

Audit Package Templates

Six fillable scaffolds. Each is usable as-is — copy, fill, attach. The audit package is the bundle a builder, an evaluator, or an external auditor will be asked for when the question is: "Prove this system works, and prove you knew it worked."

1. Evaluation Framework Doc

```
# Evaluation Framework – <system name>

**System under evaluation:** <name@version>
**Owner:** <named human>
**Quality tier (per s2-evaluation-charter):** demo | operational | high-stakes
**High-stakes triggers present:** [regulator-visible | financial | irreversible | PII |
statutory-substitute]

## 1. Scope
What this system does, in one paragraph. What it does NOT do.

## 2. Risks Retired
- User risk: <list>
- Regulatory risk: <list with statute/grant IDs>
- Operational risk: <list>

## 3. Test Families (link to s2-test-architecture)
- Functional: <count> tests; champion baseline <verdict summary>
- Safety: <count> tests; seeded from failure_ids: <list>
- Quality: <count> tests; rubric ID <ref>
- Edge-case: <count> tests

## 4. Evaluator Roles Invoked (per s2-evaluator-roster)
<table of role → actor → scope → cadence>

## 5. Scoring Approach (per s2-scoring-system)
- Pass threshold: <weighted avg>
- Critical floors honored: yes/no
- Confidence caps applied: yes/no

## 6. Provenance Coverage (per s2-audit-trail-schema)
- Schema version: <semver>
- Required fields present on 100% of runs: yes/no
- Explainable bucket reproducible into 1-page summary: yes/no

## 7. Regression Posture (per s2-regression-discipline)
- Comparison harness in place: yes/no
- Cadence: <weekly | per-change | both>
- Open intentional regressions: <count, with expiries>

## 8. Sign-Offs
<evaluator signatures table>
```

2. Auditability Checklist

A binary checklist. Every item must be checked or have a written exception attached.

```
# Auditability Checklist – <system name@version>

## Charter
```

```
- [ ] Quality tier declared and recorded in repo
- [ ] High-stakes triggers explicitly evaluated
- [ ] Risk categories (user/regulatory/operational) mapped

## Failure-Mode Coverage
- [ ] Every `critical` failure_id from s1-failure-mode-register has ≥ 1 safety test
- [ ] Every `high` failure_id has ≥ 1 safety test
- [ ] Every `medium` failure_id has a test OR a documented monitoring signal

## Provenance
- [ ] All required fields present on a sampled 50-run audit
- [ ] `decision_trace` contains ≥ 1 step on every run
- [ ] `decision_origin` values are from the allowed enum only
- [ ] `source_confidence` populated on every retrieval_source
- [ ] Explainable-bucket fields reproducible into reviewer summary

## Evaluators
- [ ] Each test family has a named signatory role
- [ ] Two-of-two requirement honored for high-stakes
- [ ] No self-clearing on high-stakes systems
- [ ] Compliance veto path operative

## Scoring
- [ ] Rubric committed and versioned
- [ ] Severity weights match s1 enum
- [ ] Confidence caps applied on retrieval-grounded criteria
- [ ] Floor rule enforced (no critical criterion below floor)

## Regression
- [ ] Comparison harness runs on every triggering change
- [ ] Change log entries are append-only
- [ ] No `unintentional_regression > 0` entries with `decision = promoted`
- [ ] Every intentional regression has pre-declared intent + expiry

## Retention
- [ ] Evidence retention policy attached (see template 5)
- [ ] PII redaction verified on retained records

Exceptions (with rationale and expiry): <list or "none">
```

3. Test-Case Library Structure

```
test_library:
  library_id: <system>-tests@<semver>
  test_count_by_family:
    functional: <int>
    safety: <int>
    quality: <int>
    edge_case: <int>
  test:
    - test_id: <FAMILY-PREFIX>-<number>      # e.g., S-S1-04
      family: functional | safety | quality | edge_case
      failure_id_refs: [<failure_id>, ...] # required for family=safety, may be empty
  otherwise
    title: <short human label>
    inputs:
      raw: <string>
      channel: <enum>
      fixtures: [<path or hash>, ...]
    expected_behavior: <one paragraph>
    pass_criteria:
      - <rule>                                # e.g., "outputs.refusal == true"
    fail_handling:
      on_fail: <enum: block | rework | log-and-triage>
      escalation_role: <role from s2-evaluator-roster> | null
```

```

rubric_ref: <criteria IDs from sample rubric>
tags: [pii, sunshine, grant, tax, jurisdiction, tone, ...]
created_at: <iso8601>
last_modified_at: <iso8601>

