COLDCODE SI LABS LIBRARY WORKING PAPER 2026-04

The Mechanism Audit: Evidence-First Review for Products That Act Without an Operator

Courage Garikai
Coldcode Si Labs Library · Harare

November 2026

Abstract

Software reviews were built on the assumption that a human performed each action and a second human watched. That assumption has failed quietly. Products now change themselves: agents commit, workflows publish, models write the copy, and the operator's role is reduced to reading a green tick. The instruments of review — the demo, the screenshot, the QA script, the changelog — were designed to certify **effort witnessed**. They cannot certify **a mechanism governed**, because nobody was in the room when it acted.

This paper sets out a method for reviewing that class of product, and the evidence for it comes from a single estate of seven ventures run by one operator and a set of agents. The audit asks for five artefacts: a **claim ledger** (what the product says it does), a **mechanism map** (what actually decides), an **evidence trail** (what the mechanism leaves behind when it runs), a **detector** (a check that fails when the mechanism stops working), and a **response** (what happens when it fails, and who is told). A product is audited when all five exist, agree, and can be re-run by a stranger.

THE MECHANISM AUDIT

Working Paper 2026-04

Copyright © 2026 Courage Garikai.

Published by Coldcode Si Labs Library, Harare, Zimbabwe.

First edition, 2026.

All rights reserved. No part of this publication may be reproduced, distributed, or transmitte
form or by any means without the prior written permission of the publisher, except in the ca
quotations in a review or certain other noncommercial uses permitted by copyright law. The
frameworks, worksheets and checklists in this publication are meant to be used, copied and
by the reader.

This is a work of analysis and practical guidance. It is not financial, legal, investment or pro
advice, and no outcome is promised: results depend on the reader's own market, effort and
Platform behaviour and figures were checked at the time of writing and change.

COLDCODE SI LABS LIBRARY · Harare, Zimbabwe

Six failure signatures are formalised, each with a production incident attached from the same week: the green run that names no subject; the allowlist that excludes what it was built to catch; the credential that authenticates but cannot act; the retry that published twice; the fallback that looked like output; and the origin that had not yet propagated. They collapse into six gap classes — claim–evidence, scope, capability, ownership, propagation, attribution — which is what a review is actually looking for.

The contribution is architectural and forensic, not statistical. The sample is one estate, the author is its owner, and the incidents were cheap. That is disclosed here rather than argued around: what is claimed is that the method found the failures before they cost money, and that every detector described is in a public repository and can be run against this sentence's author.

Keywords: product audit; mechanism governance; silent failure; evidence trail; agentic software; deploy verification; small-operator infrastructure; sub-Saharan digital infrastructure

1. The review that no longer fits

A demo is a claim made in the presence of a witness. For thirty years that was enough, because the software could not do anything the demonstrator did not do. The witness could ask "show me again", and the second showing was evidence, because a human had to be involved for it to happen.

Three changes broke the instrument:

1. **Authorship moved.** The code that ran in production was written, in part, by a system that was not in the meeting. The reviewer's question — *did you build what you said?* — has no witness to answer it.
2. **Action moved.** Products act on their own schedule: they publish, invoice, email, deploy. The interval between reviews is now full of acts that nobody watched.
3. **Success became a colour.** Continuous integration turned verification into a tick. A tick is a *statement about a run*, not a statement about the product, and the two are routinely confused — including by the people who configured the pipeline.

The consequence is specific and it is not "AI is risky". It is that **the evidence a reviewer needs is no longer produced by the process that used to produce it**. Nothing about the demo is dishonest; it is simply measuring a world that has gone.

This paper describes an audit built for the world that replaced it. The audit was developed while operating seven ventures — a desk system for customs and tender firms, a document tool, a storefront kit, an audit practice, a mechanism record, an open teaching practice, and a press — from one desk in Harare, with agents doing the work and one person accountable for it. The incidents quoted are

not case studies assembled afterwards. They are this week's failures, and each one is still in the repository's history, detector included.