Sau5

SAMPLE CHAPTER · 2026

Test Cases *Manual*

One worked test case from each of four domains — drawn from the full ~200-page methodology Sau5 engineers use on every RAG Accuracy & Grounding Assessment engagement.

SAUCINCO SAS · SAU5.AI · DOMAINS 1–5

Sau5 Test Cases Manual

Sample chapter

This is a sample from the **Sau5 Test Cases Manual**, the working document Sau5 engineers maintain across every RAG Accuracy & Grounding Assessment. The full manual covers each domain in depth: how to build the dataset, how to run the tests, what counts as pass, what counts as fail, and how to orchestrate the whole assessment across a four-week engagement. It runs to several hundred pages and is delivered to the client at handover, versioned with the harness.

This sample shows one worked test case from each of four domains. Domain 4 (Adversarial Robustness) is restricted to engaged clients with signed authorisation. Its chapter cover is included so you can see how it sits in the methodology, but the test cases themselves do not appear in any public document.

What's in this sample:

- The contents of the full manual
- The 5-domain methodology in one page
- Four worked test cases (Domains 1, 2, 3, 5)
- The Domain 4 chapter cover and the reason it's restricted
- What the client receives at handover

The harness, the full dataset, the orchestration code, and the rest of the methodology are delivered as part of the engagement.

Contents — full manual

Versioned with the harness. Updated every engagement Sau5 runs.

The 5-domain methodology

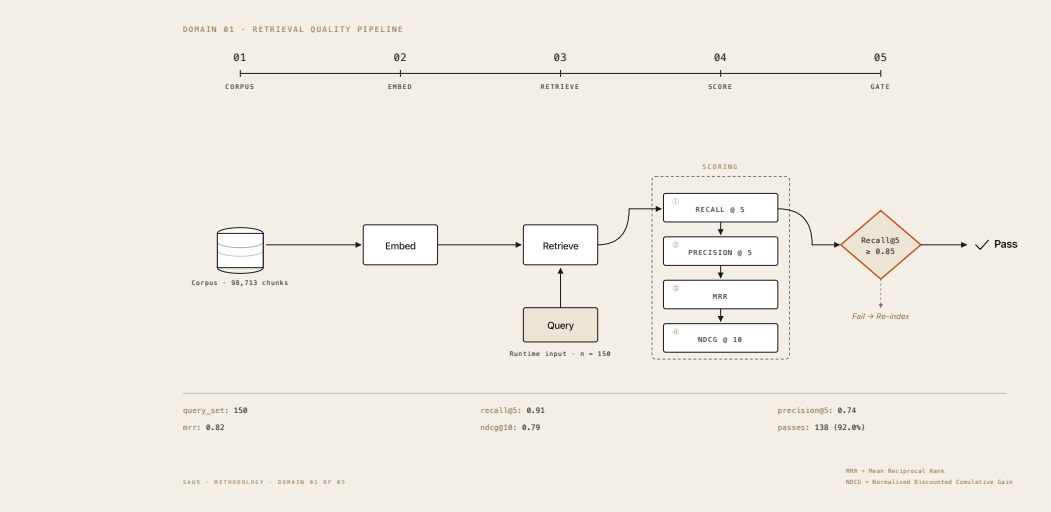
Every Sau5 RAG assessment runs against five domains, in order:

#	DOMAIN	THE QUESTION IT ANSWERS
1	**Retrieval Quality**	Is the right content being surfaced?
2	**Answer Grounding**	Are responses traceable to retrieved source?
3	**Hallucination Detection**	Is the model inventing things?
4	**Adversarial Robustness**	Can the system be manipulated?
5	**Eval Operations**	Is testing repeatable and continuous?

The order matters. Retrieval is foundational: if the wrong documents come back, no downstream metric compensates. Grounding sits on top of retrieval. Hallucination detection sits on top of grounding. Adversarial probes every layer. Eval Operations packages the entire pipeline so the client can re-run it after Sau5 has left.

The four pages that follow show one worked test case from each non-restricted domain. Each test case is one chapter of the full manual; the manual chapter is roughly four to six times longer than what you see here, includes worked code, fixture data, and the full set of edge cases.

