

expoqa[®]26

MADRID 26th, 27th & 28th May

expoqa.eu

From Flaky Pipelines to Strategic Business Driver - A 2+ Year Journey of Continuous Marginal Gains

...

By
Sharath Byregowda

Agenda

- 01 The Catalyst: Diagnosing Flakiness and Fragmentation
- 02 The Foundation: Rebuilding Trust with Playwright
- 03 The Performance Gap: Breaking the "50% Reliability" Ceiling
- 04 The Scale: Democratising Quality Across the Org
- 05 The Result: Marginal Gains as a Strategic Profit Driver

From Catalyst to Foundation



Diagnosing flakiness and fragmentation to rebuilding trust

The Triple Threat - Diagnosing Systemic Friction

01

Flakiness & Fragmentation

E2E tests for the legacy UI were notoriously unreliable. A fractured landscape of multiple frameworks led to a total breakdown in reliability and a complete lack of engineering ownership.

02

Repository Refugee Crisis

74 E2E tests were effectively exiled from the Frontend monorepo. The existing infrastructure could not support them against a correctly aligned backend instance, creating a massive technical and operational bottleneck.

03

Testing by Layer (P0)

We moved away from treating E2E as a catch-all safety net. By prioritising P0 integration flows and shifting lower-level logic to unit, component, and API tests, we redefined E2E as a lean, high-value asset.

The Inflection Point - Cypress vs Playwright

The Status Quo (Cypress)

Legacy UI/Component testing was constrained by a single-threaded browser architecture.

Severe limitations regarding third-party blocking and complex promise-chaining created "maintenance debt."

The New Core (Playwright)

Clear winner in PoC for speed and features:

`check_circle` Native Parallelisation: Dramatic reduction in execution time.

`check_circle` Modern Capabilities: Out-of-the-box cross-browser support and superior debugging.

`check_circle` Stability: High-fidelity reporting and native CI/CD integration.

Key Learning: Sometimes, achieving a unified, long-term solution requires the short-term risk of introducing a new framework to break a technical stalemate.

Velocity & Foundation - The First 60 Days

speed

The Sprint

Built the core framework and pipeline in under two months with a single dedicated engineer.

trending_up

Immediate ROI

The new P0 suite found critical bugs in its first week—defects that previously would have reached production.

architecture

The Homeless Pipeline

Because neither the FE monorepo nor the existing architecture could accommodate the E2E suite, we had to architect a bespoke, independent pipeline from scratch.

Scaling the Standard - Enablement & Ownership

groups

Enablement-First Culture

We didn't just build a tool. We ran intensive enablement sessions across the entire engineering organisation to ensure adoption wasn't forced, but felt.

account_tree

Revamped Code Ownership

We moved to a decentralised model. Teams now use dedicated fixtures and shared reusable functions, balancing custom needs with architectural consistency.

security

Total Test Isolation

We implemented a Burn-Down philosophy. The pipeline creates everything the test requires (unique users, clean data) and destroys it immediately after. No shared state, no side effects.

The Ripple Effect (Internal Social Proof)

“Your presentation on the last Frontend COP meeting inspired us to do similar thing for unit and component tests”

— AK (Senior Software Engineer)

The Takeaway:

True marginal gains are infectious. When you fix the "broken" top layer, it sets a new standard for quality that developers naturally want to replicate across the entire stack.

Closing the Performance Gap



Pushing past the reliability from 50% to 80%

The 50% Mystery: Pinpointing the Real Failure

Playwright tests were reliable, but the dashboard build pass scores stubbornly hovered around 40-50%

Owner	All ▾	Build Number	All ▾	Frontend Version	All ▾	Dgc Aggreagtor Version	All ▾									
Latest Build Status		Build Passing Rate		Last 3 Days Results Last 3 days												
Fail		58.0%		4228	4229	4230	4231	4232	4233	4234	4235	4236	4237	4238	4239	4240
				4241	4242	4243	4244	4245	4246	4247	4248	4249	4250	4251	4252	4253
				4254	4255	4256	4257	4258	4259	4261	4262	4263	4264	4265	4266	4267

Pinpointing the Real Problem

psychology

Root Cause

An earlier, necessary decision separated E2E tests from the frontend repository.

