GSIS Public Verification Worksheet Pack v0.1

An Overleaf-ready operational kit for public transition review, benchmark packets, and standards crosswalks.

Parent lens	GSIS — Universal Structural Integrity Standard
Operational controls	FMLIS — Memory, Learning, Insulation
Scoring layer	MVS — normalized metric and threshold worksheet
Purpose	Turn the public test surface into a real review packet instead of a rhetorical page.
Primary use	Packet intake, state-transition review, guard checking, contradiction logging, threshold review, and conformance declaration.

Version	0.1
Prepared for	Kapukai Governance Lab
Document type	Worksheet pack / benchmark kit / public verification companion
Compile target	Overleaf or standard pdflatex workflow
Packaging rule	Keep doctrine pages narrow. Put standards in the standards layer. Put public proof on a separate test surface.

Core GSIS framing: event packet → candidate next state → guard evaluation → transition classification → accepted, blocked, or unresolved state.

Contents

1 How to Use This Kit

Intent

This pack is designed to operationalize public review. It is not a general essay. It is a compact set of forms and registers that let a reviewer examine whether a claimed outcome is structurally supportable.

What this pack does

- Treats each packet as an attempted **state transition**.
- Forces the reviewer to write the **current state**, **candidate next state**, and **required guards**.
- Requires explicit recording of **missing provenance**, **contradictions**, and **unresolved uncertainty**.
- Crosswalks the packet against **FMLIS controls** and an **MVS-style threshold worksheet**.
- Produces a short **decision record** and a **conformance declaration**.

Result states

The public test surface and the worksheet pack should use the same four result classes:

PASS	Required proof objects are present; the transition is structurally supportable under the stated threshold.
FAIL	One or more guard conditions are not satisfied, or a required object is missing in a way that blocks supportability.
INSUFFICIENT	The packet does not meet the declared release floor; more evidence or stronger verification is required.
UNCERTAIN — MORE EVIDENCE REQUIRED	Contradictions, opacity, or unresolved guard indeterminacy prevent a final classification.

Packaging recommendation for the website

- `/observable/`: keep narrow.
- `/standard/`: list GSIS, FMLIS, MVS, TRA, then TRS if and only if the public artifacts are real.
- `/observable-test/`: use this pack to power a GSIS-style transition demo.

2 Review Sequence

Review order

1. **Intake packet** — assign packet identifier and reviewer.
2. **State framing** — write the current state and candidate next state.
3. **Guard evaluation** — test each required guard.
4. **Opacity & contradiction review** — determine whether structural traceability is broken.
5. **FMLIS crosswalk** — determine whether minimum learning and integrity controls are present.
6. **Threshold review** — summarize score / floor / release status.
7. **Decision record** — produce the result class and why.
8. **Conformance declaration** — state whether the packet is review-ready.

Discipline rule

When a transition produces irreversible impact, the verification threshold should increase proportionally to the difficulty of correcting the resulting state.

This is the operational discipline underlying the pack.

3 Packet Intake Sheet

Packet ID *required*

Reviewer *required*

Date *required*

Subject system / workflow *required*

Claim under review *one sentence*

Decision scope *what is being accepted, blocked, escalated, or deferred*

Stated threshold *release floor, caution floor, or other rule*

Source packet location *path, URL, or local archive reference*

Required proof objects

Mark each object as present, absent, or unknown.

Object	Present	Absent	Unknown
Claim statement			
Evidence items			
Source chain / provenance			
Timestamps			
Governing rule or policy			
Reviewer notes			
Correction / appeal record			
Decision output / affected state			

Immediate intake notes *what is obviously missing, malformed, or contradictory*

---

4 GSIS State-Transition Worksheet

Core framing

Current state S_t *what state existed before the claimed transition*

Event packet E_t *what event(s) or evidence packet is asserting a transition*

Candidate next state S_{t+1}^* *what new state is being claimed*

Transition function summary *how the packet says the current state became the candidate next state*

Classification target

Accepted state	Candidate next state is structurally supportable.
Blocked state	Candidate next state is not structurally supportable.
Unresolved state	Candidate next state cannot yet be classified due to missing or contradictory conditions.

Initial reviewer framing *one paragraph stating the attempted transition in plain language*

---

5 Guard Evaluation Matrix

Use one row per required guard. Add rows as needed. A missing or indeterminate safety-critical guard blocks final acceptance.

Guard ID	Guard description	Pass / Fail / Unknown	Safety-critical?	Evidence / notes
G-001	Provenance is complete and independently reviewable			
G-002	Evidence items are linked to the claim			
G-003	Contradictions are surfaced			
G-004	Contradictions, if present, are resolved or escalated			
G-005	Threshold floor is met			
G-006	Uncertainty is explicitly disclosed			
G-007	Decision is independently reviewable			
G-008	Record of decision is reconstructable			
G-009	Actor / enabling rule linkage is present			
G-010	Dependent transitions are not invalidated by impossible state			

Guard summary *which guard or guards control the final result*

6 Opacity and Contradiction Register

Opacity review

Mark each condition that weakens outcome-to-decision traceability.

