investigate failures and repeat with held-out cases. Use days 11–14 for supervised suitable work and the decision.

Use the remaining time for a supervised trial on suitable real work, only after privacy and permissions are checked. Keep the existing method available. Record what happened when the tool failed, not just what happened when it worked.

At the end, choose adopt narrowly, extend the test, or reject. Do not let a trial become a subscription by inertia. Check renewal and cancellation terms before starting any paid trial. The calendar is a suggested organising aid, not a claim that two weeks is enough for every business or risk level.

Make the final decision date and the fallback method explicit.

Fictional workshop. Documented data and source scope are identified in the text.

09 / PRACTICAL FIELDNOTE

The decision sheet

Keep evidence beside the recommendation.

Test case	Hard failure?	Handling time	Decision / limit
Missing finish	Invented finish	Record minutes	Hold / clarify

First row is illustrative. Print and complete the blank rows; protect customer information.

For each tool, record the exact task, configuration, date, tested cases, hard failures, median handling time if useful, observed cost and unresolved risks. Separate measured amounts from estimates and include the basis for any conversion to monthly cost.

If the process handles customer information, add the approved data scope and retention requirements. If it can send messages or change records, state who authorises those actions and how duplicates are prevented. A good text generator is not automatically a safe publisher.

Conclude with one bounded recommendation: for example, “Use for draft production briefs with human measurement checks; do not send directly to production.” That tells the next operator more than a five-star rating. Repeat the relevant checks when the task, model or integration changes materially.

Approve a specific use, with limits, rather than declaring a universal winner.

Fictional workshop. Documented data and source scope are identified in the text.

10 / PRACTICAL FIELDNOTE

A useful failure is still useful

Reject a tool when the evidence calls for it.



Suppose a tool handles normal notes well but invents a finish whenever the order is incomplete. The answer is not to hide the awkward cases. Either improve the workflow and retest on held-out inputs, or keep that tool away from this job.

You may discover that a form with required fields solves the problem better than an AI assistant. That is a good outcome. The aim was a reliable production brief, not a more impressive software collection.

This issue draws on task-based evaluation principles used in our internal tool-selection work. It does not publish private benchmark results or claim that any named product won a fresh comparison. Your decision needs your own task evidence, including the inconvenient failures.

The cheapest useful result may be deciding not to automate this task.

Fictional workshop. Documented data and source scope are identified in the text.

The evidence behind this edition.

Ollama: structured outputs

https://docs.ollama.com/capabilities/structured-outputs

Source check: 19 November 2026. References support the specific factual passages marked in the online edition. They do not endorse Recallant or establish customer outcomes.

How this edition was made

Original editorial writing and diagrams, with AI-assisted drafting and independent critique. Arithmetic, claims and final layouts were checked before release. This is a practical guide, not a controlled evaluation, professional advice or a promise of commercial results. The accompanying HTML provides structured headings, readable text and source links; this visual PDF is not claimed to be PDF/UA certified.

Read the series / report a correction

recallant.com/fieldnotes.html
hello@recallant.com