sync_problem

The Gap

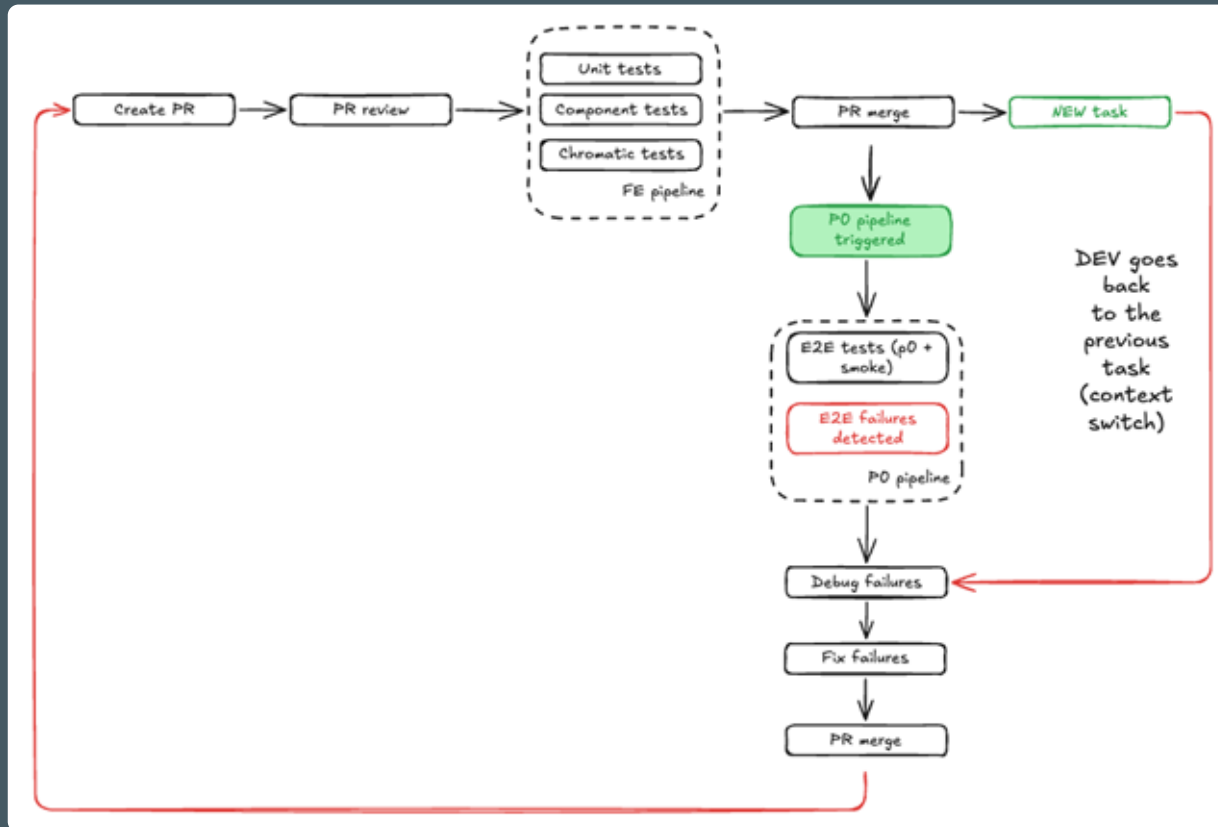
Frontend changes were merged without corresponding E2E test updates, creating constant instability.

report_problem

The Symptom

Frequent failures, broken builds, and pipeline blockages. We were constantly fighting fires.

Measuring the Impact: Reclaiming Developer Focus



The Context Switch Tax

When tests fail late in the cycle, developers are forced to abandon their current task to address historical regressions.

sync_problem Developer Flow Interrupted: "DEV goes back to the previous task (context switch)"

error_outline Delayed Feedback: Issues detected after PR merging create a backlog of "firefighting" tasks.

trending_up The Goal: Reclaiming focus by moving E2E tests closer to the development phase.

The Strategy: Co-locating Test and Source Code

track_changes

Goal

Ensure application code changes and corresponding test updates happen in the same pull request.

published_with_changes

Transformation

Aug 2024 - Jan 2025: Jenkins → GitHub Actions. Moved from separate test repo to Frontend monorepo.

trending_up

Results

By January 2025, build pass rates soared from 50% to over 85%.

Key Learning: Integrating critical tests earlier in the workflow minimises disruptions and enhances efficiency.

