



PRODUCT DESIGN · TECHNICAL ANTHROPOLOGY · BROOKLYN

leuan King akaBuild

A product designer and technical anthropologist working on the human side of applied AI — discovery, approval flows, and the interfaces that make agents legible and worth trusting. I build tools and create art.

akabuild.dev · leuan@yionvisual.com · linkedin.com/in/leuan-king · github.com/akaieuan

WHO I AM

The human side of applied AI.

I design and build the surfaces where people and AI systems meet: discovery and research flows, approval and review interfaces, evidence and citation UX, and the copy conventions that make agentic tools feel legible. I prototype in code, validate with real users, and ship production front-end alongside the design work.

Alongside design: agent skill-building and testing, custom evaluation work, front-end engineering, and procedural 3D. I also produce and perform electronic music — some of the tools in this portfolio exist because I needed them on stage.

FOCUS

PRODUCT DESIGN

HITL & AGENTIC UX

USER RESEARCH

FRONT-END

AGENT EVALUATION

PROCEDURAL 3D

3.5 yrs

UBIK STUDIO

2

STUDIOS RUN TODAY

20+

SHIPPED PROJECTS

IN THIS PORTFOLIO

- 01 Ubik Studio — a desktop AI research platform, 2023–2026
 - 02 Circleheads (co-founded) & akaOSS (independent) — the studios
 - 03 Box Populi — client work for a NYC live-techno collective
 - 04 akaCOVART — a generative album-art studio, live
 - 05 akaVSTs & Hologram — instruments & agent observability
- Practice & contact

CURRENT WORK



STUDIO · APPLIED AI · BROOKLYN

Circleheads · circleheads.com

The two-person applied-AI software studio I co-run — applied AI in production, a small consulting bench, and original games on the side.

Applied AI — agentic systems that do real work.

Consulting — a short bench, senior only.

Games — strange things, quietly built.



STUDIO · OPEN SOURCE · HITL AI

akaOSS · akaoss.dev

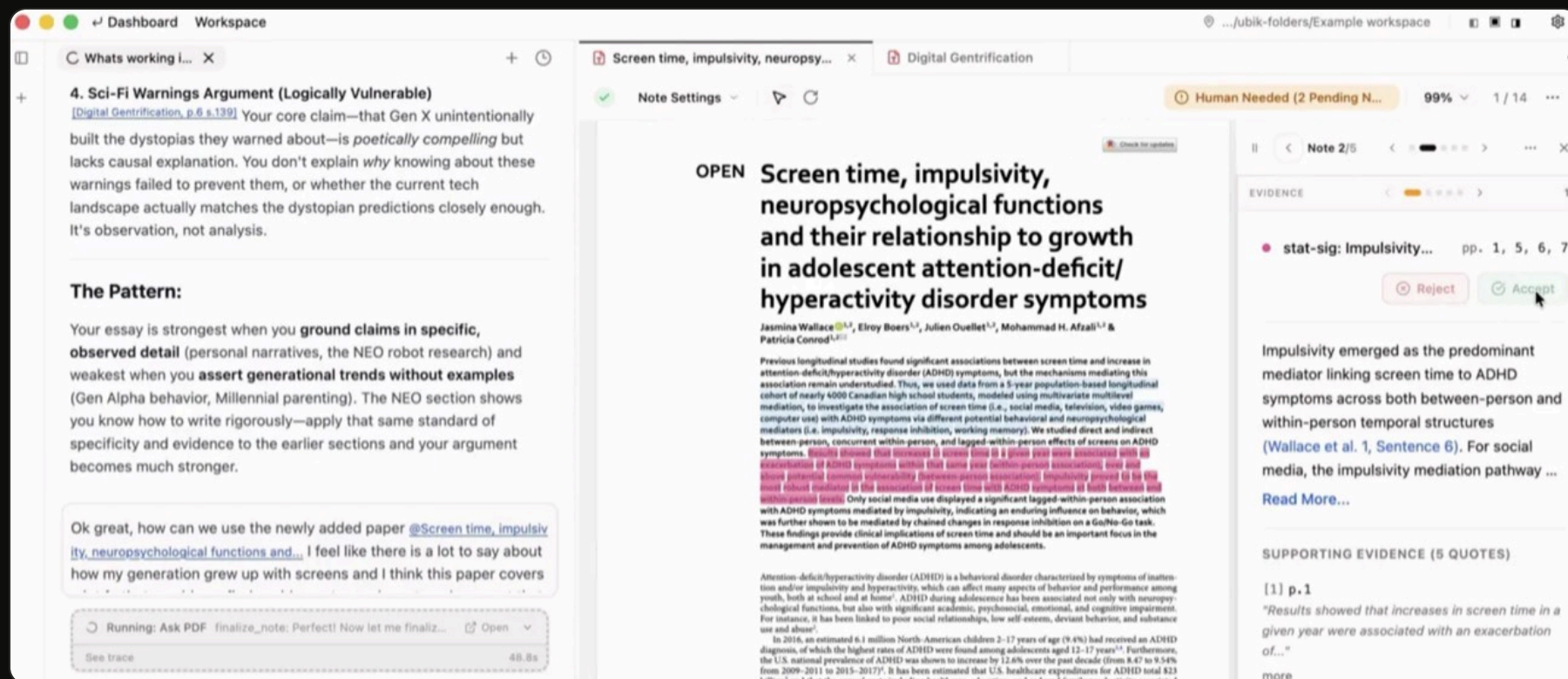
My independent open-source studio — software for human-in-the-loop AI: a measurement family for AI that assists without displacing, and tooling for the agents that build alongside you.

