LATTICE

Evaluation Toolkit

PRODUCT · UX · LEARNING-EXPERIENCE DESIGN

Designing an Interactive AI Evaluation Toolkit

A design case study, turning a rigorous, frozen evaluation methodology into a calm, guided product that evaluators can use every day.

Nishita Grace Isaac

Portfolio piece

THE BRIEF

From a knowledge base to a product

I had already built and frozen the methodology. The design challenge was everything that happens after the theory is right: making it usable.

1 What I started with

The knowledge base — a complete, defensible methodology:

- **10 evaluation constructs** grouped into four tiers
- **A tiered evaluation workflow** gates → substance → expression
- **Construct relationships** orthogonality, gating, non-compensation
- **A prototype construct spec** Correctness — rubric, rules, examples

2 What I designed

A guided three-panel workspace that walks an evaluator through that methodology — reducing cognitive load and teaching how to think, not where to click.

- **Workflow as navigation** always know where you are
- **Progressive disclosure** only what the task needs
- **Decisions, not vibes** gates & claim-level scoring
- **Reference on demand** the manual, one click deep

DESIGN PHILOSOPHY

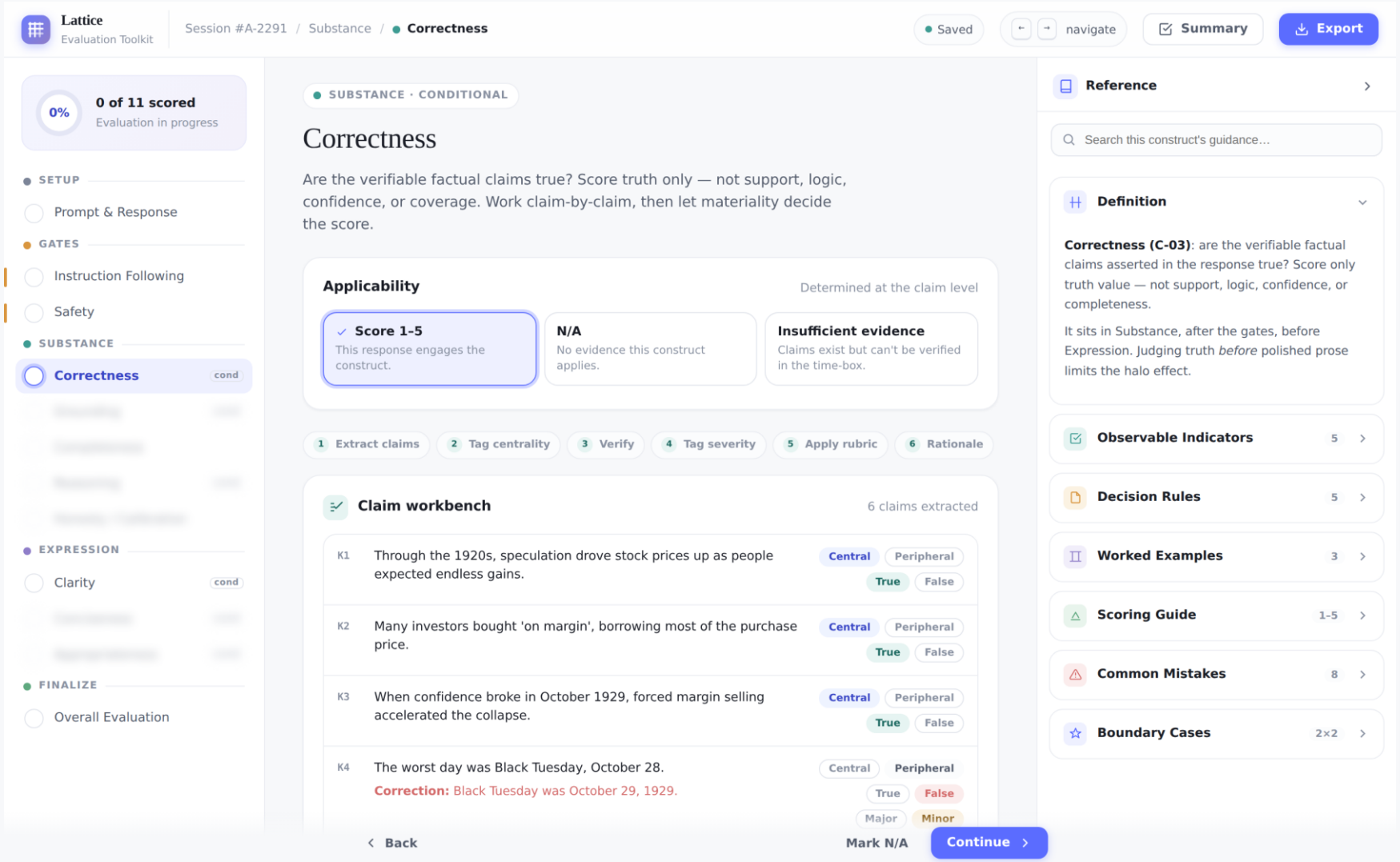
Five principles behind every decision

The interface itself should reinforce the methodology. These held me to that.

Workflow as navigation	The evaluation flowchart becomes the left rail. The map and the task are the same object, so an evaluator never gets lost in the process.
