Engineer Software

Wrote tests for the checkers themselves

Wrote tests for the checkers themselves after establishing that a checker fed only clean input will one day report clean because it read nothing — planting a misspelling to confirm the spell-check finds it, and taking an id range from the database rather than from a number in the test.

August 2026

tests for the checkers, so a clean report means something

Situation. The project runs a set of custom checkers over its own content — a spell check, an identifier-range check, a voice check, a figure-consistency check. Each of them had run green for months. A checker that has only ever seen clean input and only ever reported clean is indistinguishable from a checker that reads nothing at all, and there was no test in the suite that could tell the two apart.

Task. The checkers had to be made to prove they can fail, and the fixtures they check against had to stop being hand-maintained copies of the thing they describe.

Action. The pattern applied throughout is to plant the defect the checker exists to find and require the checker to find it. The spell check is fed a deliberately misspelled word and the test fails if the run comes back clean. The voice check is fed prose in the first person and must reject it. The buzzword check is fed a banned word. Each of these is a small test and each one closed a real blind spot, because two of the checkers turned out to be reading a narrower set of files than their documentation claimed and had been silently skipping content. The second change is about where a test gets its expectations: the identifier-range check previously compared against a number written in the test file, which meant every content addition required editing a test, and an editor who updated the number without looking had disabled the check. It now derives the range from the database, so the check describes the data rather than a stale memory of it.

Result. Every checker in the project now has a test that proves it fails on bad input, and the range assertions read from the source of truth. The uncomfortable part is what this exposed: a green check had been meaningless in at least two places for an unknown length of time, and there is no way to find out retroactively what passed through.

Bring this experience to your project

Use “Wrote tests for the checkers themselves” as a starting point for a similar challenge in your organisation. The services behind this work are available for your project. An agent wired to the systems you use, tested on cases your own experts write, overseen by your people and never tied to one model.

Explore our services Get in touch →

Look behind the result

Read the situation, decisions and result in this example. We can discuss which parts transfer to your constraints and which need another approach. Wrote tests for the checkers themselves — tests for the checkers, so a clean report means something

Explore our services Get in touch →

Is your quality gate checking anything at all?

Talk through your project with the engineer responsible for delivery. We agree the scope, price basis and handover in writing before work begins.

Role context: Test and Quality Engineer. It draws on Databases, Data Governance, Automation & CI/CD, Testing & QA and Python. And while you are here, we are rather good at Data Governance & Quality, DevOps & CI/CD Automation, Custom AI Agent Development and Technical Leadership & Consulting.

Get in touch Explore our services

One piece of **a longer record** — and it is all on this site.

Every entry is written the same way: the situation, the task, what we did and what changed.

See the whole record

Define the improvement you need

Describe the workflow, the existing interfaces and the users who need the change. We can define a testable delivery and a handover that lets your team maintain it.

See what it costs Get in touch →

Is your quality gate checking anything at all?

Ask about the decisions behind “Wrote tests for the checkers themselves”. You will speak with the people who can explain the work and assess your own project.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://software.engineer.company/portfolio/wrote-tests-for-the-checkers-themselves/