

Making AI work accountable.

A practical operating system for AI-assisted work, and an experiment workflow that lets evidence refuse a candidate.

Shed Happens is a wildlife data company that ships an app. Its operating system coordinates work behind field-data collection, wildlife intelligence and measurement tools. The strategic filter is: **does this grow the dataset?**

Construction discipline, encoded.

Chuck brought a construction-management structure to AI work: a written scope, assigned labor, proof, separate inspection and owner authority.

Work sequence	Record / boundary
Scope / probe & build	Goal, allowed changes, acceptance criteria and stop conditions.
Evidence / review	Raw outputs stay with the work; FINAL review checks scope and proof.
Delivery / memory	Consequential actions require founder approval; verified findings reach company records.

Who did what

Chuck directs the work, defines measurement rules, reviews labels and makes operating and promotion decisions. The implementation was built with Claude and Codex. This exhibit was prepared with AI assistance from project records.

The architecture makes work inspectable. It does not make every judgment correct. An early audit found same-family self-review behind different role labels: identity verification matters alongside role separation.

Approval belongs to one request.

A saved July reproduction shows one top-level approval incorrectly clearing two unrelated pending requests. The repair bound the decision to the request thread.

Synthetic fixture	Before repair	After repair
Two pending requests	A and B	A and B
Reply intended for B	A and B could both resolve	B resolves; A stays pending
Stray top-level reply	Could clear both	Neither resolves

What changed

Later work added sanctioned action execution and signed, single-use approval handling. Stored tests address request selection, later rejection, incomplete thread reads and untrusted approval records.

What the evidence can claim

The saved fixtures show the original behavior and the corrected cases. They were inspected for this exhibit, not rerun. The display normalizes the stored outputs into request A/B labels. These are synthetic machinery tests, not customer records. [E1-E2]

Keep the authority with the owner

Production changes, external distribution, new spending and public or mass communications require specific founder approval. A role label or an agent-authored statement of approval is not human authorization.

This is a specific regression and repair story. It is not a universal security guarantee. Private implementation paths, infrastructure identifiers and original operating records remain outside this exhibit.

Let the test refuse the candidate.

APEX combines AI-drafted labels, human correction, certified 3D measurements, training and controlled evaluation. Records report 70 human-approved pilot elk labels.

A new-scan test

This historical diagnostic illustrates evaluation, not validation of the current model. A tip proposer looked promising on seen scans. Its code and acceptance rule were frozen, then tested on **26 deer/elk sides from 18 cards**. Two antelope sides were reported separately. [E3]

Decision measure	App	Candidate
Independent tine tips within 0.5 inch	21/41(51.2%)	26/41(63.4%)
Median stray proposals per side	0	1

Not adopted.

The rule required at least a 10-point improvement **and** no increase in median strays. The second condition failed. The estimated improvement was **+12.2 percentage points**, with a 95% interval of **-3.2 to +29.3**, spanning no improvement.

Reference labels can also be biased

Some taps were exact copies of the app’s visible proposals. The report separated independently placed taps so the evaluation did not simply reward the app for anchoring its human reference. The candidate’s attractive headline was kept with the failed rule and uncertainty.

Current limits

An elk learned-lengths diagnostic reached 28/62 within five inches versus 26/62 for its baseline, without a clear established gain. A later October 27 entry records that Chuck stopped the retrain during rendering: no model was produced. A same-day audit reported training/test overlap and missing retained renders in the earlier lineage. Chuck held training and ordered a reset from public base weights, with every training image retained. These are attributed audit findings, not an independent re-audit here. Earlier training is treated as unproven. [E4-E5]

Measure the work. Reduce the waste.

The operating rules address session length, context size, model selection and weekend batch scheduling. The usage report keeps a cost estimate alongside the work.

Period	Reported API-equivalent Claude estimate
September 2026	\$12,633
October 1-27	\$4,071
June-October 27	Approximately \$26,300

Stated subscription cost: \$200/month. Codex usage is separate. These are counterfactual estimates, not invoices or realized cash savings. The monthly rows sum to \$26,302; the arithmetic was checked. Raw usage events and pricing assignments were not independently reconstructed. [E6]

An efficiency claim needs an output denominator

The report associates new rules with lower usage rates, but does not isolate causation or hold workload and quality constant. Most processed tokens were cache rereads. Useful measures are accepted labels, validated experiments and certification minutes per unit of usage.

Evidence and limits

- E1-E2: July approval-repair fixtures and July/September hardening records.
- E3: October 20 frozen-code tip-proposer fair test and per-tap outputs.
- E4: October 20 learned-lengths diagnostic, leave-one-antler-out on 62 elk.
- E5: October 20-27 APEX planning, progress, data-flow and retrain records.
- E6: October 27 user-supplied local-log cost estimate.

The October 24 scheduled run missed its start; the new watchdog was built but not installed when checked. Historical experiments were not rerun, and model quality and production state were not revalidated. This exhibit was prepared offline; public sharing was authorized separately. No training or changes to product production systems were performed.

Private user pins, coordinates, customer records, repository paths and infrastructure details are excluded. Originals and source hashes remain in a separate private manifest. Public context: shedhappensllc.com and the [founder page](#).