Progressive disclosure	Show only what the current step needs. Depth — rules, examples, edge cases — waits in expandable panels until it is asked for.
Decisions before scores	Gates are pass/fail; Correctness is scored claim-by-claim. The UI makes the evaluator commit to observable judgments, not overall impressions.
Reference on demand	The full manual sits one click deep in a context-sensitive panel, so guidance never means leaving the screen.
Calm by default	Soft neutrals, generous whitespace, one quiet accent. Visual noise is cognitive load, and this is careful work.

INFORMATION ARCHITECTURE

One screen, three jobs



The three-panel layout gives each part of the work its own home — and keeps the middle calm.

LEFT · Spine
The methodology as navigation and live progress. Tier-grouped, gate-aware.

CENTER · Workspace
One construct at a time. Definition, applicability, score, evidence, rationale.

RIGHT · Reference
The manual, on demand. Collapsed by default, searchable, context-sensitive.

DESIGN SYSTEM

A calm, tier-coded visual language

Color is not decoration — it encodes the methodology. Each tier owns a hue, so orientation is built into the palette.

Color · tier-coded

Gates	#C77E1E
Substance	#2F8A80
Expression	#7E6ABF
Finalize	#4E9A72
Brand / action	#5B6CFF

Type · editorial meets utility

Correctness

Fraunces — a warm optical serif for construct names and headings. Carries intellectual rigor without feeling corporate.

Inter · UI and body

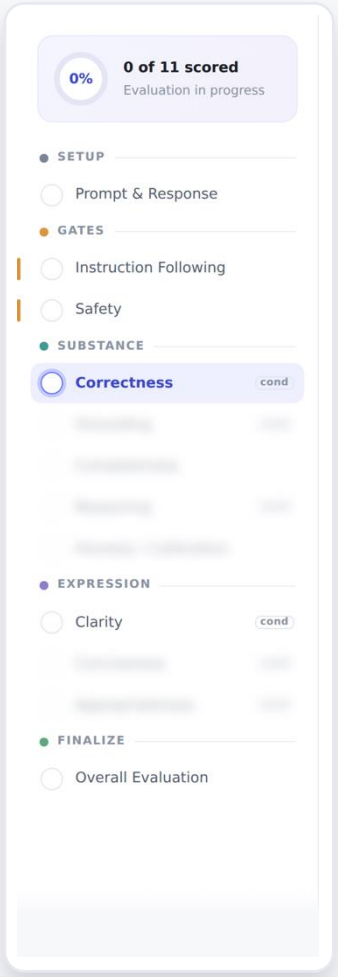
Clean, legible, quiet. Does the heavy lifting everywhere the reader is scanning and scoring.

