



HOW EQUIDEFI INCREASED TEST COVERAGE TO 90%+ WHILE CUTTING REGRESSION TESTING TIME BY NEARLY 40%

Customer Overview

EquiDeFi is a private investment platform connecting issuers offering investment opportunities with investors looking to purchase shares.

With a lean team supporting a growing financial technology platform, EquiDeFi needed a way to maintain high product quality without allowing regression testing to slow down releases.

Before testRigor, testing was almost entirely manual. Critical-path testing relied largely on a document-based checklist, with QA manually working through important investment and issuer flows.

Today, EquiDeFi combines testRigor automation with targeted manual testing to achieve more than 90% test coverage while completing regression testing significantly faster, with zero critical bugs in production.

90%+

Test coverage gives broader release confidence

40% Faster

Faster release validation

51

Critical tests automated

Zero

Major production escapes

The Problem

Before adopting testRigor, EquiDeFi's critical-path testing was entirely manual.

The team worked from a document that functioned primarily as a checklist. Testers followed the list while determining some of the actual test paths as they went. There was limited formal test infrastructure and no automated test coverage.

The checklist represented approximately 25–30 critical paths covering areas such as investor flows and issuer actions.

This approach created several challenges.

Time-Consuming Regression Testing

Critical-path testing required approximately 8–10 hours per person, averaging around 9 hours for each QA involved in a release.

For a small team, spending an entire working day validating a release created a significant constraint.

Limited Test Coverage

The team estimated that its previous approach provided roughly 60% coverage. Increasing coverage through manual testing alone would have required substantially more QA effort and could have made release testing even slower.

Limited Visibility Into Test Execution

Manual testing also lacked a strong audit trail. When QA reported that a workflow had passed or failed, other team members largely had to rely on that report. There was no automatically generated history containing detailed execution steps, screenshots, and videos that the wider team could review.

As QA Analyst AJ Vincent described it, testRigor helped replace the "take my word for it" approach to QA with evidence that everyone could see.

What EquiDeFi Wanted

Faster Testing

The primary challenge was time. EquiDeFi wanted to test releases faster without sacrificing coverage.

Greater Coverage

The team needed to validate more critical user journeys while keeping the release process manageable for a small QA team.

Reliable Automation

Enable the existing QA team to automate scenarios without requiring extensive Automation needed to be dependable enough that QA could confidently rely on it during critical release testing.

Better Test Evidence

The team also needed clearer, auditable evidence of what had actually happened during testing.

Key Objectives

- Reduce regression testing time
- Increase critical-path test coverage
- Build reliable, low-maintenance automation
- Reduce production defect escapes
- Keep QA focused on product quality

"There's something about testRigor's exactness that I don't see anywhere else."

AJ Vincent
QA Analyst, EquiDeFi

Why EquiDeFi Chose testRigor

EquiDeFi evaluated multiple options, including Datadog, but testRigor stood out for its **pricing flexibility, ease of use, and responsive support**. The ability to start small and scale as needed made it particularly attractive for a lean team.

Aaron joined EquiDeFi as a manual QA professional with no traditional automation engineering background, but quickly saw an opportunity to automate time-consuming critical paths with testRigor. He began building automated tests in February 2025, with his first critical end-to-end test running successfully by March 2025.

The Solution

EquiDeFi built a complementary testing model where testRigor handles an increasing share of regression testing while targeted scenarios remain manual, particularly financial workflows requiring human review. Today, the team has **51 automated tests**, with around **44 automated staging tests** running alongside **40 manual tests** during a typical release cycle.

This enables EquiDeFi to cover roughly **84 tests per release**, including complex end-to-end scenarios, while executing across Chrome, Firefox, Edge, Android, and iPhone. testRigor also provides **screenshots, videos, and shareable execution URLs**, giving the team an auditable history of test results and improving communication around release quality.

"You can't argue with a testRigor execution that's right in front of you."

AJ Vincent
QA ANALYST, EQUIDEFI

Solution Highlights

- Grew from **0 automated tests to 51 testRigor tests**
- Executes approximately **44 automated tests during a typical critical release cycle**
- Runs automation alongside approximately **40 targeted manual tests**
- Supports roughly **84 combined tests during a release cycle**
- Increased test coverage from around **60% to more than 90%**
- Reduced regression testing from approximately **9 hours to 5–6 hours per QA with 90%+ coverage**
- Added coverage across browsers and mobile environments

Before testRigor

- Testing was 100% manual, with no automated tests.
- Test coverage was estimated at around 60%.
- Regression testing took around 9 hours per person.

During and After testRigor

- EquiDeFi now has 51 automated tests in testRigor.
- Test coverage has increased to 90%+.
- Regression testing now takes approximately 5–6 hours per person with increased coverage.

The Result

EquiDeFi can now test more of the application in less time, with test coverage increasing from around **60% to 90%+**. Despite this significantly broader coverage, regression testing time has dropped from approximately **9 hours per person to 5–6 hours per QA**.

Production quality has improved alongside testing efficiency, with major production defects previously occurring roughly **once every 6–8 weeks**. The team has now gone an extended period without a major critical production defect, with AJ viewing testRigor's reliable automation as an important contributor to this stability.

Summary

Objective 🎯	Result 📊
Increase Test Coverage	~60% → 90%+
Reduce Regression Time	~9 hours per QA → 5–6 hours per QA
Introduce Automation	0 → 51 automated tests
Expand Release Testing	~44 automated + ~40 targeted manual tests
Improve Test Visibility	Auditable executions with screenshots, videos, and execution URLs
Reduce Production Escapes	From roughly one critical defect every 6–8 weeks to no major critical defect for an extended period
Keep QA Focused on Quality	Less dependence on code-heavy automation engineering

testRigor can also help you

“It creates this space where I can actually do my job maintaining quality, not forever maintaining these tests.”

AJ Vincent
QA ANALYST, EQUIDEFI

testRigor enables QA teams to build and maintain automated tests in plain English, making automation accessible without deep programming expertise across web, mobile, desktop, APIs, databases, AI features, and mainframe applications. By reducing the engineering and maintenance effort traditionally associated with test automation, teams can expand coverage, accelerate releases, and focus more on product quality.