



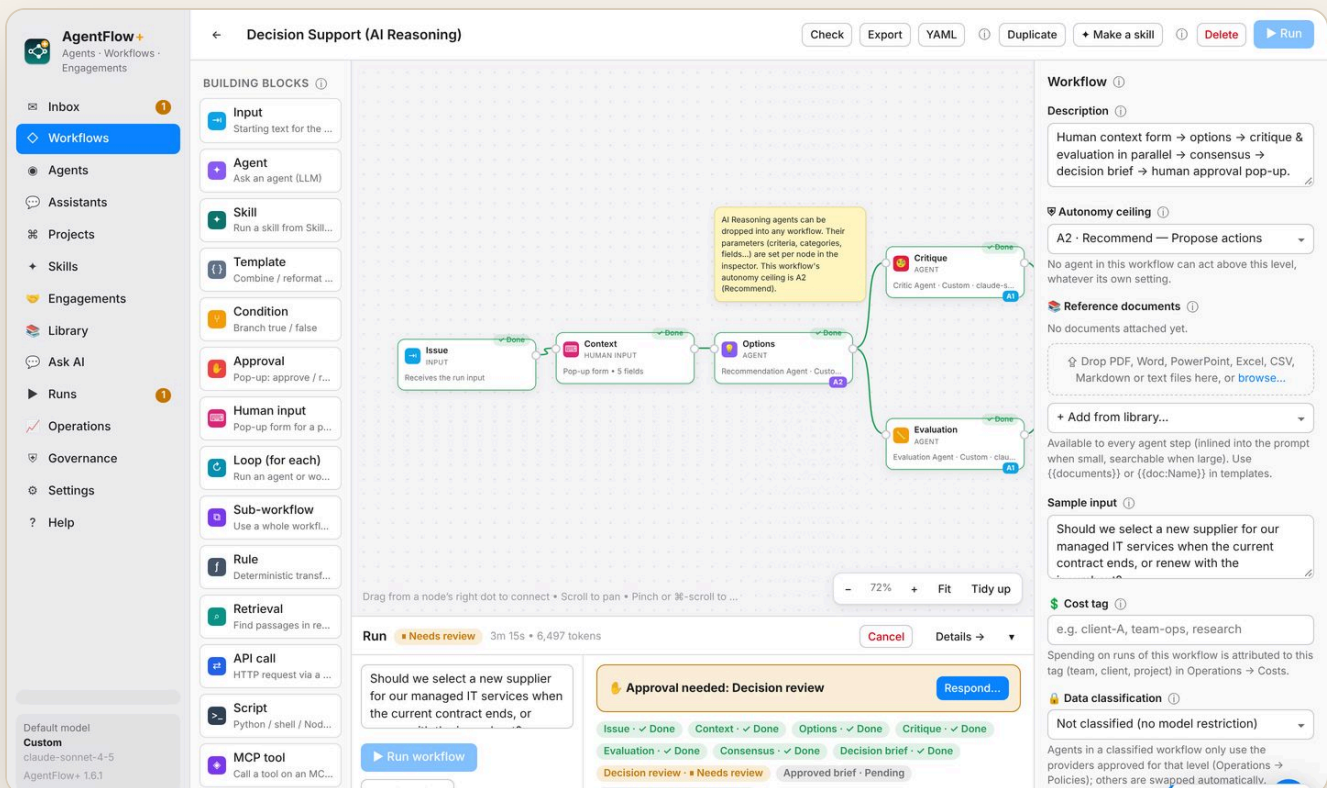
AgentFlow+

AI AGENT WORKFLOW MANAGER FOR MACOS · VERSION 1.6

Build, govern and run AI agents on your Mac.

Visual workflows, 25 ready-made agents, a built-in Skill Studio and governance enforced at every step — in one local application. Nothing leaves your Mac unless you decide it should.

US\$10,000 per year · single-user licence



A chat is not a system

Prompts that work once are not processes a client can rely on. AgentFlow+ turns them into durable, repeatable, governed workflows — with a history you can show to an auditor. Every AI project has to answer three questions; AgentFlow+ is built around them.

What did it do?

Durable, typed runs. Every step is saved as it completes; outputs are validated against a JSON schema; runs resume after a crash and can be re-run from any step; results export to Word with the full audit trail.

Who approved it?

Enforced autonomy and bound approvals. Every agent, step and workflow carries an autonomy ceiling. Approvals record what was shown and what was approved; a tampered payload is refused; external writes never repeat on retry.

