
THE PLATINUM BUILD METHOD — ON DEMAND

Training for the people who never meant to become developers

Your analysts and engineers are already building internal systems with AI. Most of that software has no spec, no tests, and exactly one person who understands it. **This course teaches non-developers to build things that survive** — delivered by an AI tutor that sits beside their work, in the same discipline it teaches.

01 It teaches the thing the tool courses don't

Prompting courses teach the interface. This teaches what happens eighteen months after an enthusiastic build — the undocumented spreadsheet-turned-system, the owner who changed jobs — and the method that prevents it. Thirty years of enterprise systems discipline, aimed at the people who just acquired the power to build.

02 Learners produce real work, not coursework

Exercises are done in the learner's own tools, on their own systems. The tutor reads the actual spec and build diary from their project folder and marks the evidence. Training that ends with working artefacts and a method — not a folder of prompts.

03 A tutor, not a video library

A panel beside their Claude window that explains, answers in text or voice, sets the task, and marks what comes back. Nobody advances until the work passes. The gate is what makes completion mean something.

04 The course stays current

AI training rots in months. This course watches Claude's changes weekly and refreshes the perishable material — human-reviewed before anything ships. The course bought in January is still true in June.

05 It is its own proof

The course teaching your people to build properly with AI is itself an AI-built, AI-delivered system, run under the same rules it teaches. We are not aware of anyone else in the training market who can say that honestly.

THE SHAPE OF THE COURSE

LEARN

Beside the work

The tutor panel sits next to the learner's own Claude window. Short sections, plain language, spoken aloud if preferred — designed for real working days, in sittings of ten to fifteen minutes.

BUILD

In their own tools

Every exercise is a real piece of their real work: a spec, a build diary, a system that runs. Nothing happens in a sandbox, because their job doesn't.

PASS

On evidence

The tutor reads what they actually produced and marks it against the method. Pass, and the next section opens. Fall short, and it says exactly which criterion failed and invites another go.

A certificate that certifies something

Completion is not attendance. It means every gated exercise was passed and real artefacts were produced — and the certificate says so, with a verifiable ID that resolves to the completion record: what was passed, when, on which version of the course. The learner chooses what evidence to show alongside it.

CERTIFIED CLAIM

"Can build internal systems that survive their builder."

Exercises passed	evidence-marked
Verification	by certificate ID
Artefacts	learner-controlled

FOR TEAMS

- Seats bought by the manager; progress and produced work visible per learner.
- Evidence, not completion percentages — you see the specs and systems your people actually made.
- Pairs with our Shadow AI Audit: score the team before and after, and put a number on what changed.

WHAT IT IS NOT

- Not a prompting course. Prompts are the easy part.
- Not a video library with a quiz at the end.
- Not a sandbox. Your systems, your tools, your evidence.
- Not static. The course updates as the tools do.

Built around Claude. The method transfers; the examples are current.