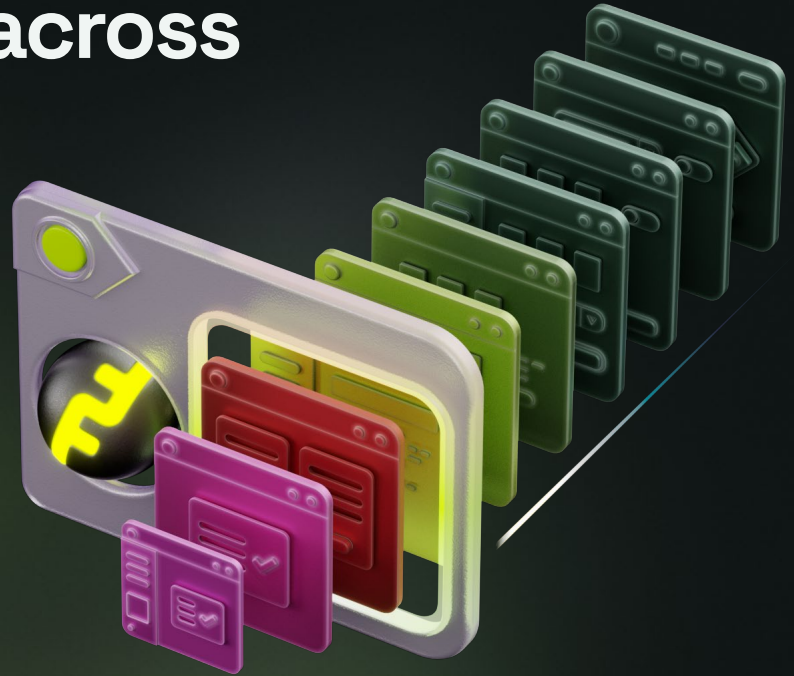


5 standards for validating agent-built code across the AI-DLC

Agents write more of your code every sprint. In the AI-driven development lifecycle (AI-DLC), every change needs proof that is fast, approved, and auditable. Check the standards you meet today.



- ☐ **01** Every change is validated from its intent, and your team approves it.
- ☐ **02** Agent-written code is validated end to end before it ships.
- ☐ **03** The full regression runs in parallel on every release.
- ☐ **04** A red run tells you right away: real bug or known change.
- ☐ **05** Coverage keeps pace as agents change the app.

Left a box unchecked?

Each gap is where code outpaces proof.
The next section shows how teams close it.



Why each standard is hard, and how Studio meets it

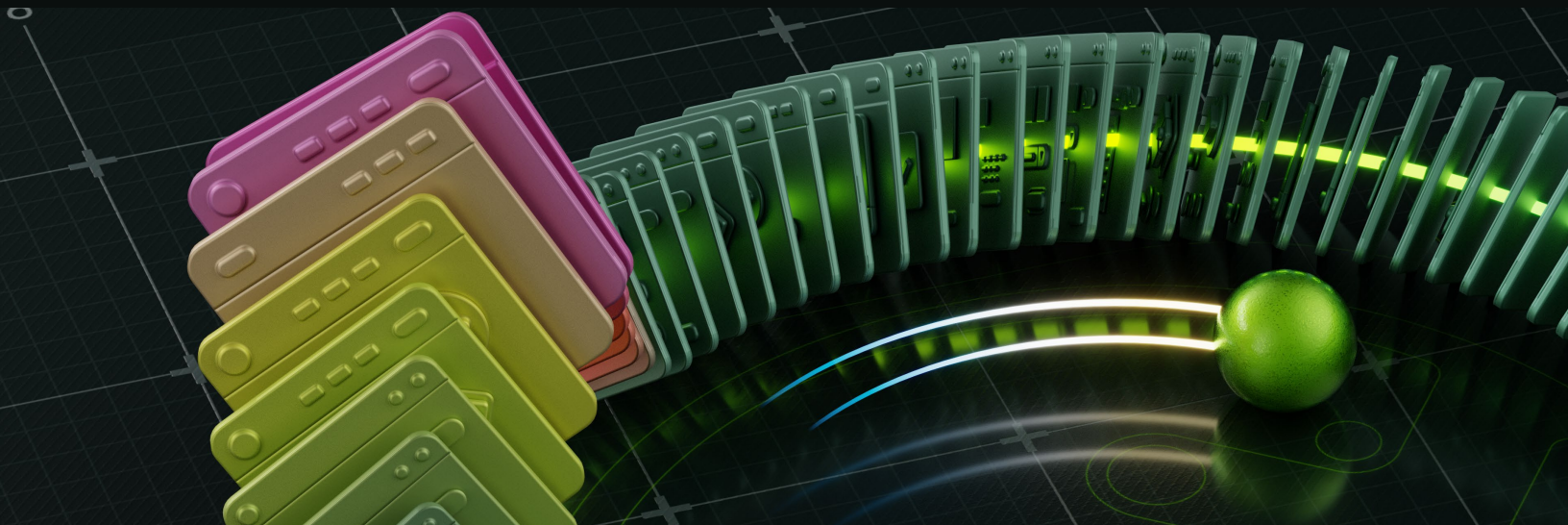


STANDARD

WHY IT'S HARD TODAY

WITH FUNCTIONIZE STUDIO

01 Every change is validated from its intent, and your team approves it	Agents write a change in minutes. The end-to-end check still takes a person hours, so unit tests become the only proof.	Describe the journey, or let your coding agent pass the intent through MCP. Studio builds the validation, and a reviewer approves it.
02 Agent-written code is validated end to end before it ships	Code volume went up. Proof still runs at human speed, so unvalidated change piles up.	Approved tests run in your CI/CD pipeline against the live app, not just the diff.
03 The full regression runs in parallel on every release	Releases come faster and suites keep growing. When the suite is slow, the release date wins.	Runs start in 3 to 5 seconds and run in parallel across browsers, without the flakiness. The full regression fits the release window.
04 A red run tells you right away: real bug or known change	Selector-based scripts break when the UI shifts, so triage eats the time meant for the fix.	Every step runs on a System One model, with 99.999% element accuracy, about 1 miss in 100,000. Agents explain a failure when you ask.
05 Coverage keeps pace as agents change the app	Script upkeep grows with the suite, so teams cap coverage at what they can keep green.	Tests keep finding the right element after the app changes. When a flow changes, agents update the test when you ask.



Agents write the code. Functionize Studio proves it works.



Studio is the validation layer for the AI-DLC. Agents build, run, maintain, and diagnose validation. Your team approves every one.

WHERE STUDIO WORKS IN THE AI-DLC



DEFINE

Describe a journey, or pass the intent from your coding agent. Studio validates.



VALIDATE

Approved tests run in your CI/CD pipeline on a System One model.



MAINTAIN & DIAGNOSE

Tests keep finding the right element as the UI changes. Agents explain a failure.

BUILT FOR REGULATED TEAMS

Control

Nothing becomes a test until your team approves it. AI never changes your test unless you ask.

Auditability

Every step of every run saves screenshots, so anyone can see what happened.

Scale

Up to 3,000 steps per journey, parallel runs, and isolated execution.

SOC 2 Type II

SSO and RBAC

Isolated execution

Data residency

Encrypted tunnels

Cloud or private

Web, cross-browser



Try Studio on your critical user journey



[FUNCTIONIZE.COM](https://functionize.com)

Studio

