

expo IQA 25

MADRID
May 20th,
21st & 22nd
2025



expoqqa.eu

Automation First Approach

From Monolith to Modularity:
Rethinking QA Automation

Presented by: Andrea Baccharin & Laura Rodríguez

How we started Automating Tests Cases 60 Days Earlier



Andrea – Who am I?

- Italian in Madrid
- Translation PM Background
- QA Analyst at SQLI Spain
- Padel enthusiast



Laura – Who am I?

- Born in Barcelona
- 15 years in QA & Testing — now QA Manager at Nespresso
- The Little Prince collector: 100+ editions in 80+ languages collected over 20 years

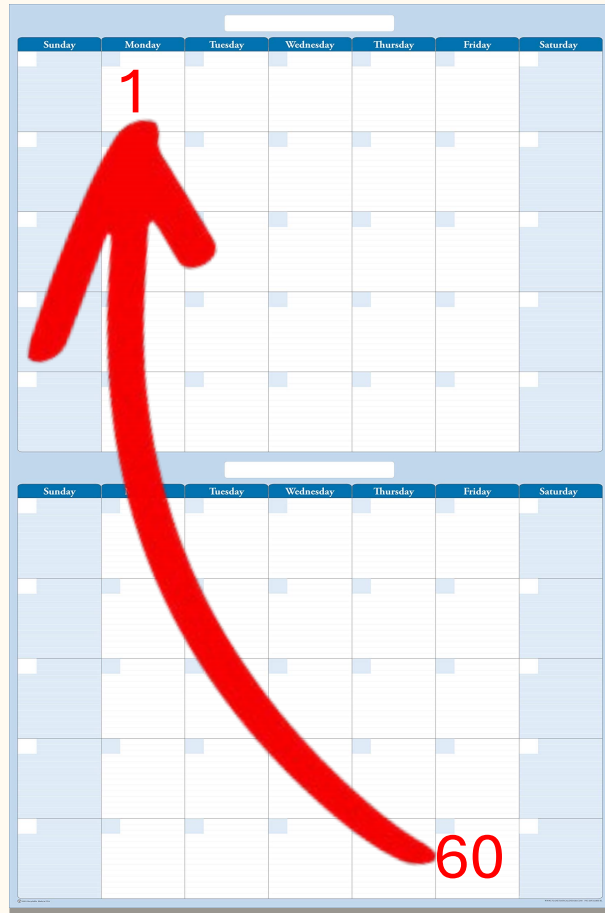


Who We Are

- Two colleagues who've always worked remotely.
- First time meeting in person: for this event!
- Proof that remote collaboration can work and lead to big wins.



What If We Told You...

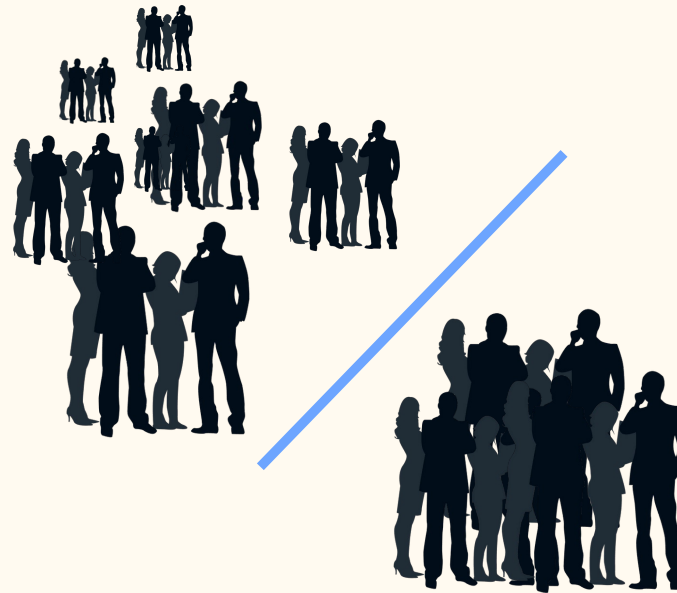


- We've managed to automate test cases 60 days earlier than before?
- And not just once — consistently.
- Want to know how?