HITL Kit — HITL AI, measured properly · shadcn CLI + npm

eval-kit — measurement for multi-step agents

tag-kit · **Collapse** · **Hologram** — annotation & agent tooling

Ubik Studio

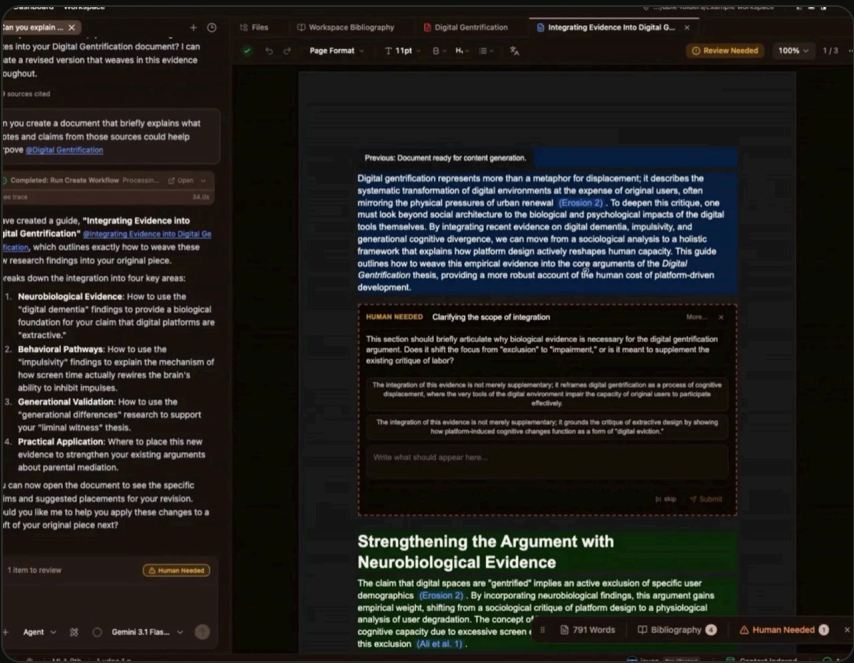


The three-pane workspace — agent chat on the left, the source paper in the middle, and an evidence panel on the right where each extracted claim is accepted or rejected by hand.

Three and a half years co-founding a desktop-native AI research platform. Any folder became a research workspace: agents searched the literature, read sources in parallel, and drafted documents where every claim traced back to a real quote on a real page — and everything of consequence passed through a human first.

I led product design end to end — the workspace model, review surfaces, evidence and citation UX — built front-end throughout, ran the user research cycles, and designed and built the custom evaluation framework and ARC eval suite we used to train and tune our agents and orchestration systems, with direct, measurable effects on output accuracy, quality, and real-world usability.