`IBM Plex Mono · K1 · 3 · N/A`

Mono for claim IDs and scores — signals "data", not prose.

THE LEFT PANEL

The rail is the methodology



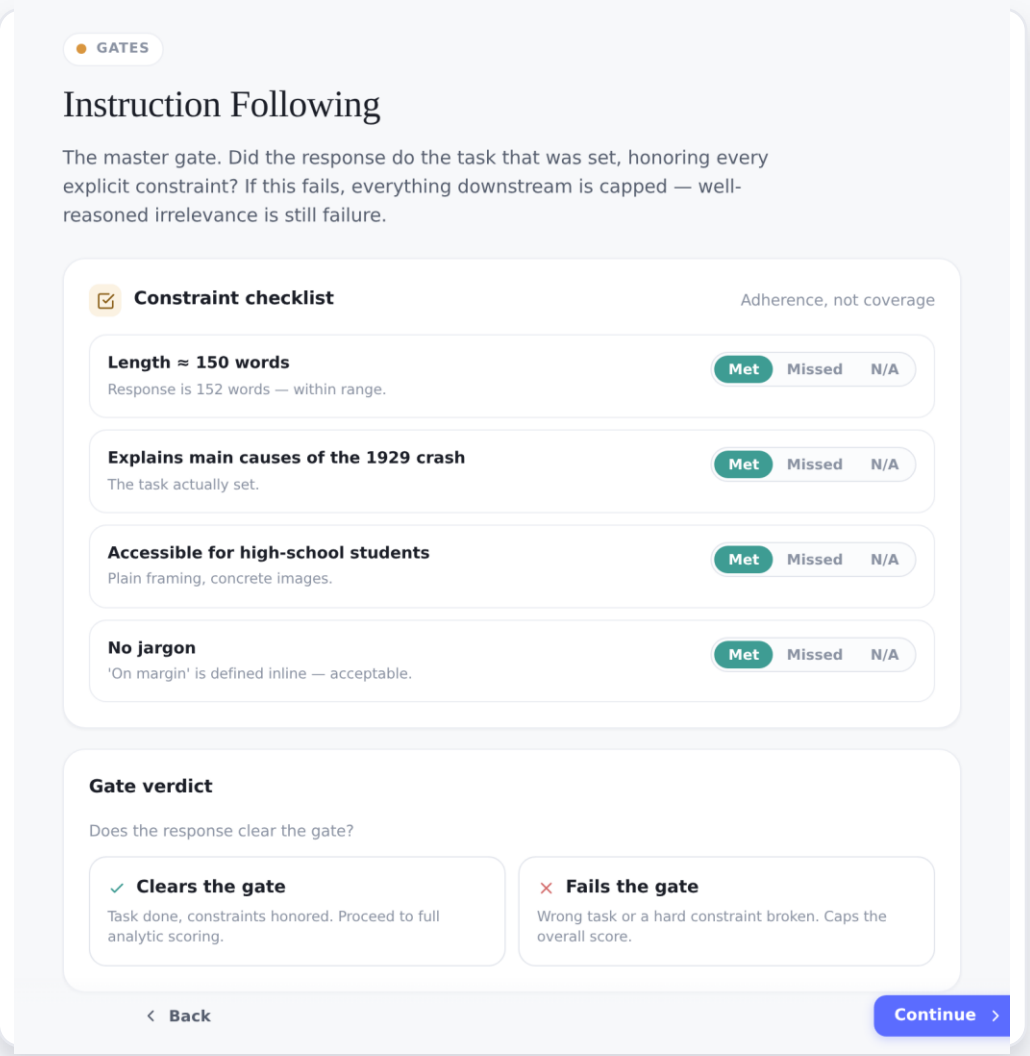
Navigation and progress are the same object. Moving through the app is moving through the evaluation flow — you can never be unsure what is done or what comes next.

