Architecture overview

What this platform is built to achieve

Autonomous development and operations platform

Context

The platform connects a live website, local engineering tools, an authorised Windows workstation and specialised AI assistants. Its objective is a controlled remotely operated workspace that prepares, executes, verifies and documents bounded technical tasks within an active work session.

Architectural intent

The goal is not to maximise the number of autonomous agents. It is to combine existing components with clear privilege boundaries. The owner defines the business objective and retains approval authority; ChatGPT structures work and coordinates available tools.

Practical value

Requirements, files and outcomes need not be continually copied between chat, Windows, PowerShell, WordPress and Claude Code. Logs, hashes, explicit system states and durable checkpoints enable verifiable handoffs.

Contribution

This is an independently developed project-based architecture and integration reference. Personal contributions include objectives, component selection, security and cost decisions, orchestration, acceptance criteria and independent QA. Many implementation and test components were produced with AI assistance.

> Documented project evidence, not a universal systems guarantee. Production publication requires separate final approval.

Autonomy guardian

Bounded tool routing, not an unrestricted agent

Autonomous development and operations platform

Natural-language requests

Autonomy Guardian A v1.3 can interpret a suitable scoped task and select an established route without making the owner specify script names or interface parameters.

Implemented scope

Its tested local execution bridge supports tightly allowed sandbox text-file creation/inspection and read-only checks of clearly identified windows in the authorised interactive Windows session. It cannot turn arbitrary requests into unrestricted shell writes.

Test evidence

A documented 10/10 safe integration test exercised a sandbox roundtrip with hash verification, read-only window inspection and handoff to existing workbench entry points. Autonomous production WordPress publishing was not part of this test.

Trade-off

The guardian remains a small on-demand routing helper. Avoiding a permanent background service or a new universal orchestrator reduces access exposure, integration overhead and unnecessary token use.

> Documented project evidence, not a universal systems guarantee. Production publication requires separate final approval.

Outcomes and transfer

Professional architecture relevance

Autonomous development and operations platform

Documented proof

The evidence portfolio includes 10/10 scoped guardian integration cases, 17/17 status checks, pre-login Windows remote availability, controlled WordPress previews and independent AI QA. These are multiple bounded proofs, not a single combined end-to-end acceptance of everything.

Daily capabilities

The system supports authorised local and remote tool use, preparation of website and course changes, code and test coordination, detection of unexpected changes, structured readback and resumption from documented checkpoints.

Limits

It does not publish arbitrarily without approval and is neither conscious nor completely self-running. Actual behaviour depends on available tools, interfaces and authorised rights.

Career significance

The case demonstrates solution design, systems integration, risk management, API-/MCP-based workflows, QA discipline, documentary evidence and cost-aware choices. AI Solutions Architect is a professional target, not a previously held position.

> Documented project evidence, not a universal systems guarantee. Production publication requires separate final approval.