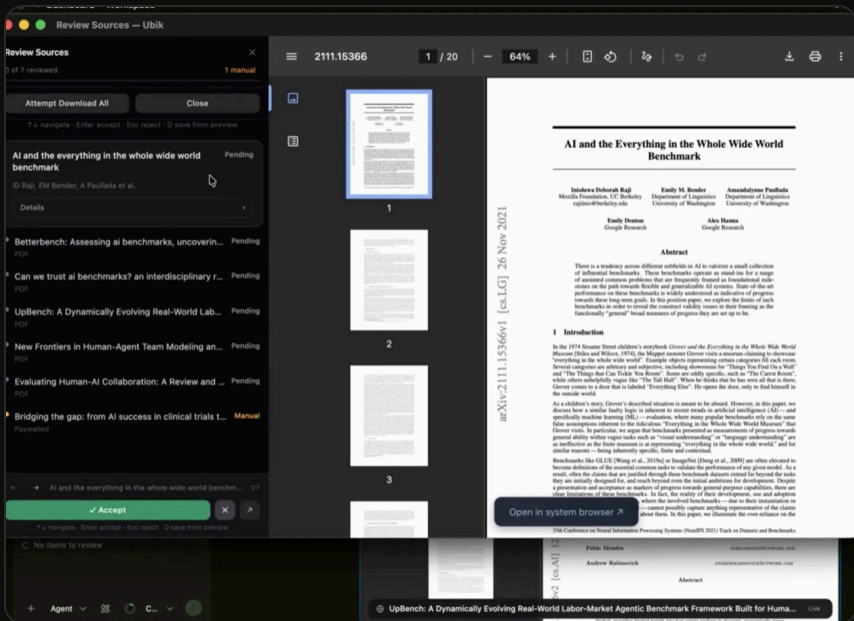
SELECTED WORK · 01 — CONTINUED



An agent pauses mid-document and asks for human judgment exactly where it belongs — with candidate drafts and a free-write.

"Your job is not to replace human thinking — it is to amplify it. Optimize for the loop: you draft, the human refines, you incorporate, the human approves."

— from the product's own agent design



The source review queue — every flagged paper is previewed in full and human-approved before it enters the workspace.

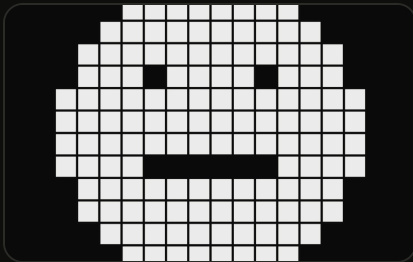
Human control wasn't a confirmation dialog bolted on at the end — it was load-bearing architecture. Agent actions were approved in batches; every review decision left an auditable trail; and if there was no evidence to cite, the agent didn't get to write the claim.

Circleheads

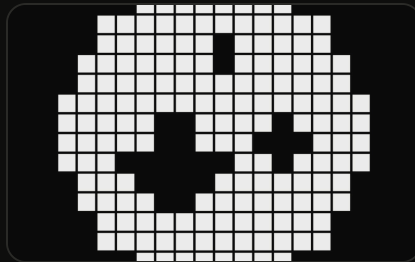
Co-founder · Applied-AI software studio · circleheads.com

We watch the work first, then ship agents that do it in production — with the data they need, the skills they use, and approval gates that keep humans in control. Plus a short senior consulting bench and original games built in-house.

- **My hand:** product design and technical anthropology, plus skill-building, agent testing, front-end, and procedural 3D.
- **Brand system:** I designed and built the studio's generative pixel identity — a canvas animation engine shared across our sites.



The circlehead — brand engine still



The akaOSS spark — same pixel language

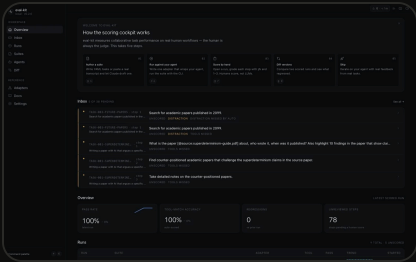
APPLIED AI CONSULTING GAMES

akaOSS

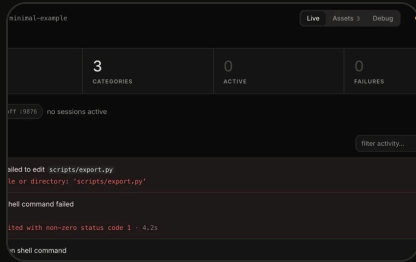
Founder · Open-source studio for HITL AI · akaoss.dev

Five open-source projects served as one site, built on one thesis — Assist-Not-Complete: evaluate AI on whether it assists humans without displacing them, not on whether it can finish the task alone.