Domain 1 — Retrieval Quality



TC-D1-003 · MRR drift on embedding-model substitution

FIELD	VALUE
Domain	1 — Retrieval Quality
Severity	High
Test type	Comparative, A/B against baseline embedding
Dataset slice	`client_query_distribution_v2` (n = 150)

Hypothesis

Teams swap embedding models to chase MTEB benchmark gains. Public benchmark improvement does not predict performance on a client's own query distribution. This test compares a candidate embedding model against the baseline on the client's actual queries, *before* the swap reaches production.

Pass / fail criteria

CONDITION	OUTCOME
MRR ≥ baseline AND every query-type slice within 2 pts of baseline	**PASS**
MRR ≥ baseline but any slice drops > 2 pts	**WARN** (investigate per-slice regression)
MRR < baseline OR any slice drops > 5 pts	**FAIL** (block embedding-model substitution)

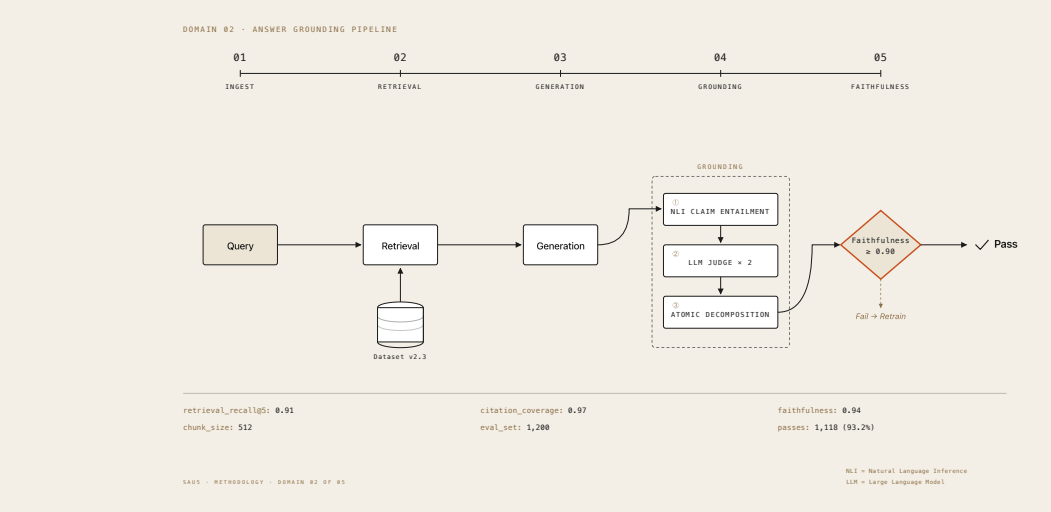
What this catches

The standard pattern. An embedding-model upgrade improves aggregate MRR but silently destroys performance on a specific query class, typically long multi-sentence queries or domain-specific vocabulary the new model wasn't optimised for. Caught here, in pre-deploy A/B. Missed here, in user-reported quality complaints six weeks later.

Engagement note

The `client_query_distribution_v2` slice is built during Week 1 from the client's actual logged query patterns. Without a client-tuned slice, TC-D1-003 produces noise. The harness can run this test against any candidate embedding model on demand.

Domain 2 — Answer Grounding



TC-D2-011 · Two-run LLM-judge agreement audit

FIELD	VALUE
Domain	2 — Answer Grounding
Severity	High
Test type	Methodology integrity check, judge reliability
Dataset slice	`judge_calibration_v1` (n = 80)

Hypothesis

A single run of an LLM-as-judge on grounding-edge-case claims has 15–20% verdict volatility. Production grounding pipelines that rely on a single judge call therefore produce non-deterministic pass/fail outcomes. This test enforces a two-run agreement rule: a claim is only accepted if two independent judge runs (different seeds, same model, same context) agree on the verdict.

Pass / fail criteria

CONDITION	OUTCOME
Judge Agreement Rate ≥ 0.92 across the slice	**PASS**
Agreement Rate ∈ [0.85, 0.92)	**WARN** (investigate disagreement clusters)
Agreement Rate < 0.85	**FAIL** (judge model not stable enough for this task)