Measuring the Impact

analytics

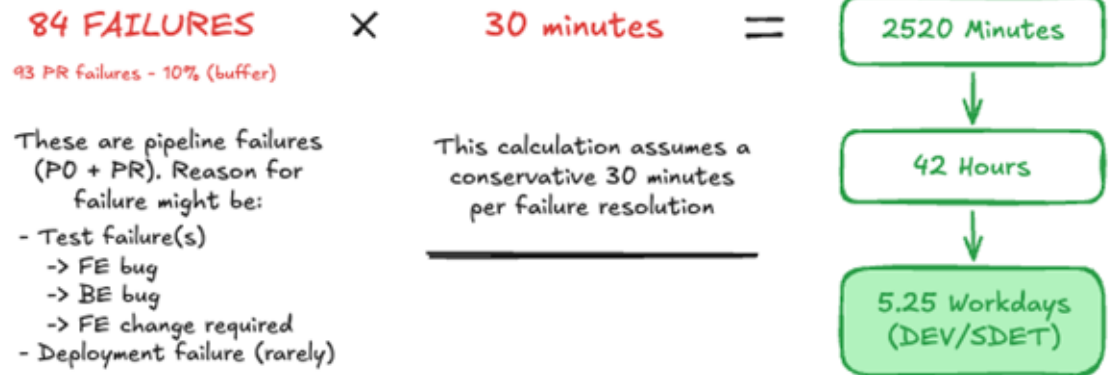
The Hard Data

84 post-merge pipeline regressions were successfully prevented (based on March 15th metrics).

timer

Productivity Dividend

Reclaimed 42 hours equal to 5.25 business days in a single day.



This calculation assumes a conservative 30 minutes per resolution.

This is well below the hours these fixes typically drain, making this a highly modest estimate of time saved.

Scaling Impact



Making Quality a Shared Responsibility and the Birth of the
TAE Team

TAE Team Charter: Beyond the Baseline Pipeline

Following the proven impact of our initial E2E initiatives, our temporary task force officially evolved into a permanent team: the Test Automation Enablement (TAE) team.

bolt Core Mission

Champion "Shift-Left" Quality: Push validation earlier into the development lifecycle to catch issues before merge.

Pioneer AI-Driven Testing: Investigate and integrate AI in automation/testing capabilities.

Build High-Value Insights: Execute rapid POCs and deliver metrics dashboards that drive actual engineering decisions.

Optimise Developer Experience: Remove friction from local and pipeline testing workflows.

Scale Technical Capability: Empower both developers and testers to test faster, smarter, and with greater confidence.

vpn_key Technical Ownership

Playwright Infrastructure: Serving as the architectural backbone for all enterprise E2E testing.

GitHub Deployment & Test Actions: Orchestrating PR validation gates, nightly regression runs, P0 checks, and customised team pipeline automation.

Artillery Load Framework: Supporting and maintaining our core performance and scalability testing utilities.

Automation CoP & Strategic Consulting: Cultivating the engineering community through continuous mentorship, best practices, and expert guidance.

In the first 3 months

Reducing Error Log Noise

Log volume dropped ~92% (from ~3000 to 200 entries per run)

Enhanced Playwright Docs

Month 1

Failure Insights for E2E

First-of-its-kind dashboard to pinpoint exact failing steps.

New Dashboards Built

Month 2

Launched TAE Bot

Slack app for test ideas, envs, and pipeline reports.

Workato Recipes Published

Month 3

"TAE's amazing step-by-step report on flaky tests is a gold mine of actionable input to improve testing. It even made us realise that we own THE MOST flaky test step in the entire repository."

— SM, Engineering Manager

By the end of the year

Smarter e2e execution

Introduced analyze-changes stage combining Playwright & NX affected.

New API Strategy

Introduced reqover-proxy + OAS coverage reports.

Launched CoP

Hosted Brussels Testing Tech meetup.

TAE Bot Upgrades

OTEL added; actions like generate-sprint-report, get-ticket-builds.

Cross-browser Tests

Nightly tests reflect real user envs (macOS + Win).