Can we prove it?

Evidence by default. A hash-chained event log, lineage of the workflow, agent and skill versions used by every run, risk cards, evaluation results and a Word handover document.

Seven capabilities, one application

1 Visual workflows

Branches, parallel steps, loops, sub-workflows, rules, retrieval, API calls, scripts, MCP tools; typed outputs, retries, fallbacks, run limits and cost estimates.

2 Agents and assistants

25 agents in seven categories, including a 15-agent AI Reasoning library; any agent becomes a conversational assistant with its tools, documents and skills.

3 Governance

Six autonomy levels, payload-bound approvals, policy as code, reviewer roles, risk cards, environments dev / test / prod, emergency stop, tamper-evident log.

4 Skill Studio

Describe → Specify → Generate → Edit → Test → Release; trials and baselines; security scan; release kits for Claude, Agensi.io, SkillsMP and AI Skill Market.

5 Engagements and assurance

Scored use cases, evaluation datasets, prompt-injection suite, release gates, shadow mode, bilingual Word handover and customer export.

6 Operations

Dashboard with cost per successful outcome, triggers, HTTP connectors, MCP servers, incidents, agent memory, policies by data classification, retention.

7 Project Hub and local-first design

Claude Code and Cowork project folders opened in place; one data folder on the Mac, Keychain, per-launch token, no telemetry; Anthropic, OpenAI-compatible or Ollama.

Workflows that survive anything

Workflows are drawn on a canvas. Each block is an agent, a skill, a template, a condition, an approval, a human-input form, a loop, a sub-workflow, a deterministic rule, a retrieval step, an API call, a script, an MCP tool or a Claude Code task. The inspector sets the workflow's autonomy ceiling, reference documents, cost tag and data classification.

Typed outputs

Any producing block can declare a JSON output with a schema. Malformed output fails the step, so retries, fallback text or a fallback agent apply before anything downstream runs.

Durable runs

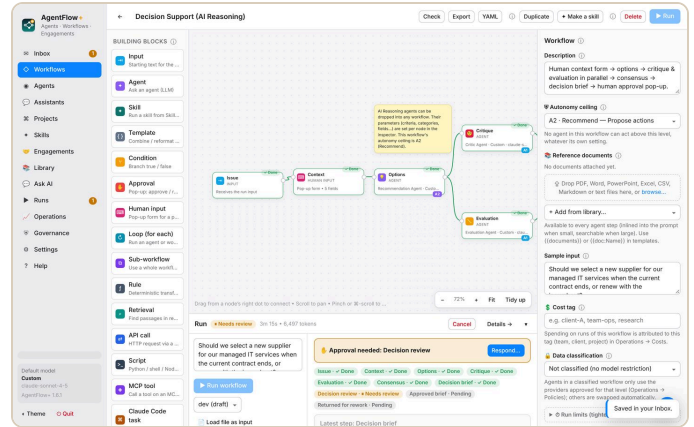
Every completed step is saved at once. An interrupted run shows Resume and continues where it stopped. Re-run from any step keeps earlier results; safe replay runs the same input with actions disabled.

Limits and cost

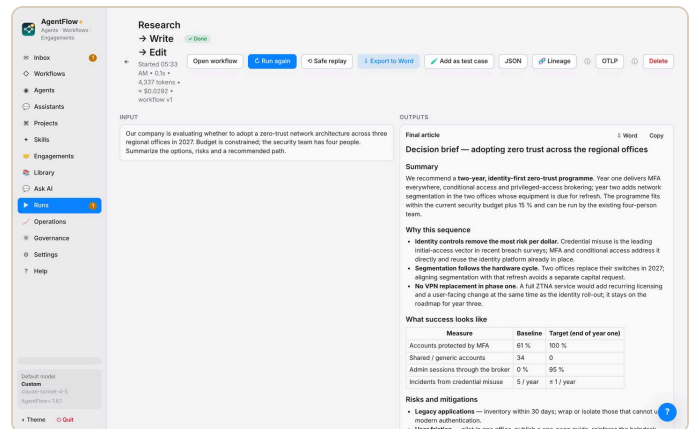
Minutes, model calls, tool calls, tokens and spend per run — global defaults tightened per workflow. Cost is estimated from an editable price table and attributed to a cost tag.

Export

Results or a full report (every step, human decisions, governance audit) to Word; JSON and OpenTelemetry export for your own tooling.