- **HITL Kit** — 15 React primitives for human-in-the-loop UI, installable via the shadow CLI, with a perspective paper.
- **eval-kit** — a measurement instrument for multi-step research agents: humans score, not LLMs.
- **Collapse & Hologram** — developer tools for Claude Code skills and live Blender→gITF pipeline observability.



eval-kit — run overview



Hologram — live pipeline dashboard

OPEN SOURCE HITL AI RESEARCH

Box Populi

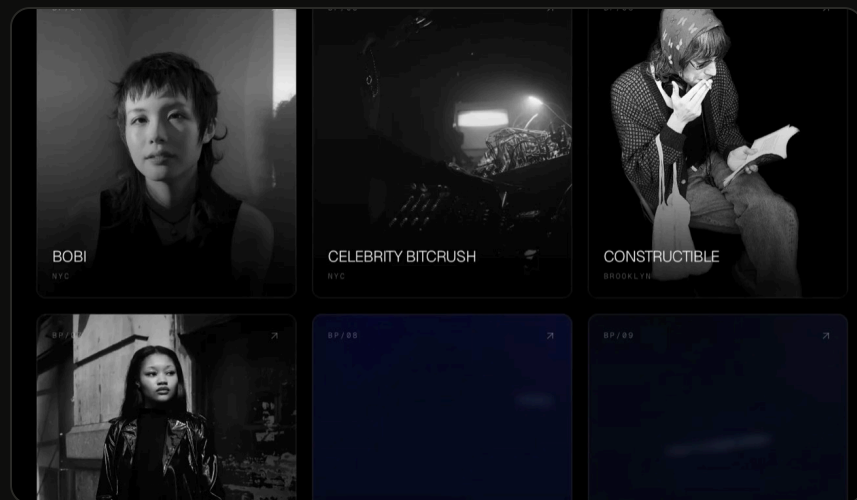


The landing — a club-flavored home for a NYC live-techno improvisation collective, with a persistent header player that keeps the latest set playing across every page.

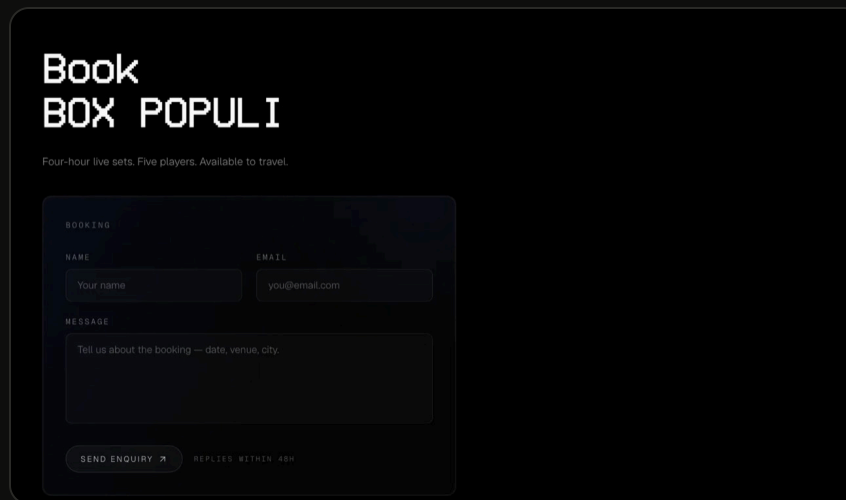
A client project for Box Populi, a rotating cast of NYC artists playing continuous, multi-hour improvised techno. The brief: a fast, dark, on-brand public face for the crew's roster, live sets, and bookings — built so the client can run it themselves. (I also play in the collective, which made the client feedback loop unusually direct.)