Initial Context

Setup

- 10 Delivery Teams → Build new features
- 1 Regression Team → Test, maintain & automate after delivery



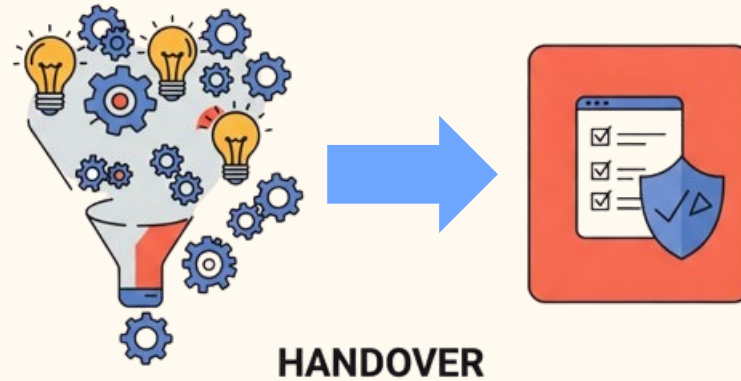
Why?

- Inherited from Waterfall days, with less teams
- Regression scope too large for Delivery Teams

Initial Context

Challenges

- Complex Handovers (Delivery → Regression)
- No QA automation in Delivery Teams
- Manual Regression scope too big