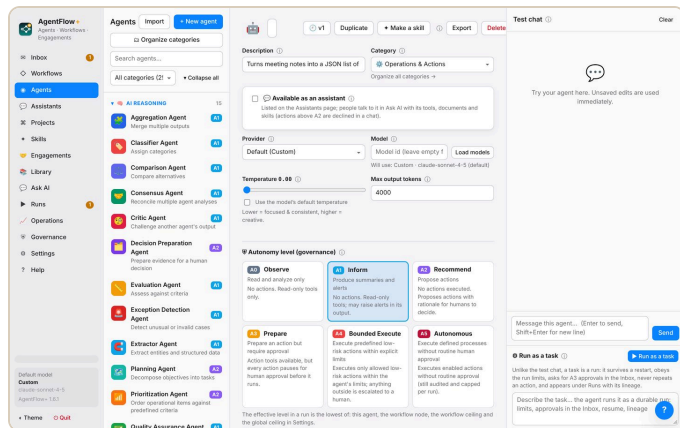
The workflow canvas: building blocks on the left, the inspector with the autonomy ceiling on the right, a live run at the bottom.



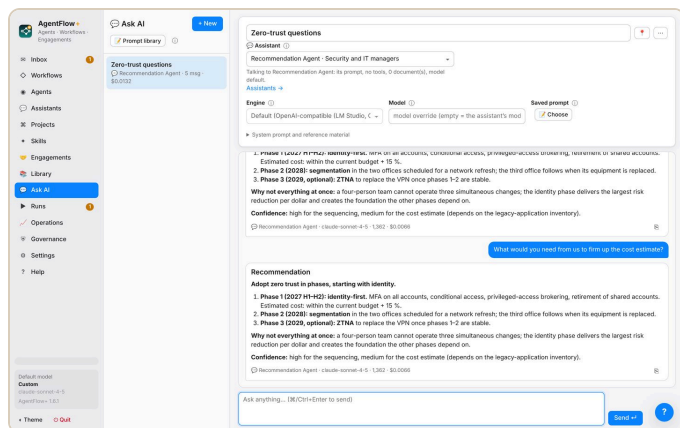
A run page: input, outputs, human interactions, steps with their autonomy badges, lineage and Word export.

Agents and assistants, ready on day one

AgentFlow+ ships with 25 agents in seven categories — AI Reasoning, Research & Writing, Analysis & Decision, Review & Quality, Operations & Actions, Teaching & Grading and General. The 15-agent AI Reasoning library covers classification, extraction, comparison, critique, consensus, verification, planning, prioritisation, quality assurance, recommendation and more; each has parameters, a default autonomy level and an output format.



The agent list, with the autonomy level picker in the editor.



An Ask AI conversation with an assistant persona; the trace and cost sit under each answer.

Per-agent settings

Provider and model (or routed by tier with failover), temperature, tools, reference documents, pinned skill releases, autonomy level, iteration cap, completion criteria.

Assistants

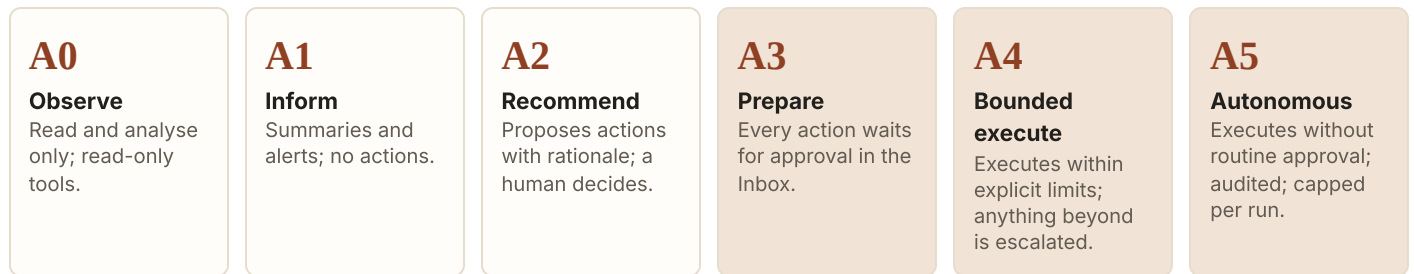
Tick “Available as an assistant” and people talk to the agent in Ask AI with its prompt, tools, documents and skills. Every answer shows its tool calls, skill releases and cost; every turn is a job with Stop; questions back to the user end the turn with one-click options. Chats are capped at A2.