Supporting Teams

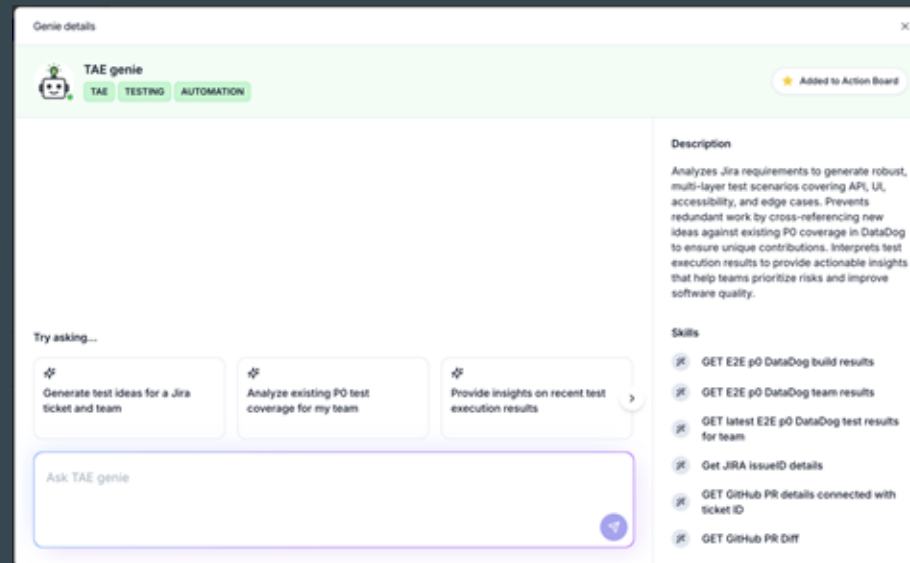
Launch of TAE-Genie

Execution

API & CoP

Bot & OTEL

Browser & Teams



Performance Metrics & Insights

check_circle

82.0% Pass Rate

Across 2204 total builds recorded

bolt

10.5 min Avg Execution

Average execution time per build



TAE Bot paying dividends

Effort Reclaimed

40+ Hours

In the first 8 weeks alone

Invocations

391

Total TAE Bot runs

Efficiency Gain:

Equivalent to onboarding an additional engineer for a full week, at zero cost.

< 🕒 2026-01-01 00:00:00 to 2026-02-28 23:59:59 > 🔍 🔄 Refresh ▾

TAE Bot Usage Stats

Action	Success	Error	User Error	Total	Working Days Saved
deploy-env	175	0	0	175	1.1 days
generate-test-ideas	53	4	2	59	2.2 days
get-aggregator-changes	50	2	1	53	1.0 days
generate-sprint-report	38	3	0	41	2.4 days
run-e2e-tests	26	0	0	26	0.2 day
report-failures	23	0	0	23	0.5 day
help	21	0	0	21	0.0 day
monitor-e2e-workflow	0	5	0	5	NaN
Total	391	15	3	409	1.1 weeks

Hello 2026

Exploratory Testing Agent

AI-powered agent that knows your product, its domain model, its APIs, its browser interaction quirks, its open findings and uses that knowledge to test like an experienced tester rather than executing steps mechanically

Exploratory Agent

E2E Tests Creation with Skills

- e2e-tests-ownership
- e2e-tests-schema-scaffolding
- e2e-tests-data
- e2e-tests-scaffolding
- e2e-tests-users
- e2e-tests-creation
- e2e-tests-orchestrator

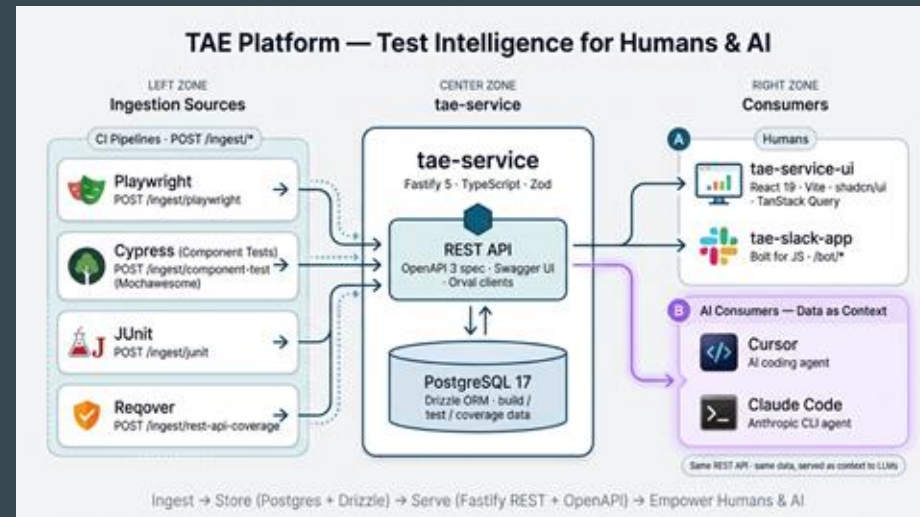
E2E Skills

Flaky Test Analyser Skill

e2e-tests-flaky-test-analyzer

TAE Service and many more

Analytics & More



The Philosophy



Compounding marginal gains into a strategic profit driver

The Power of Marginal Gains

We believe that small, continuous technical optimisations compound into massive strategic value for the engineering org. By moving fast and ruthlessly eliminating minor inefficiencies, we unlocked exponential performance wins:

settings_suggest

Streamlining Docker

Smoother container environments for developer setups.

fact_check

Automated P0 Testing

Integrated directly into daily PR workflows.

volume_off

92% Less Log Noise

Slashing non-critical backend logs to clear the noise.

auto_fix_high

Evolution to Skills

TAE Bot -> TAE Genie -> Skills for Agents

layers

Test Consolidation

Global setup executes once rather than per shard.

psychology

Exploratory Agent

Retains context and memory for testing.

Thank You



expoqa[®]26

MADRID 26th, 27th & 28th May

Thank you for attending

expoqa.eu