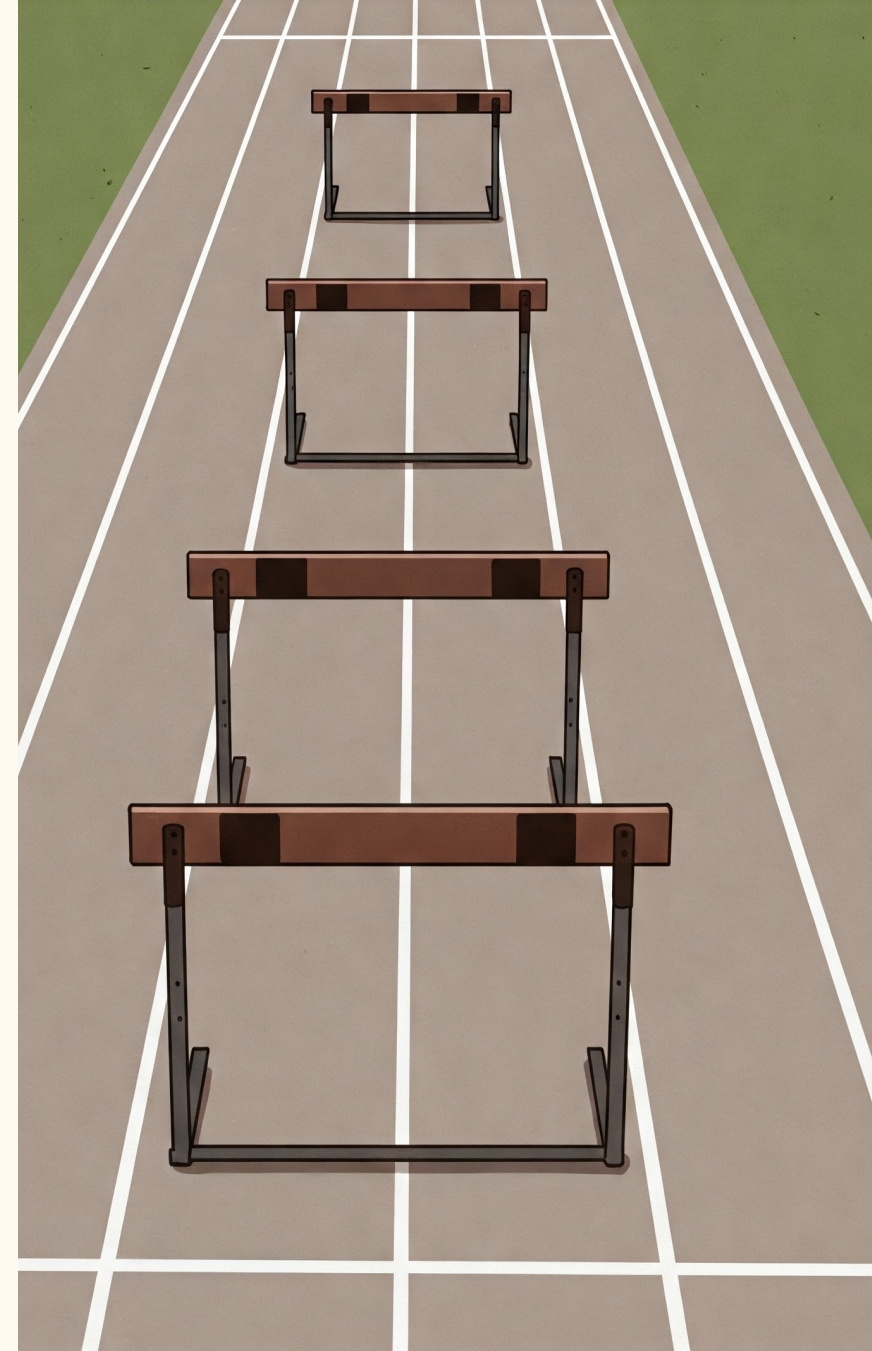


What we tried:

- Added automators to Delivery Teams
→ **Result:**
 - More automation, but fewer features delivered
 - No added value
 - Not efficient

Challenges

- Delivery Teams:
 - No time for automation
 - Automation was not a priority
- Challenging handover to Regression Team
- Complex Multimarket maintenance (+40 markets)
- E2E tests with manual steps

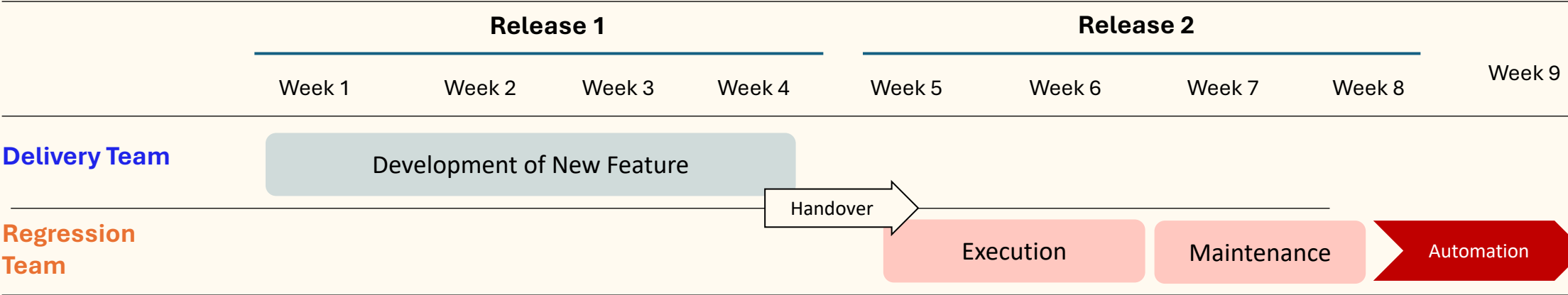


Challenges

Result:



Between development and automation



Goals

- Reduce execution & maintenance time
- Automate earlier
- Improve handovers



What's the most effective solution?



Solutions

New **DevOps Test Lead**

Change **Mindset**

QA Automators in Delivery Teams and Redesign of the **Strategy**

Less E2E, more **Modular** tests

Clear, Shared Language: **Gherkin**

Right automation **framework**

Solutions

New **DevOps Test Lead**'s mission

- Define Initial Plan:
 - Design smart testing strategies (Agile and DevOps mindset)
 - Assess frameworks, tools, and test case design quality
- Help on the transition of automation
- Coach teams to adopt the change

Solutions

Change Mindset:

- Full code and UI not needed to start Automation
- What do we need for automating?

