The 6-point test-trust checklist

How to know your green Playwright test is real, and not a false green hiding a broken checkout.



WEAK · CHECKS WHAT THE APP SAYS

```
await expect(page.getByText('Order
confirmed'))
  .toBeVisible();
```

Passed. The order was never saved.

STRONG · CHECKS WHAT THE APP DID

```
expect(orders, 'there must be exactly 1 saved
order')
  .toHaveLength(1);
```

Failed. Real defect caught.

1 Define a trusted oracle

Assert the outcome where it lives (database, API, downstream system), not the message on screen.

- [ ] The test verifies the result, not just a toast, redirect or status text.

2 Test known-bad behavior

Break the app on purpose. If the test stays green, it isn't protecting you.

- [ ] We seeded a bug and confirmed the test fails.

3 Inspect repair diffs

A fix, by a person or an AI agent, may change how a test reaches the page, never what it checks.

- [ ] Every repair keeps the original assertions. Loosened or deleted ones get a human decision.

4 One test, one clear reason to fail

A failure should tell you what broke without opening the code.

- [ ] Each assertion has a specific message, and each test covers one outcome.

5 Control your test data

Shared or leftover data can make a broken flow look healthy, or a healthy one look broken.

- [ ] Each run creates and cleans up its own data.

6 Classify every failure before you retry

Red is a product defect, a test problem or an environment issue. Blind retries hide all three.

- [ ] We check the trace and label the failure before rerunning.

6/6	Your green means something.
4–5	One gap could hide a real defect.
≤3	Start with points 1 and 2.

Amikoo is an AI QA agent that writes Playwright tests in your repo for your team to review, tells you if a failure is the product, the test or the environment, and proposes repairs your team approves before merge.

amikoo.ai