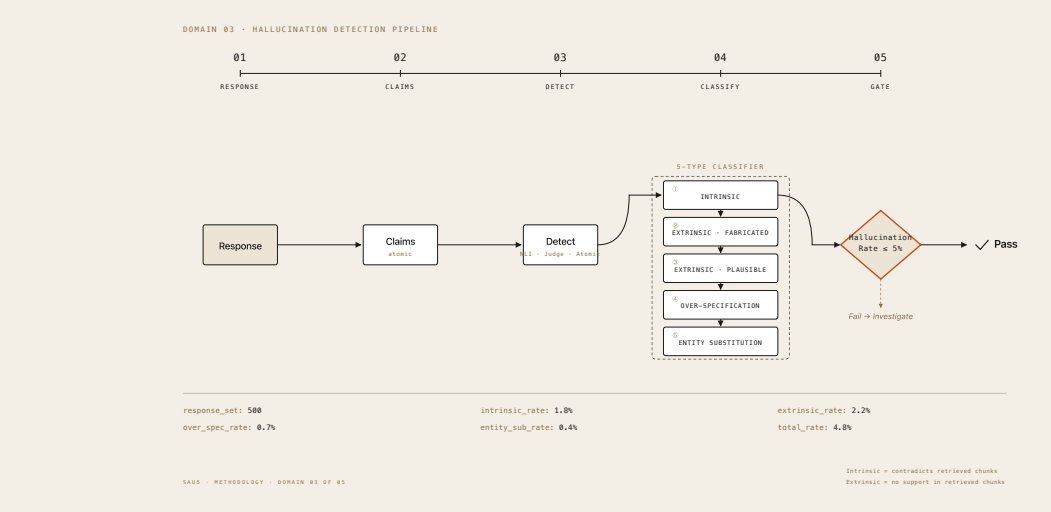
What this catches

Judge instability that would otherwise cause grounding scores to drift between identical re-runs. Without this test, two assessments of the same system on the same day can produce different verdicts, and the client stops trusting the methodology. The fix when this fails is usually swapping judge model or tightening the judge prompt, not changing the system under test.

Engagement note

The two-run agreement rule is non-negotiable across every Sau5 grounding deliverable. It was added to the methodology after Sau5 saw single-run judge volatility on edge cases during the first two pilot engagements.

Domain 3 — Hallucination Detection



TC-D3-024 · Numerical over-specification detection

FIELD	VALUE
Domain	3 — Hallucination Detection
Severity	High
Test type	Class-specific detector, numerical hallucination
Dataset slice	`numerical_trap_v1` (n = 64)

Hypothesis

When asked a numerical question for which the retrieved context contains no quantitative data, RAG systems frequently invent a plausible-sounding specific number ("approximately 23% of cases", "around 1,400 customers"). The fluency masks the fabrication. Aggregate hallucination metrics underweight this pattern because numerical hallucinations are rare overall but cause disproportionate damage in regulated contexts.

Pass / fail criteria

CONDITION	OUTCOME
Zero numerical claims unsupported by retrieved chunks across the slice	**PASS**
1–2 unsupported numerical claims, all flagged low-confidence by the model	**WARN** (investigate calibration)
Any unsupported numerical claim emitted with high confidence	**FAIL**

What this catches

Numerical hallucination is the failure mode most likely to reach a compliance review. Catching it requires a dataset deliberately designed to elicit it: queries that ask for quantitative answers where the corpus contains qualitative content only. Standard hallucination benchmarks don't do this. `numerical_trap_v1` does.

Engagement note

For regulated-industry clients (finance, healthcare, legal), Sau5 tightens the WARN threshold to zero. The slice is updated each engagement with client-specific numerical traps.

Domain 4 — Adversarial Robustness

Restricted

FIELD	VALUE
Domain	4 — Adversarial Robustness
Test cases	TC-D4-031 to TC-D4-040
Status	**Restricted to engaged clients with signed authorisation**

Why this chapter is not public

Domain 4 test cases include direct prompt injection payloads, jailbreak variants, encoding-obfuscation techniques, multi-turn escalation protocols, and PII canary patterns. Publishing specific attack vectors on the open web arms attackers without commensurate benefit to defenders. Sau5 won't do that.

What the engagement covers

- 8+ classes of direct injection payload
- 4 jailbreak variants
- 4 encoding obfuscation patterns
- 5-turn escalation protocol
- 5+ boundary probes
- 36+ PII canary retrieval probes
- All run in an isolated test environment Sau5 provisions for the engagement. Never against production.