Agent tasks

“Run as a task” turns a standalone request into a durable run with limits, Inbox approvals for A3 actions, stall detection on repeated identical tool calls and DONE / BLOCKED completion criteria.

Governance that is enforced, not written down

Autonomy is the central control. The effective level of any step is the lowest of the agent's level, the node's level, the workflow ceiling and the organisation-wide ceiling. Above the ceiling, nothing runs.



Approvals bound to the payload

An approval records what was shown and what was approved (hashes). You can edit the content before approving — the edited version is what runs. A changed payload is refused by the dependent step.

Idempotent external actions

API and MCP writes, file writes, notifications and webhooks never repeat on retry or resume; a second run of the reference workflow reports "alreadyExisted".

Policy as code

Rules with conditions (classification, block types, providers, tools, environment) and effects (cap autonomy, deny providers or tools, require classification, production needs a released version). Four sensible defaults ship.

Reviewers, roles, environments

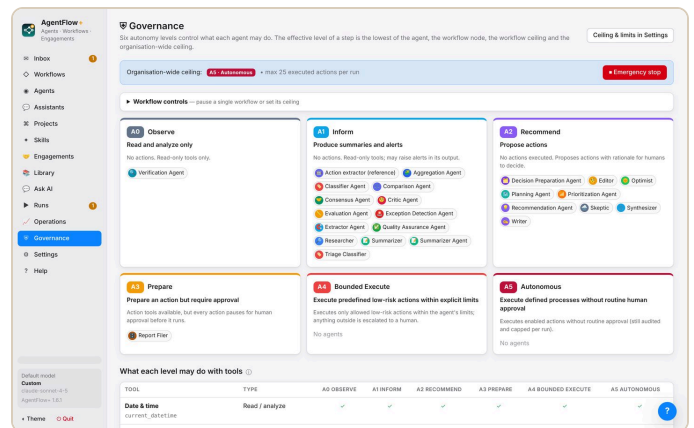
Approver, releaser, auditor and operator roles with delegation; dev / test / prod promotion with rollback; production requires a released version and the releaser role.

Tamper-evident log and emergency stop

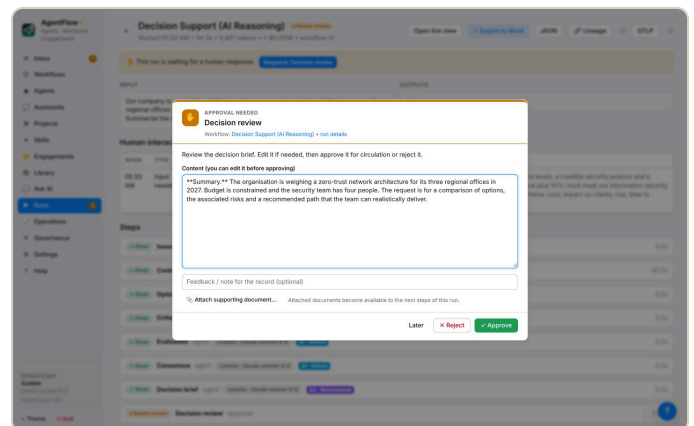
Every run start and finish, human decision, action, rollback, backup and stop is hash-chained; integrity is checked on the Governance page. Emergency stop cancels everything and blocks new runs.

Prompt-injection defences

Documents and tool results wrapped as untrusted data, an injection guard that flags or blocks, and a 12-attack injection suite for evaluation datasets.



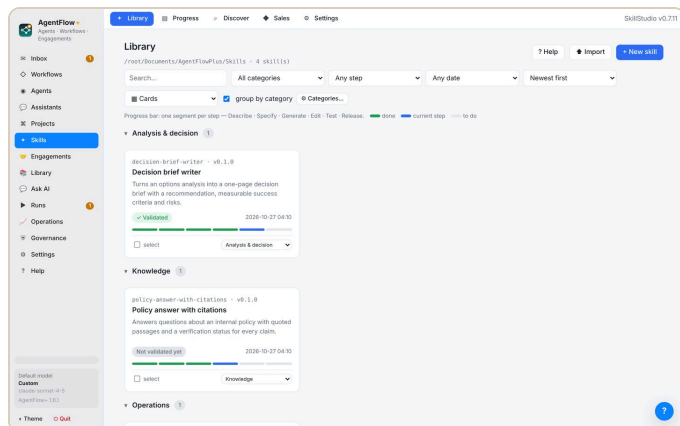
The Governance page: organisation-wide ceiling, the A0-A5 matrix and which agents sit where.