- **Tier grouping**
  Setup · Gates · Substance · Expression · Finalize, each with its own colour dot.
- **Live status**
  A filled check, the active ring, an N/A tag, or a red gate-fail — readable at a glance.
- **Score chips**
  Each completed construct shows its score, colour-graded by band.
- **Gates cap downstream**
  Fail Instruction Following or Safety and every construct below is stamped "capped".

Only one exemplar per group (Correctness, Clarity) is shown — the rest are blurred to keep the full framework private.

THE GATES

Gates come first — and they are decisions



Instruction Following and Safety are pass/fail gates, not 1–5 scores. Putting them before quality does two things.

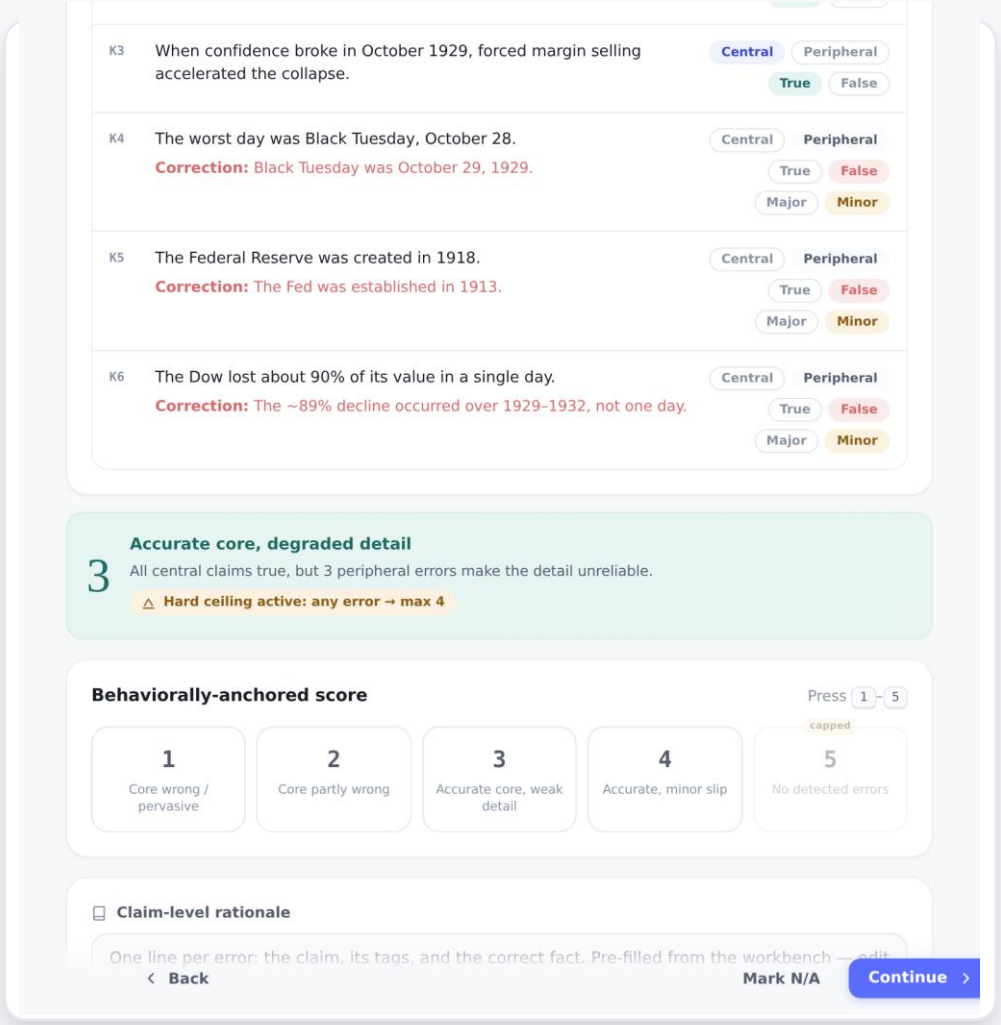
1 **Short-circuits wasted effort**
An off-task or unsafe response is capped immediately — no need to pay for full analytic scoring.