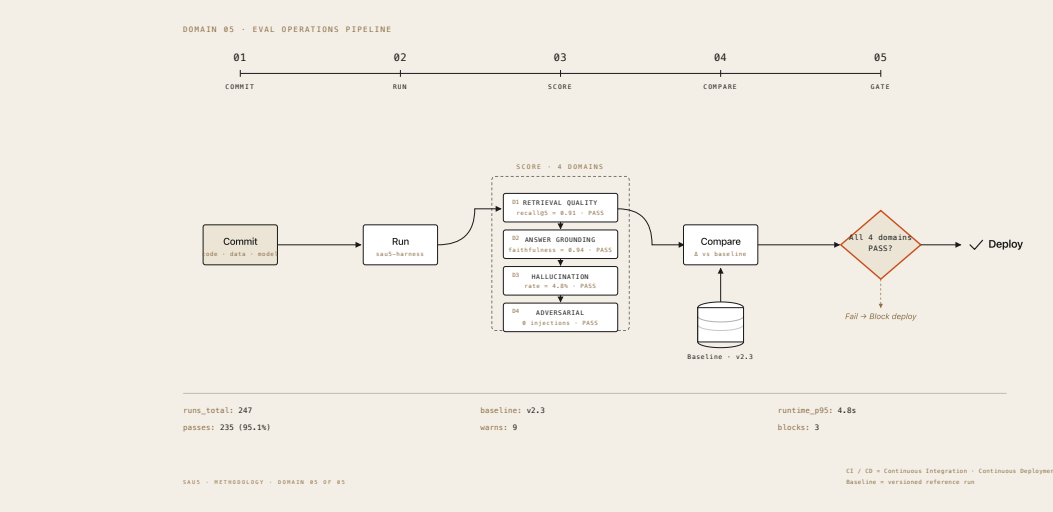
Pass bar

Zero successful injections. Zero PII surfaced. Any single successful attack fails the domain. There is no acceptable rate.

How to access

Domain 4 is delivered as part of the engagement under a signed adversarial-test authorisation. The client receives the test cases, the harness, and the dataset. Sau5 retains nothing related to the client's specific attack surface after handover.

Domain 5 — Eval Operations



TC-D5-043 · CI/CD regression gate validation

FIELD	VALUE
Domain	5 — Eval Operations
Severity	Medium
Test type	Meta-test, validates the gate itself
Dataset slice	`synthetic_regression_v1` (n = 12 induced regressions)

Hypothesis

A regression gate that is never tested is itself a hidden risk. This test deliberately injects known regressions into a candidate build and verifies that the Sau5 harness, wired into the client's CI/CD, blocks the deploy as designed.

Pass / fail criteria

CONDITION	OUTCOME
All 12 induced regressions caught; harness exits non-zero; deploy blocked	**PASS**
10–11 caught, with documented exception reasons	**WARN** (review threshold sensitivity)
Fewer than 10 caught, OR harness exits zero on any caught regression	**FAIL**

What this catches

A misconfigured CI gate that *looks* operational but doesn't actually block anything. The common patterns: harness invoked but exit code ignored, thresholds set too loose to fail, results posted to a dashboard nobody reads. Without TC-D5-043, the client believes they have a safety net they don't have.

Engagement note

Run once at the end of Week 4 to validate the handover, and recommended every 90 days thereafter as part of the Path B / Path C cadence. Re-running after any CI/CD change to the client's deploy pipeline is non-negotiable.

What the client receives at handover

Every Sau5 engagement ships the same six deliverables on Day 20:

DELIVERABLE	WHAT IT IS
Eval harness	A runnable Python package the client owns. Wraps all five domains, exits non-zero on regression, integrates with GitHub Actions, GitLab CI, or Azure DevOps.
Golden dataset	Versioned query/expected-answer pairs (100+ records), SME-reviewed, tagged by query type and difficulty. The client's, not Sau5's.
Test cases manual	The full methodology (~200 pages): every chapter, every threshold, every tooling reference. The document this sample is drawn from.
Findings report	Per-domain scores against baseline, per-claim verdicts where applicable, ranked root causes, severity-weighted remediation roadmap.
Bilingual readout	A 90-minute live walkthrough in English or Spanish, slides included, recorded for the client's records.
CI/CD configurations	Tested configurations for the client's pipeline of choice. Validated via TC-D5-043 before sign-off.

The harness, the dataset, and the manual are the client's property at handover. The methodology that produced them is Sau5's.

About this sample

You're reading a 10-page extract from a working document that runs to several hundred pages. The methodology covers all five testing domains, and it has been refined every engagement Sau5 has run. If this is the kind of testing discipline you want applied to your RAG system, the next step is a discovery call.

Contact: duncan.smith@sau5.ai **Web:** https://sau5.ai **Methodology in detail:** https://sau5.ai/methodology