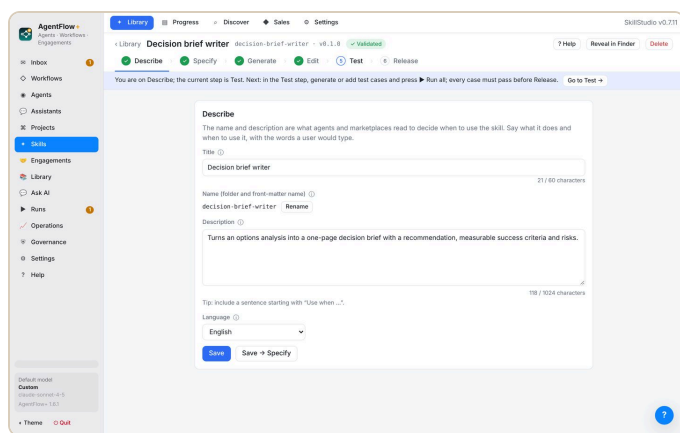
An approval pop-up with the actual brief, editable before approving.

Skill Studio, built in

Skill Studio is the author's standalone SkillStudio application, built into AgentFlow+ and sharing its providers, keys and data folder. Every skill is a plain folder; files are the truth. One linear flow is always visible: Describe → Specify → Generate → Edit → Test → Release.



The skills library with categories and the progress rail.



A skill at the Describe step; the rail shows where it stands.

Specify and generate

A guided intake or a form captures objective, requirements versus assumptions, inputs and outputs, acceptance criteria and target environment. Generation produces a validated proposal with traceability; nothing is written without your click, and every AI change can be undone.

Edit and validate

Section locks, Revise with diff, Review with anchored findings, a deterministic validator (references, JSON and schema checks, clarity lint, scripts analysis).

Test

Realistic AI-written cases — activation among decoys, execution with checks and an independent judge, boundaries, adversarial, script cases in a sandbox — that arrive unapproved. Run 3 or 5 trials per case for pass rate and flaky detection, and compare each case with a no-skill baseline.

Release and sell

Security scan (accept with reason), originality check, readiness checklist, immutable numbered releases with evidence and notes; kits for Claude.ai, Claude Code, Agensi.io, AI Skill Market and GitHub / SkillsMP; listing limits checked; sales ledger.

Skills at runtime

Agents, workflow steps and assistants pin a skill release (or the working copy for testing); runs record name@release and content hash; a pinned skill or release cannot be deleted or renamed.

Engagements and assurance

One workspace per customer or programme, with the data classification visible everywhere. Take a use case from scored intake to a signed-off handover with evidence.

Intake

Discovery notes; use cases scored 1–5 on value, feasibility, data readiness, low risk and urgency (weighted); measurable acceptance criteria you tick off; decision, assumption and baseline / benefits registers.

Evaluation

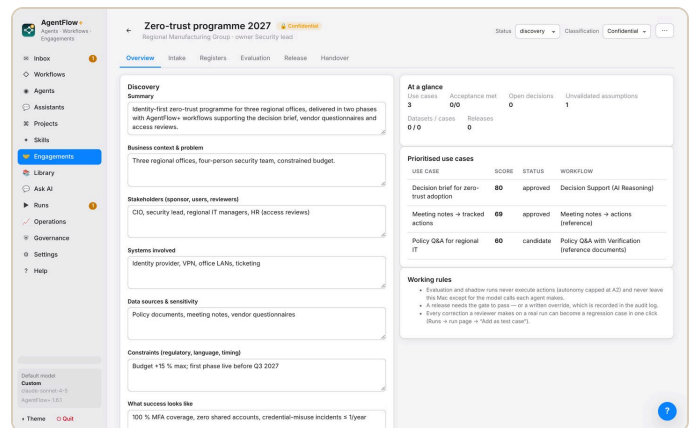
Datasets graded by rules, an LLM judge with your rubric and a reference answer, or a human reviewer. Runs use the real workflow with actions disabled; compare two runs to see regressions; pin an older version to re-test it; add the 12-attack injection suite in one click; turn a corrected run into a regression case.