The whole site is roster-driven: a typed data model generates the landing page, the crew grid, and a profile page for every artist, so the collective can grow without anyone touching layout code. Custom on-brand players run over the official SoundCloud Widget API — dark glass cards with our own transport, SoundCloud as the sanctioned host.

Honest engineering: a module-level audio bus pauses every other player when one starts, and the header set survives navigation. iOS only starts audio in a synchronous tap — so touch devices get a one-time "Double Tap" hint instead of a hack. No user-agent sniffing; every capability check is CSS.

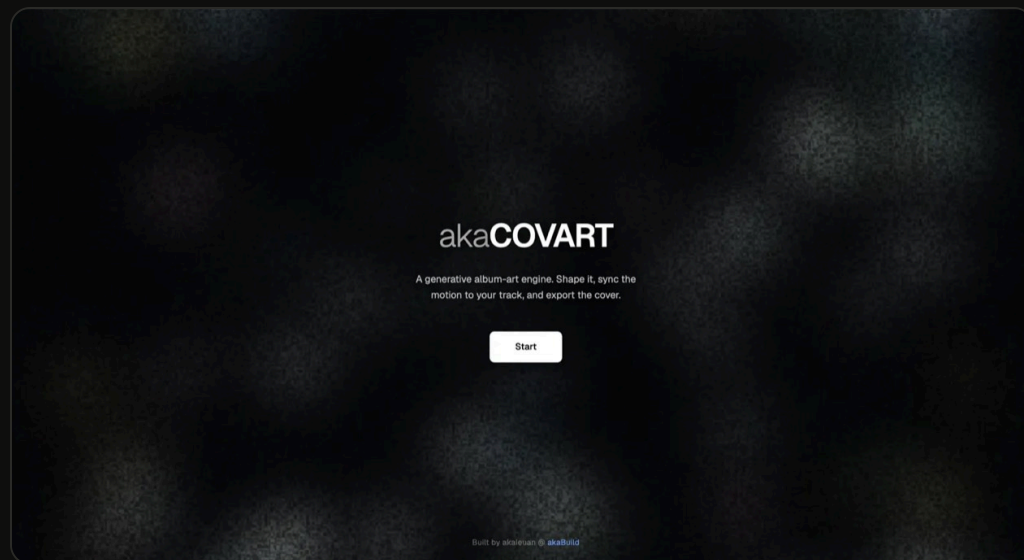


The roster grid — every card generated from a one-line data entry, so the crew grows without layout work.

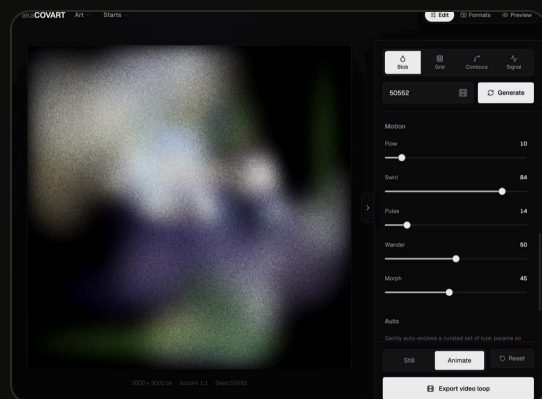


The booking pipeline — "four-hour live sets, five players, available to travel," backed by a transactional-email enquiry form.

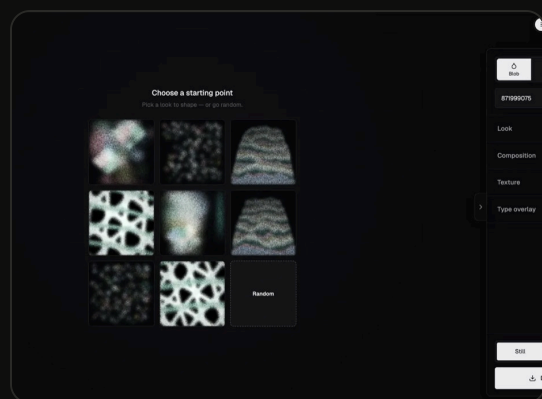
akaCOVART



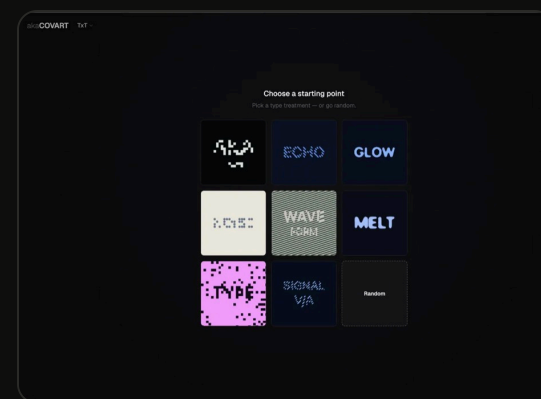
The studio's front door — a generative album-art engine, live at akacovart.com.