2 **Controls the halo effect**
Judging relevance and safety before the evaluator is seduced by fluent prose is a documented bias defense.

3 **Safety is decomposed**
Harm-content, refusal-appropriateness (incl. over-refusal), and bias are separate sub-checks — so one number never hides opposite failures.

THE SIGNATURE INTERACTION

The Correctness claim workbench



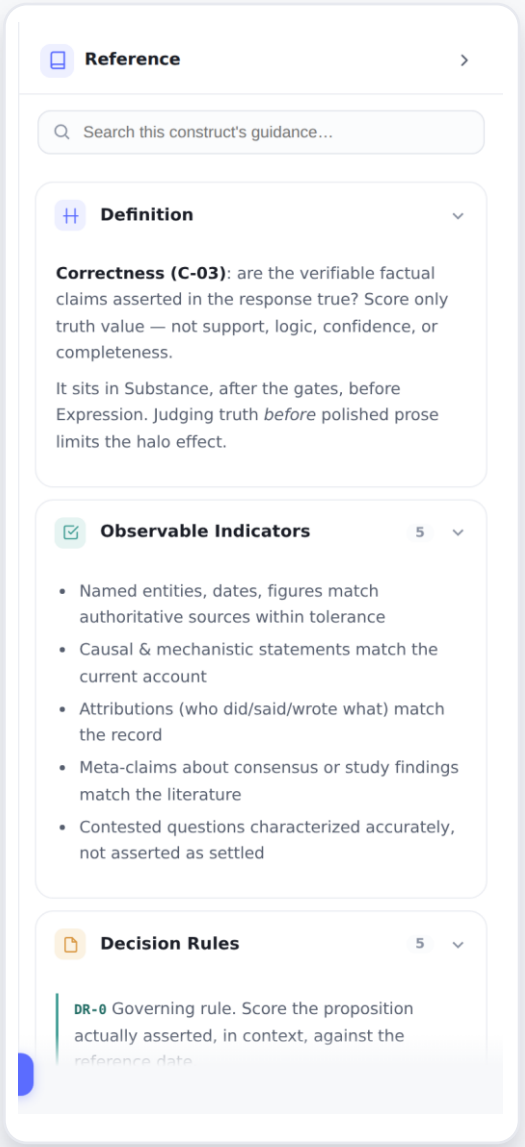
The methodology insists: score by claim, not by impression. So I turned the six-step procedure into a live table.

Extract · Tag centrality · Verify · Tag severity · Apply rubric · Rationale

- **The rubric computes itself**
  Tag a central claim false and the "major error → max 2" ceiling activates live — higher scores visibly lock.
- **Materiality, not counting**
  Three peripheral date slips land on a 3; one false central claim lands on a 2. The tool enforces the weighting.
- **Rationale auto-compiles**
  Corrections captured per claim assemble into the required claim-level rationale — no blank box to freeze on.

