Run this before anything an AI agent built goes live.

A code change, a listing update, a published page, a sent message, a deleted record. Two minutes start to finish if the spec was written up front. Every line below exists because skipping it cost us something real.

BEFORE THE AGENT STARTS

- [ ] **The spec is written down** — one or two sentences describing exactly what "done" looks like, specific enough that a stranger could check it without asking what you meant.
- [ ] **The agent has the real artifact, not your summary of it** — the actual file, page, or data, linked or pasted directly.
- [ ] **Standing constraints are restated in this brief**, even ones that feel too obvious to repeat. A rule isn't in force unless it's in front of the agent right now.
- [ ] **Off-limits resources are named** — files, tabs, accounts it may not touch.
- [ ] **No delegation** — it does the work itself, and reports partial progress rather than going silent if it runs short.
- [ ] **The model tier is chosen on purpose** — cheap for mechanical or checked-downstream work, capable for anything with real exposure if it's wrong.
- [ ] **A verifier is named — and it isn't the agent doing the work.**

WHY THIS HALF MATTERS

Most agent failures are briefing failures. Everything on this page happens before any work starts — and it's the half everyone skips.

Free to use, forward, or paste into your own project

BEFORE YOU CALL IT SHIPPED

- [ ] **The agent that made the change did not verify the change.** A different context checks it — a second agent, a second pass, a second person.
- [ ] **The check reloaded the artifact fresh — no cache.** A plain reload right after a save can silently serve the pre-change version and read as proof nothing landed.
- [ ] **A success message was not treated as success.** A toast or a "Saved" label proves a request was sent — not what state the object is actually in. Reload and read it again.
- [ ] **A dialog's "Done" was not treated as saved.** If a page-level save exists separately from a modal's close button, that save was clicked and confirmed — not assumed.
- [ ] **The check ran from the view the world actually has** — logged out, no personalization, no cached session — not the flattering logged-in view.
- [ ] **If this was a removal:** the check confirmed specific evidence it ran — a log line, a before/after comparison, an independent count. "I don't see it" looks identical whether the removal worked or silently didn't.
- [ ] **If a number was involved:** it was checked with independent arithmetic or a separate method — never by re-running the same code that produced it.
- [ ] **If this had a fallback branch:** that branch was actually forced and tested — not just the primary path that normally runs.
- [ ] **A helper agent's report was treated as one hop from the source.** Anything load-bearing was checked against the actual file or output — the way you'd treat a coworker's summary, not your own eyes.
- [ ] **"Checked, empty" and "could not check" were not collapsed into one report.** If something blocked the check, that is the finding — not a false all-clear.

Free to use, forward, or paste into your own project

Put it to work in the next five minutes.

Copy the prompt below into any AI assistant after your agent reports "done." It is the smallest useful version of the discipline on the previous two pages.

A STARTER VERSION OF THE VERIFIER PROMPT

```
You are checking finished work. You did not do the work and must not assume it was done
correctly.

Here is what the finished result is supposed to look like: [PASTE YOUR SPEC]

Go look at the actual result — the live page, the real file, the actual data — not any report
about it. Compare what you find against the spec, point by point. Report every difference,
including "could not check" items as their own category. Do not soften findings.
```

THE ONE QUESTION UNDERNEATH ALL OF THEM

If I'm wrong about this being done, what would that look like — and did I just check for that, specifically? Or did I check for something else and call it close enough?

Free to use, forward, or paste into your own project