**Clear
Preconditions**

**Amount of Test Cases
Needed**

Expected results

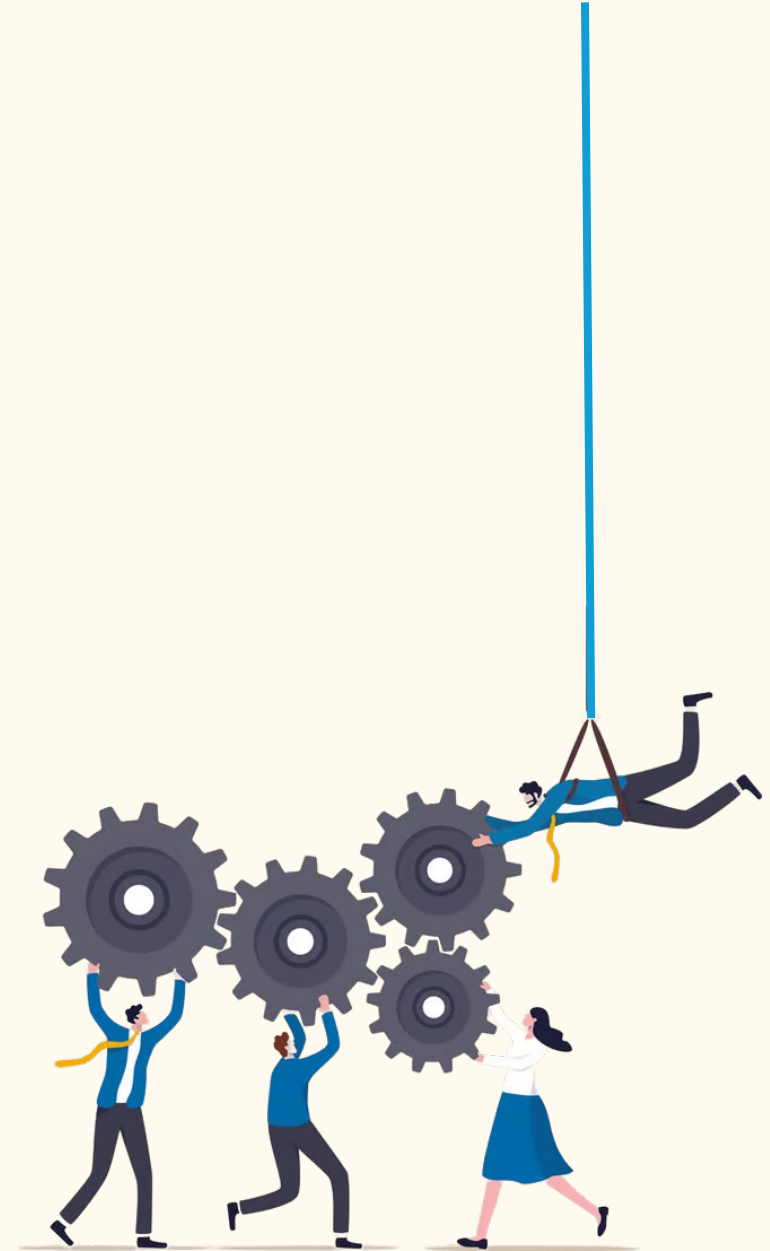
Change Definition of Done:

- Feature and automatic tests delivered together (**In-Sprint Automation**)

Solutions

Change teams' setup and reshaping of the strategy:

- QA Automator in Delivery Teams from Regression Team
- Triad Meeting (BA, Dev, QA)
- QA Automator spots issues early
- Automate tests in parallel with development
- Delivery Team owner of all testing activities (also Regression)
- Handover no longer needed



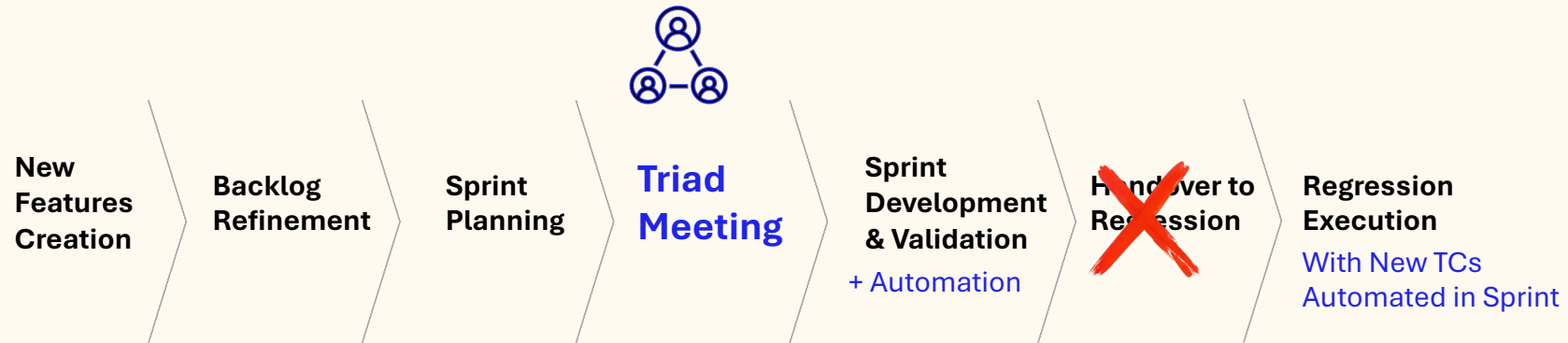
Strategy

Before



Strategy

After (with Automation First Approach)