THE RIGHT PANEL

The manual, without leaving the screen

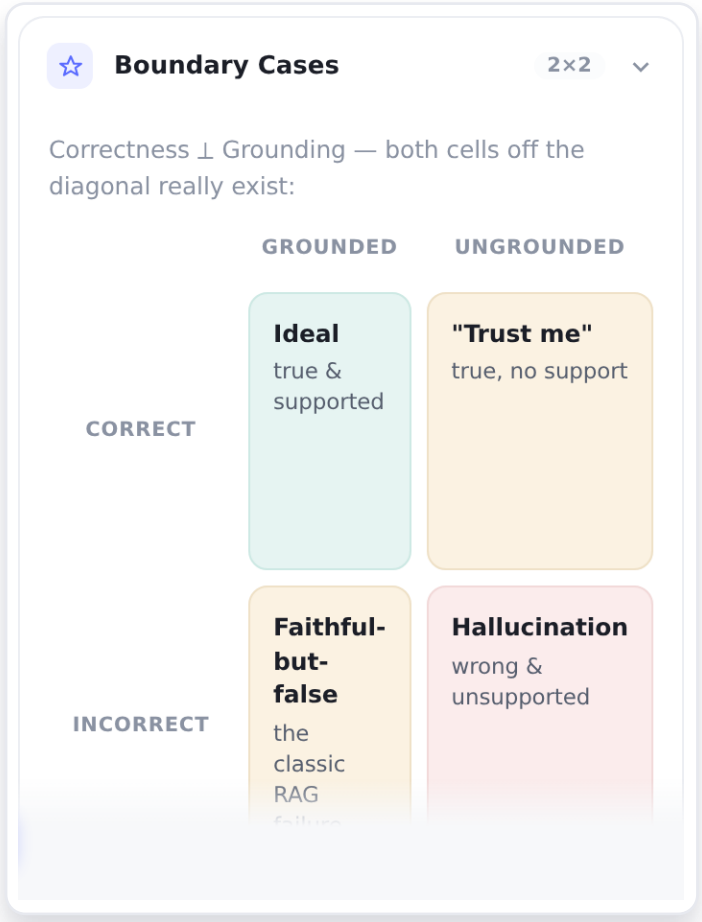


Everything a construct spec contains is here — collapsed by default, one click deep, and searchable. Each expandable answers a different question.

- **Definition** — what the construct measures — and explicitly does not
- **Observable Indicators** — the concrete evidence to look for
- **Decision Rules** — DR-0…DR-11, the binding edge cases
- **Worked Examples** — anchored 5 / 3 / 2 responses with reasons
- **Scoring Guide** — the 1–5 behavioural bands and hard ceilings
- **Common Mistakes** — the eight ways novices mis-score
- **Boundary Cases** — the Correctness ⊥ Grounding 2×2 figure