Release

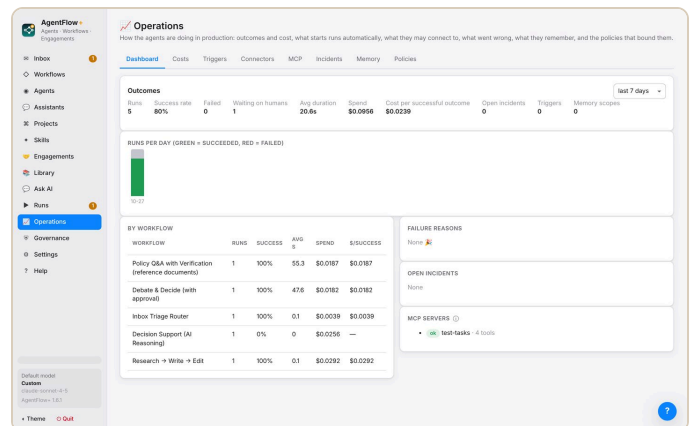
Gates (minimum pass rate, minimum cases, injection 100 %, human review done, cost and time per case) with written overrides that go to the audit log; shadow mode runs the released and candidate versions side by side.

Handover

A validated Word document in English or French — summary, scope, architecture, governance and risk, evaluation and releases, operations, registers, benefits versus baseline, traceability matrix, checklist, audit extract — plus a customer export .zip and solution templates for the next customer.



An engagement overview with scored use cases and working rules.



The Operations dashboard with cost per successful outcome.

Operations and integration

The Operations console is where you watch what the agents do in production and what they may connect to.

Dashboard

Runs, success rate, failures by reason, spend, cost per successful outcome, per-workflow table, open incidents, MCP health.

Triggers

Schedules (every N minutes, daily on chosen weekdays), webhooks (POST to this Mac) and folder watch (new files are OCR'd or extracted, attached and run).

Connectors and MCP

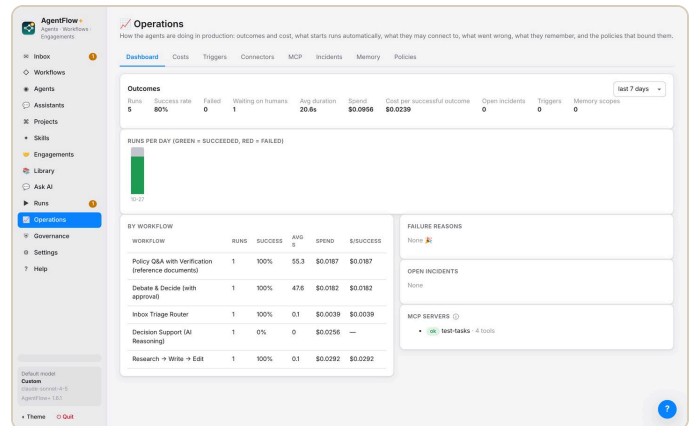
Named HTTP endpoints with credentials in the Keychain, auth styles bearer / header / basic / query and an allowed-domain list; stdio MCP servers (import a project's .mcp.json) with a read-only flag; tools appear in the agent tool picker.

Incidents, memory, policies

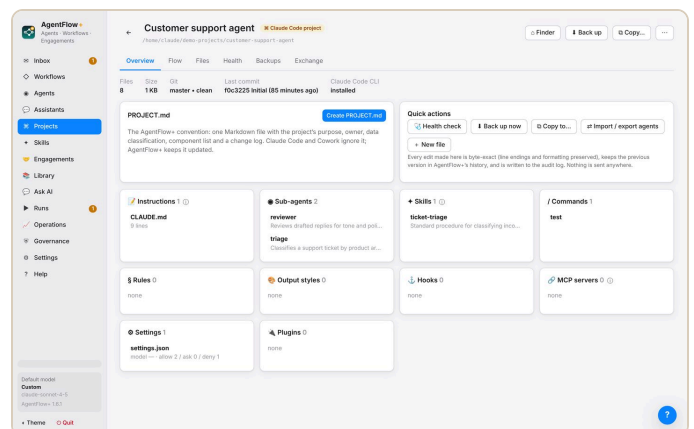
Incidents opened automatically on failed runs; agent memory with remember / recall tools per scope; approved providers per data classification; injection guard; output secret redaction; retention with purge.

Test adapter

External integration ships with a labelled TEST task adapter (local JSON, every answer marked) so the whole approval-and-write path can be proven before pointing at a real system.