The Edit view — a Blob cover with its Motion rack (Flow, Swirl, Pulse, Wander, Morph) and a one-click video-loop export.



Art focus starting points — curated looks across the Blob, Grid, Contours, and Signal engines.



TxT focus — the title becomes the subject: Dither, Lines, and Blur type engines with per-glyph composition controls.

SELECTED WORK · 05

akaVSTs

Three instrument plugins · JUCE / C++ · VST3 & AU



akaBLEEP — acid synth with a 64-step P-lock sequencer.

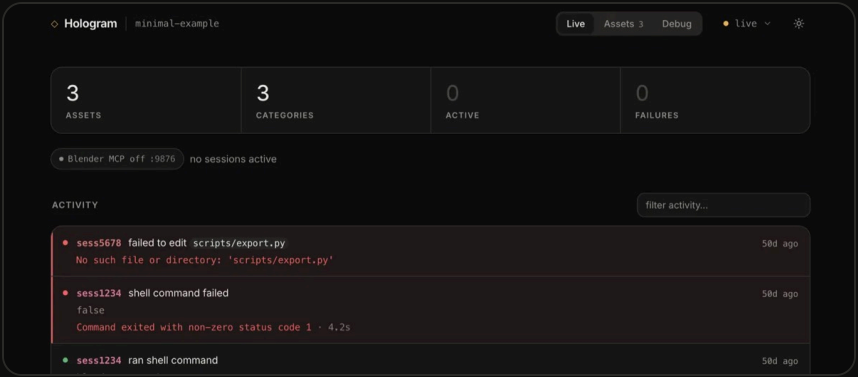


akaENZYME — four-layer lo-fi synth with shared voice pool.

Instruments I play live and use in my own music — built for live recording with dedicated FX and performance-focused controls, not preset-browsing.

Hologram

Agent observability for 3D pipelines · v0.6.0 on PyPI · MIT



The live dashboard — the agent's pipeline activity next to the assets it produces.

My AI coding agent writes the Blender scripts and runs the glTF exports for the games I build — and I had no idea what it was doing. Hologram closes that gap: it tails one event log and streams the agent's live activity — edits, shell commands, exports — next to the assets those actions produce, in one local dashboard.

Then it hands the same pipeline back to the agent as MCP tools — enumerate, introspect, health-check, and a non-destructive render through live Blender, so the agent can see an export, not just count its nodes. Deliberately read-only; it never modifies an asset. Stdlib Python, no build step, no database.

PRACTICE

How I work.

Watch the work first.

Mixed-methods research — structured interviews, behavioral observation, and session replays — synthesized into prioritized UX decisions rather than slide-deck summaries. Every cycle is anchored in trust, evidence attribution, and meaningful human control, and the findings feed interaction specs, flow changes, system prompts, and microcopy in the same shipping rhythm as the product. The validation logs stay public.

Prototype in code, at scale.

I turn intent into high-quality working surfaces — web and desktop — fast enough to stay aligned with the people using the product. The same weeks usually include interviews, screen replays, and light data analysis, so the next build is shaped by what we're seeing in the field, not by a static milestone.

Measure what matters.

Most AI is judged on autonomous completion while real deployment is collaborative. I evaluate agents on whether they assist people without displacing them — in the product, in the copy, and in how the systems behind it are trained and tested. The long-form version of that argument is my paper, An AI Measurement Problem, at kraa.io/akaieuan.

CONTACT

akabuild.dev

ieuan@yionvisual.com

linkedin.com/in/ieuan-king · github.com/akaieuan

Full write-ups, interactive demos, and live projects at akabuild.dev/demo.