Production readiness checklist for AI automations

Twenty-four yes-or-no questions that separate a demo from a system a client can run without you.

Iron Web Business · free · version 1.0

Why this exists

Every practitioner has had the same week. The demo worked on three examples. The client signed. In week two the workflow returned a different answer to the same input, a step failed silently, and the client's customer noticed before anyone else did.

None of that is a model problem. It is a missing constraint, a missing check, and no measurement. This checklist is the list of those missing things. Run it before you bill for an automation. Run it again before every change.

The rule: a "no" anywhere in *Check*, *Measure* or *Own* means the automation is not ready to bill for. A "no" elsewhere is a known risk you can name to the client.

Scope

- ☐ The steps that need a model are named. Every other step is ordinary logic.
- ☐ The proposal states an expected automation rate and what happens to the rest.
- ☐ The client has agreed what a failure costs and how many are acceptable per month.

Contain

- ☐ Each model call makes one decision.
- ☐ Each model call returns a constrained output (a schema, a list of allowed values, a number), not free text parsed afterwards.
- ☐ Each model call has fixed settings and a worked example. The prompt is versioned.
- ☐ The context sent is the smallest that decides the case.

Check

- ☐ Every model output passes a validation step before anything acts on it.
- ☐ Outputs that fail validation go to a human lane with the suggested answer attached.
- ☐ Actions involving money, customers or records have a second check or a human gate.
- ☐ The human lane has been used at least once and the client knows where it is.

Measure

- ☐ An evaluation set of real cases exists, at least thirty, with expected outputs.

- [ ] The pass rate on that set is known and written in the proposal.
- [ ] The set is rerun after any prompt or model change.

Observe

- [ ] Every run writes a record: input, output, lane taken, cost, duration.
- [ ] The client can open that record without you.
- [ ] A failure is visible within a day, not when a customer complains.

Cost

- [ ] The monthly cost at expected volume is in the proposal.
- [ ] A ceiling is enforced in code. The system stops or degrades rather than overspends.
- [ ] The smallest adequate model is used for each call.

Own

- [ ] The system runs on infrastructure the client controls.
- [ ] Nothing depends on your accounts or subscriptions.
- [ ] The code is in a language a hired developer can read. Dependencies are pinned.
- [ ] A README says what it does. A runbook says how to restart it and where the logs are.
- [ ] The client has run it once with you absent.

What a "no" usually means

Group	The usual cause	The fix, in one line
Scope	"Fully automated" was promised	Measure the rate; design the human lane; sell the system that includes it
Contain	One long prompt doing five things	One call per decision, with a schema
Check	The model's output goes straight to the action	A validation step and a queue for what fails it
Measure	"I tried it on a few and it worked"	Thirty real cases, a pass rate, rerun on every change
Observe	Nobody knows it failed until the customer says so	One row per run, in a file the client can open
Cost	The bill surprised the client	A monthly figure up front and a ceiling in code

Group	The usual cause	The fix, in one line
Own	It runs on your Zapier account	Their infrastructure, their accounts, a runbook

One number from a real run

We took one model step, a client brief in and a website sitemap out, and ran thirty briefs through it. With the output schema enforced but its limits never stated in the prompt, the check sent 43% of cases to a human on the small model and 77% on the large one. The large model wrote more, so it broke the unstated limit more often. Stating the limits in the prompt took the human lane to 3% and 0%. The model never changed. That is what *Contain* and *Measure* are for.

What this is the start of

This checklist is the table of contents of a method for taking an automation from a workflow tool to a small program the client owns. The first chapter, *From workflow to owned code*, walks one model step through Contain, Check, Measure and Observe inside the workflow you already have, and ends with your first owned script. Iron Web Business