SquadVault — Build Phase: Sprint 1 — First Engineering Sprint Plan

Execution-only sprint plan (no new features, no architecture changes) · Date: 2026-01-21

1. Purpose and Constraints

This document defines Sprint 1 for SquadVault Build Phase. The sprint operationalizes the locked contracts into small, boring, verifiable engineering tasks that can be executed across multiple chats without drift.

Sprint 1 objective: implement Writing Room Intake (v1.0) and Selection Set Schema (v1.0) as deterministic library code with unit tests, without wiring it into the existing weekly recap golden path yet.

- No new features.
- No architecture changes.
- No schema changes to existing storage tables.
- No behavior changes to the existing weekly recap lifecycle in this sprint.

2. Inputs Already Locked

- Signal Scout Contract (locked)
- Writing Room Intake Contract (locked)
- Selection Set Schema (locked)
- Engineering Handoff Checklist (locked)

3. Sprint 1 Scope

In scope:

- Create deterministic, tested library modules for Writing Room Intake + Selection Set Schema.
- Codify enum values, required fields, and determinism rules (ordering + hashing).
- Unit tests that prove: stable ordering, stable fingerprints, correct exclusion and withholding behavior (given deterministic inputs).
- Smoke checks proving imports and tests run in the current repo environment.

Out of scope (explicitly excluded):

- Any integration into the existing recap selection/render lifecycle.
- Any changes to canonical event selection logic or weekly window logic.
- Any ranking/scoring optimization or heuristic “smart” selection.
- Any UI/export changes.

4. Repository Reality (Observed)

Observed from current repo scan:

- Existing weekly recap lifecycle + debug tooling already exists (scripts/recap.py, scripts/check_golden_path_recap.sh, scripts/golden_path_recap_lifecycle.py).
- withheld_reason is already persisted in recap_artifacts and normalized in src/squadvault/core/recaps/recap_artifacts.py.
- No existing code matches Writing Room / Selection Set naming beyond withheld_reason usage in artifacts and scripts.
- Tests exist in both Tests/ and tests/ (capitalization differs); do not invent a new test runner.

5. Sprint 1 Deliverables

- D1: Selection Set Schema v1.0 (code)
- D2: Writing Room Intake v1.0 (code)
- D3: Unit tests for determinism + intake rules (code)
- D4: Golden-path regression check evidence (command output saved by operator)

6. Task Breakdown (Exact, Boring, Verifiable)

All tasks are intentionally small. Complete tasks in order. Each task has a verification step that must pass before moving on.

Task	Description	Files / Touchpoints	Verification	Notes
T0	Repo hygiene: ignore scratch outputs used during sprint	.gitignore	git status -sb (clean)	Keep diffs focused
T1	Create module skeletons for writing_room	src/squadvault/recaps/writing_room/__init__.py; selection_set_schema_v1.py; intake_v1.py	PYTHONPATH=src python -c "import squadvault.recaps.writing_room.selection_set_schema_v1"	No functional changes yet
T2	Implement Selection Set Schema v1.0 (dataclasses/enums, deterministic ordering, SHA256 fingerprint helpers)	selection_set_schema_v1.py	Unit tests: ordering + fingerprint determinism	Must match locked schema and enum names
T3	Implement Writing Room Intake v1.0 using	intake_v1.py	Unit tests: confidence gate, lineage gate,	Selection discipline only

	extractor/adapters (no Signal schema invention)		window gate, withheld cases	
T4	Add unittest suite for selection set + intake	Tests/test_writing_room_selection_set_v1.py (or consistent existing tests location)	python -m unittest discover ... (green)	Use existing test conventions
T5	Golden-path regression: prove existing recap pipeline unaffected	scripts/check_golden_path_recap.sh (no edits)	./scripts/check_golden_path_recap.sh --db ... --league-id ... --season ...	Must match pre-sprint behavior

7. Correct Implementation Order

1. T0 — Repo hygiene
2. T1 — Module skeletons + import smoke check
3. T2 — Selection Set Schema v1.0 + determinism helpers
4. T4 (partial) — Tests for ordering + fingerprint determinism (locks determinism early)
5. T3 — Writing Room Intake v1.0 (extractor-driven)
6. T4 (complete) — Tests for intake gates and withholding behavior
7. T5 — Golden-path regression run

8. Golden-Path Validation Steps

These validations are mandatory. Do not declare sprint complete until all are green.

- V1: Import smoke test: Writing Room modules import under PYTHONPATH=src.
- V2: Unit tests: deterministic ordering + hashing + intake gating pass.
- V3: Regression: existing weekly recap golden path passes unchanged.

Recommended commands (copy/paste):

```
PYTHONPATH=src python -c "from squadvault.recaps.writing_room.intake_v1 import build_selection_set_v1; print('ok')"
```

```
PYTHONPATH=src python -m unittest discover -s Tests -p "test_*.py" -v
```

```
./scripts/check_golden_path_recap.sh --db .local_squadvault.sqlite --league-id 70985 --season 2024
```

9. Explicit DO NOT INVENT Reminders

Sprint 1 is contract implementation only. The following are forbidden in this sprint:

- Do not invent a Signal schema inside Writing Room. Treat signals as opaque objects accessed through extractor/adapters passed into the intake function.
- Do not add ranking/scoring or optimization. No heuristics beyond explicit gates from the contract.
- Do not modify existing recap lifecycle behavior, storage schema, or selection_fingerprint semantics used by weekly recap artifacts.
- Do not auto-group signals unless deterministic identifiers are provided upstream. Groupings may be empty by default.
- Do not introduce new exclusion/withhold reason codes. Use the locked enums exactly.

10. Definition of Done

- All Sprint 1 deliverables (D1–D4) are present in the repo and committed.
- Unit tests pass locally via the repo's existing test runner.
- Golden-path recap validation passes with no behavior change from pre-sprint baseline.
- Selection Set Schema v1.0 produces stable ordering and stable fingerprints across repeated runs.
- Writing Room Intake v1.0 produces deterministic included/excluded/withheld outputs given deterministic inputs.

Appendix A. Suggested File Locations

- src/squadvault/recaps/writing_room/selection_set_schema_v1.py
- src/squadvault/recaps/writing_room/intake_v1.py
- Tests/test_writing_room_selection_set_v1.py (or match existing tests directory convention)

Appendix B. Operator Evidence Capture

When the sprint completes, save command output (V1–V3) into a single timestamped text file and commit it ONLY if your repo policy allows evidence artifacts. Otherwise, keep it local for review.