Opacity condition	Present?	Notes
Long dependency chain		
Record fragmentation		
Guard indeterminacy		
Contradiction density		
Technical complexity without public traceability		
Missing actor or enabling rule linkage		

Opacity classification *Transparent / Traceable / Opaque / Non-Resolvable*

Contradiction register

Item ID	Contradiction	Resolved?	Resolution note or escalation path
C-001			
C-002			
C-003			

7 FMLIS Crosswalk

Use this section to verify whether the packet and the surrounding workflow satisfy minimum learning and integrity controls.

FMLIS control	Requirement	Status	Evidence / notes
FMLIS-010	Non-erasable adverse-event log / immutable audit trail		
FMLIS-011	Record of Decision linking inputs, rules, operator role, timestamps, and outcome		
FMLIS-012	Provenance retained with source, time, and chain-of-custody metadata		
FMLIS-020	Immediate redaction / anonymization before routing where applicable		
FMLIS-021	User-owned copy / receipt / hash of submission		
FMLIS-022	Pseudonymous case identifier		
FMLIS-030	Role separation: intake, evaluation, audit		
FMLIS-040	Insurability evidence / audit logs / corrective actions		

Crosswalk summary *Does the packet fail because the packet is weak, or because the surrounding system lacks minimum controls?*

---

8 MVS Threshold Worksheet

This is not a full MVS implementation. It is a compact review sheet so the packet can record threshold posture in a disciplined way.

Metric	**Weight** w_j	**Score** m_j	**Product**	**Notes**
Performance / correctness				
Calibration / fit to reality				
Fairness / parity				
Reproducibility / robustness				
Legal defensibility / transparency				
Cost / efficiency				

Final score *sum of weighted products*

Threshold class *Safe / Caution / Unsafe / Not enough information*

Threshold rationale *state whether release, restriction, recall, or more evidence is appropriate*

---

9 Decision Record

Result class *PASS / FAIL / INSUFFICIENT / UNCERTAIN — MORE EVIDENCE REQUIRED*

Plain-language result *one sentence*

Why this result was returned *three to seven short bullets*

Decision logic summary

- Which required proof objects were missing?
- Which guard failures controlled the outcome?
- Did opacity or contradiction prevent final supportability?
- Did FMLIS control failures materially limit trust in the packet?
- Did threshold review support release, restriction, recall, or deferment?

Independent review recommendation *can a second reviewer reproduce this result from the packet?*

---

10 Conformance Declaration

Reviewer declaration

I declare that this packet was reviewed as an attempted state transition, not as a narrative dispute. I recorded the current state, candidate next state, required guards, contradiction posture, opacity classification, and threshold posture before assigning the result class.

☐ Packet is review-ready for independent replication.

☐ Packet is not review-ready; missing objects or unresolved contradictions remain.

☐ Packet should be escalated for additional evidence.

☐ Packet should be preserved as a benchmark example.

Reviewer name *print*

Signature *optional*

Date *required*

---

11 Benchmark Packet Sample A

Purpose

This is a compact worked example to show how the public test page should behave.

Sample name	Sample A — Clean Packet
Current state	No adverse action executed; packet pending review.
Candidate next state	Adverse action or high-impact decision becomes structurally supportable.
Expected result	PASS
Why	Provenance complete, evidence linked, no unresolved contradictions, threshold met, decision independently reviewable.

Expected public-test behavior

When the user selects this sample on the website, the right-side result panel should update from the default placeholder state to:

- status = PASS
- title = Clean Packet
- summary = all required artifacts are present, linked, reviewable, and sufficient under the release threshold
- guard matrix = mostly green / satisfied
- reasons = present as a short bullet list

Why this matters

A non-updating demo destroys trust. The public page must visibly change when the sample changes.

12 GSIS Public Test Spec

Replace the generic checklist with a transition demo

The public test page should not feel like a generic form. It should visibly walk a reviewer through:

event packet → candidate next state → guard checks → transition class → accepted / blocked / unresolved state

Page contract

Left panel	sample packet selector + current state + candidate next state
Middle logic	required guards + contradiction / opacity indicators
Right panel	result class + why + next action
Downloads	standards PDF, worksheet pack, sample benchmark packet, answer key

Website integration recommendation

1. Keep `/observable/` narrow.
2. Keep `/standard/` as the standards directory.
3. List GSIS first as the parent structural lens.
4. Keep FMLIS, MVS, and TRA visible as distinct standards.
5. Publish TRS only when the PDF and worksheet pack are both real.
6. Rebuild `/observable-test/` as a GSIS transition demo, not a generic proof widget.

Standards order recommendation

- GSIS — umbrella structural integrity lens
- FMLIS — minimum memory, learning, insulation controls
- MVS — quantitative threshold and scoring layer
- TRA — transition-analysis instrument
- TRS — companion framework only when operational artifacts are live