```

4. Scoring Rubric Template

```

# Rubric – <name>

**Applies to:** <system + test family>
**Scale:** 0-5 integer (per s2-scoring-system)
**Tier required:** demo | operational | high-stakes
**Weighted-average pass:** <number>

## Criteria

| # | Criterion | Severity | 0 anchor | 3 anchor | 5 anchor | Floor (min score) |
Confidence-capped? |
|---|---|---|---|---|---|---|
| 1 | <name> | critical|high|medium|low|info | <description> | <description> |
<description> | <int> | yes|no |
| 2 | ... | | | | | |
| 3 | ... | | | | | |

## Confidence Cap Map (per s2-scoring-system, C4 contract)
- high → no cap (max 5)
- medium → capped at half-credit (max 3)
- low → capped at 0
- unknown → capped at 0 AND mark_for_review = true (routes to domain-expert-reviewer)

## Worked Example
<one filled-out sample with scores and weighted average computation>

```

5. Evidence Retention Policy

```

# Evidence Retention Policy – <system name>

## Records in Scope
- Provenance records (per s2-audit-trail-schema)
- Scorecard fragments (per s2-scoring-system)
- Regression reports and change-log entries (per s2-regression-discipline)
- Evaluator signatures and rationales

## Retention by Class
| Class | Default Retention | Trigger Extending Retention |
|---|---|---|
| High-stakes provenance records | 7 years | open litigation hold, regulator inquiry, grant audit window |
| Operational provenance records | 18 months | open incident, regression investigation |
| Demo provenance records | 90 days | none |
| Evaluator signatures | matches record class | matches record class |
| External-auditor findings | 7 years | always |

## PII Handling
- `inputs.normalized` is the redacted record. `inputs.raw` containing PII is retained only when required for incident reconstruction, encrypted at rest, access-logged.
- `pii_flags` array on every record drives redaction-on-export.
- Florida Sunshine: public-record obligations override deletion schedules; consult compliance-reviewer.

## Disposal

```

```
- Disposal log appended to the audit package on every batch deletion: `{record_class, count,
disposed_at, disposed_by, schedule_ref}`.

## Owner
Retention policy owner: <named human>. Reviewed annually.
```

6. Evaluation Report Template

The deliverable an evaluator hands a reviewer or auditor at the end of a cycle.

```
# Evaluation Report – <system name@version>

**Period covered:** <start> – <end>
**Quality tier:** <tier>
**Reporting evaluator:** <named human>

## Headline (Pyramid Principle)
**Recommendation:** Promote | Hold | Roll back | Investigate
**One-sentence rationale:** <answer first>

## Results by Test Family
| Family | Tests run | Pass | Fail | Partial | Weighted avg | Drift flags |
|---|---|---|---|---|---|---|
| Functional | | | | | | |
| Safety | | | | | | |
| Quality | | | | | | |
| Edge-case | | | | | | |

## Failure-Mode Coverage
<table mapping failure_id → test_id → verdict; gaps highlighted>

## Notable Provenance Findings
- `decision_origin` distribution: agent <%, human-override <%, fallback <%, escalation
<%>
- `source_confidence` distribution: high <%, medium <%, low <%, unknown <%>
- Caps applied: <count>; capped claims summary

## Regression Posture
- Comparison report ID: <ref>
- Delta summary: <improved | unchanged | intentional | unintentional | new-fail>
- Open intentional regressions: <count> (with expiries)

## Open Items
- Risks accepted: <list>
- Findings to remediate: <list with owners and deadlines>

## Sign-Offs
<evaluator signatures with roles, scopes, verdicts, and timestamps>

## Appendices
- Linked provenance records: <list of record_ids or query>
- Linked scorecards: <list>
- Linked change-log entries: <list>
```