The Operations dashboard with cost per successful outcome.



Project Hub

Local by design

AgentFlow+ is a native macOS application that serves its own interface on 127.0.0.1 with a per-launch security token. Only the window it opened can talk to it. There is no cloud component.

Data

One folder in Application Support: a SQLite database, documents and a workspace. Skills are plain folders in Documents. Scheduled backups to any folder, optionally AES-256 encrypted, with restore.

Secrets

API keys, tokens, connector credentials and the backup passphrase live in the macOS Keychain. Keys never appear in files or logs.

Privacy

No analytics, telemetry or phone-home. OpenTelemetry export is opt-in. Only the prompts you choose go to the provider you configured; with Ollama nothing leaves the Mac.

Models

Anthropic (streaming), any OpenAI-compatible endpoint (OpenAI, Groq, LM Studio, vLLM...) or Ollama locally. A mock provider lets you design and test offline at no cost. Provider per agent, routing by tier with failover, policies by data classification.

Project Hub

Claude Code and Cowork project folders opened in place: overview of CLAUDE.md, sub-agents, skills, commands, hooks and MCP servers; health check with one-click fixes; lossless front-matter editing; file history; agents and skills exchanged in both directions.

Engineering

Python standard library and vanilla JavaScript — no frameworks to patch. 211 automated tests on the current build plus end-to-end UI tests. Contextual help beside every control, in English and French, and a full User & Training Guide.

Licence and pricing

US\$10,000

per year

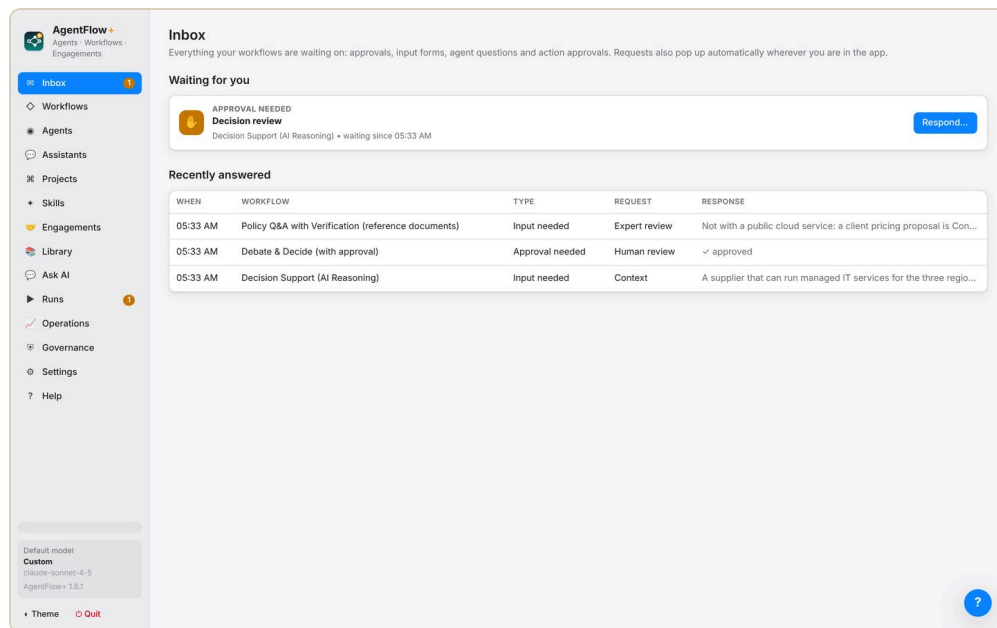
- Single-user licence for one Mac
- Every update released during the licence term
- E-mail support from the author
- No per-run, per-seat or per-token fees — ever
- Model usage billed by your own provider, or free with local Ollama models
- macOS 11 or later, Apple Silicon or Intel; delivered as a signed application

Built for

- Consultants and boutique firms delivering AI automation to clients
- Risk, compliance, audit and security professionals
- Solutions and forward-deployed engineers
- Skill authors selling on the marketplaces
- Educators and researchers

Next step

Ask for a 30-minute demo of the reference workflow — meeting notes → schema-validated extraction → bound approval → idempotent write — end to end, on one Mac.



Marc-André Léger, PhD, MBA · marcandre@leger.ca · www.leger.ca

© 2026 Marc-André Léger. AgentFlow+ is proprietary software; it may not be copied, modified or redistributed without written permission. Product names of third parties belong to their owners.