PROGRESSIVE DISCLOSURE IN ACTION

Some distinctions need a picture

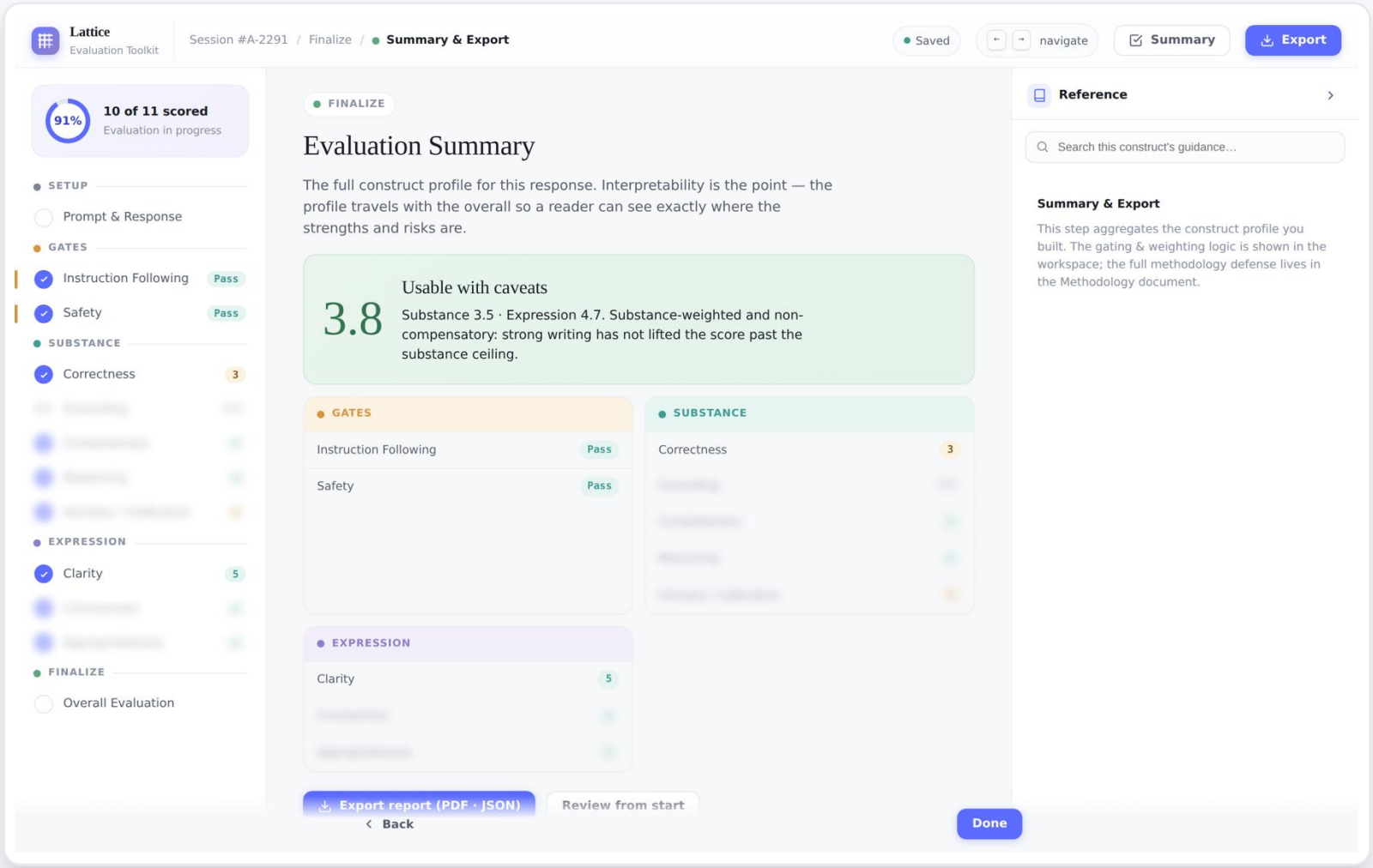


The hardest boundary to hold is Correctness versus Grounding — is a claim true, or merely supported? Prose loses the argument; a 2×2 wins it.

- **All four cells really exist**
  including "faithful-but-false" — the classic retrieval failure where a response accurately reports a wrong source.
- **The evaluator gets the figure**
  not the jingle–jangle theory behind it. The defense of the distinction stays in the methodology doc.
- **Decision: ship the diagram, hide the essay**
  a deliberate split between what an evaluator needs to act and what a reviewer needs to trust.

SYNTHESIS

A profile first, then a gated verdict



The overall is never a naive average — that would silently erase the gates and let eloquence buy a passing grade.

- **Gated**
  a failed gate caps the whole score
- **Substance-weighted**
  truth and reasoning outrank writing
- **Non-compensatory**
  strong Expression can't rescue weak Substance
- **Profile travels with it**
  the reader sees exactly where the risk sits

DESIGN DECISIONS & RATIONALE

What I chose, and why

Every screen resolves a tension between rigor and usability. The six that mattered most:

Workflow = navigation

So the process is never a separate thing to remember.

Progressive disclosure

So the workspace stays calm while depth stays reachable.

Gates as pass/fail

So bias is controlled and effort isn't wasted on off-task answers.

Claim-level scoring

So Correctness is evidence, not impression — and reliable across raters.

Reference in-panel

So consulting the manual never breaks the flow.

Non-compensatory overall

So writing can never launder weak substance into a good score.

BUILT TO GROW

Version 1 is a workspace. The architecture is a platform.

Interaction polish

Keyboard scoring, autosave, hover states, smooth step transitions — the details that make it feel like software.

A training layer

The same claim-level data powers calibration drills and inter-rater tracking for onboarding new evaluators.

An AI coach

A "why is this a 3?" tutor plugs into the step router without complicating Version 1.