TESTMU AI · CUSTOMER CASE STUDY

How a Retail Bank Replaced Self-Reported Coverage With Machine-Read Evidence

- **4,200 tests scored on what they actually verify**
- **41% of the suite found to be asserting nothing meaningful**
- **Coverage is now machine read rather than self-reported**

About the Bank

The bank is one of the country's large, full-service retail banking groups, serving millions of customers across deposits, cards, lending, and wealth products through its branch network and digital channels. Its digital banking division, a 240 person engineering organization supported by 38 SDETs, runs the mobile and web journeys customers use to check balances, move money, and manage accounts.

Because the division operates in a regulated industry, it answers to internal audit, external regulators, and customers. Both periodically require demonstrable proof, rather than assurance, that the systems behind those journeys are properly tested. That requirement prompted a review of what the automated test suite was establishing.

The Challenge

The division ran 4,200 automated UI and API tests each night against its retail banking platform. Dashboards were green, and the coverage figure reported in the quarterly business review was 87%.

The issue was how they produced that figure. The team set the coverage target internally, and maintained the mapping of tests to requirements by hand. The number was self-reported at every stage: entered into a slide rather than read from the system.

A review commissioned by a new Head of Quality Engineering examined the suite itself. It found hundreds of tests whose only assertion was `assertTrue(result != null)`, soft asserts that were collected but never verified, and steps that scrolled a page and were counted as checks.

The division could not identify which of its 4,200 tests verified anything meaningful, and the coverage report could not answer the question either, because it measured execution rather than proof. A test that ran to completion and a test that verified an acceptance criterion appeared identical on every dashboard the bank had. In a regulated business, where internal audit periodically asks how the division knows its controls are tested, each of those results was a claim that had to be accepted on trust.

Why TestMu AI

The decision point was an internal audit question the team could not answer accurately: how does the division know these tests prove the requirements they are mapped to? The answer was that the mapping asserted it rather than demonstrated it.

The team evaluated Kane CLI because it reverses that relationship. Coverage is derived from what the system executed and verified rather than entered manually. Each result carries a score, each score is supported by an artifact, and each artifact is linked to the requirement it verifies.

The Solution

The team applied Kane CLI to the existing suite rather than rewriting tests at the outset.

The first full scan scored every test on what it verified, separating tests that confirmed a real acceptance criterion from tests that only ran to completion. Null checks, unverified soft asserts, and scroll and count steps were identified as execution without verification.

The division then rebuilt its coverage picture on Kane's chain of evidence, which links source requirement to use case, acceptance criterion, scenario, test, and evidence artifact. The coverage figure became a reading produced by the system rather than a value maintained by a person. Any stakeholder can move from a passing result to the score behind it, the artifact behind the score, and the requirement behind the artifact.

The Results

The first machine read figure was substantially lower than the 87% the division had been reporting. 41% of the suite was found to prove little or nothing.

It was also the first coverage figure the division did not have to defend, because it could be inspected directly. Over the following quarter, the team retired or repaired the weakest tests and rebuilt coverage against the requirements that mattered, with supporting evidence attached.

> *"We found that a substantial share of our suite was not verifying anything. That was difficult to absorb. But when audit now asks how we know, we can show the evidence rather than a spreadsheet."*
>
> ***Head of Quality Engineering, retail bank***
>
> *"A test that runs and a test that proves something used to look the same to us. They no longer do."*
>
> ***Senior SDET, digital banking platform team***

Ready to improve your testing? Schedule a demo to see how TestMu AI can help your team achieve similar results.

Industry: BFSI, Retail Banking
Employee size: 12k+
Product: Kane CLI