Delivery Team



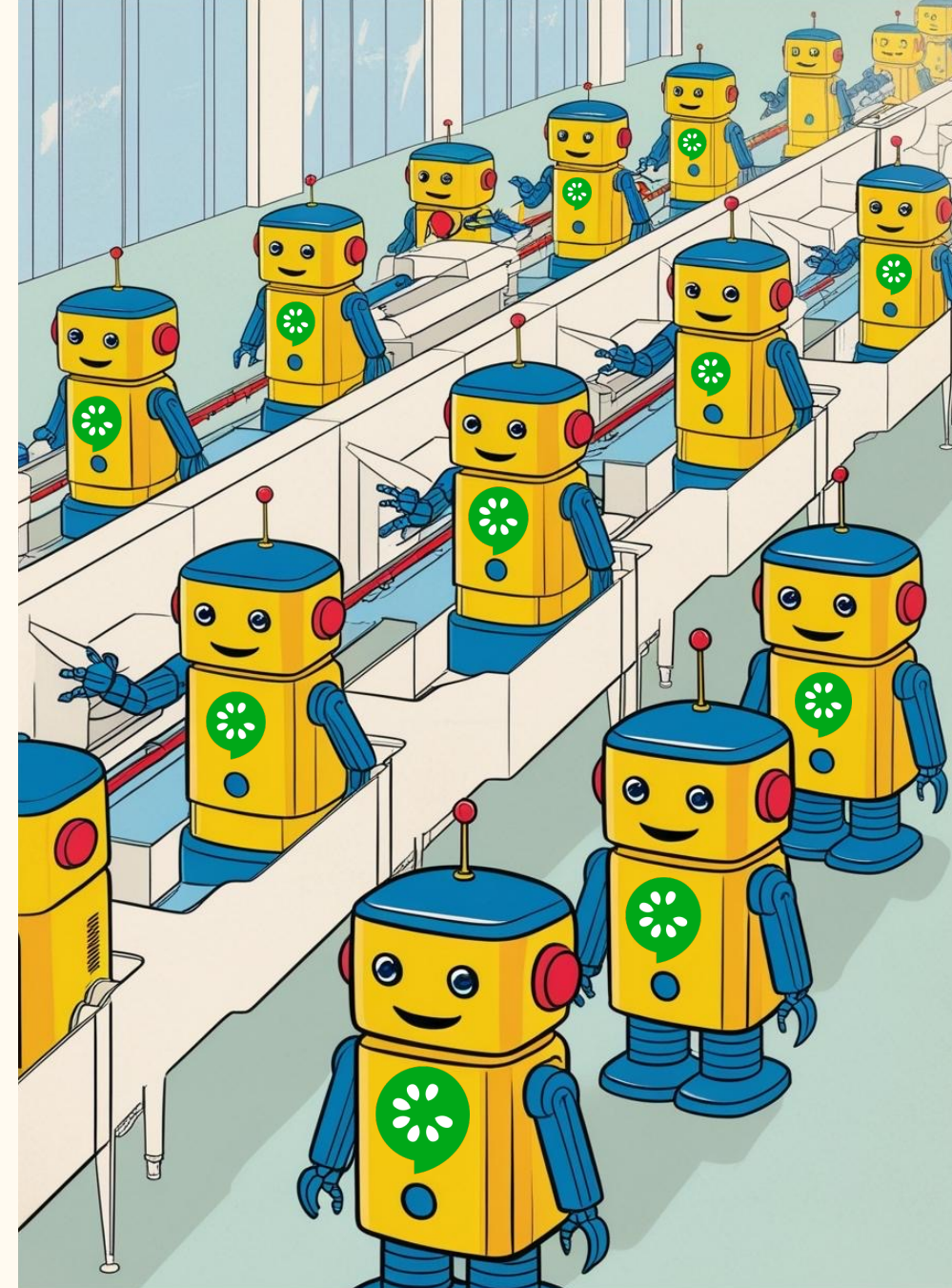
Solutions

Less E2E, more Modular tests:

- Cut down End-to-End tests
- Gherkin language:

Easy to read for all
stakeholders

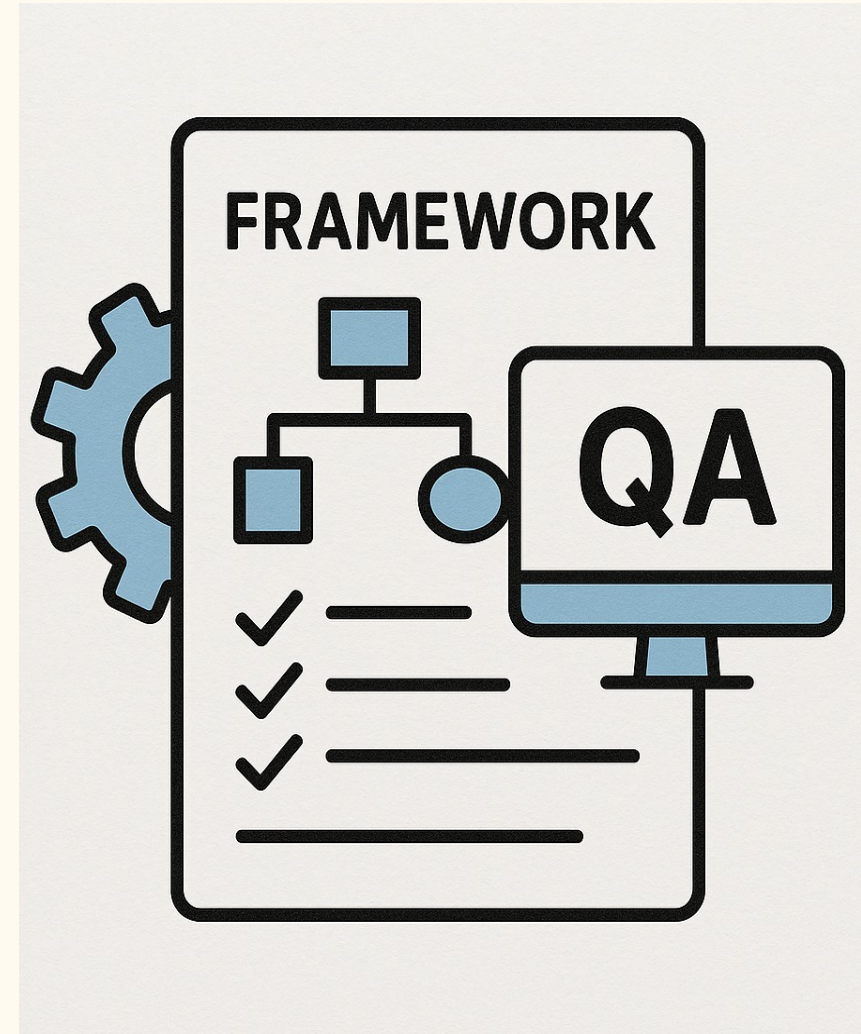
Focus on what to
test, not how



Solutions

Choosing the right framework:

- Clear vision for Test Automation
- Benchmark requirements
- Execution and outcome
- Training to the current teams



Choosing the Right Framework

Our vision for Test Automation:

- Full or partial regression anytime, anywhere
- Results within hours
- Consistent and reliable results
- Multi-market
- One framework for UI and API testing
- Easy integration (CI/CD, BrowserStack, Jira and Xray)
- Scalable, maintainable, developer-supported






Choosing the Right Framework

Criteria:

Integration with Nespresso environment	Integration with Third Party applications	Versatility	Initial Investment
Jenkins	TDM Icaria	Speed	Licences cost
Bitbucket	BrowserStack	Parallel Execution	Setup and maintenance
Jira Xray		Reporting	Training Effort
		Debugging	Required skills
		Reusability	New Hires
		AI Support	
		Customization	
		Maintenance	






Choosing the Right Framework

Candidates:

 <p>AFT Se</p>	 <p>Playwright</p>	 <p>cypress</p>	 <p>HP UFT</p>	 <p>Katalon</p>
Internal, Selenium, SikuliX, Java based	Open-source from MS, user-friendly	Open-source, modern approach, JavaScript	Well known, with Object Repository, good with E2E	AI and TestOps integration

Choosing the Right Framework

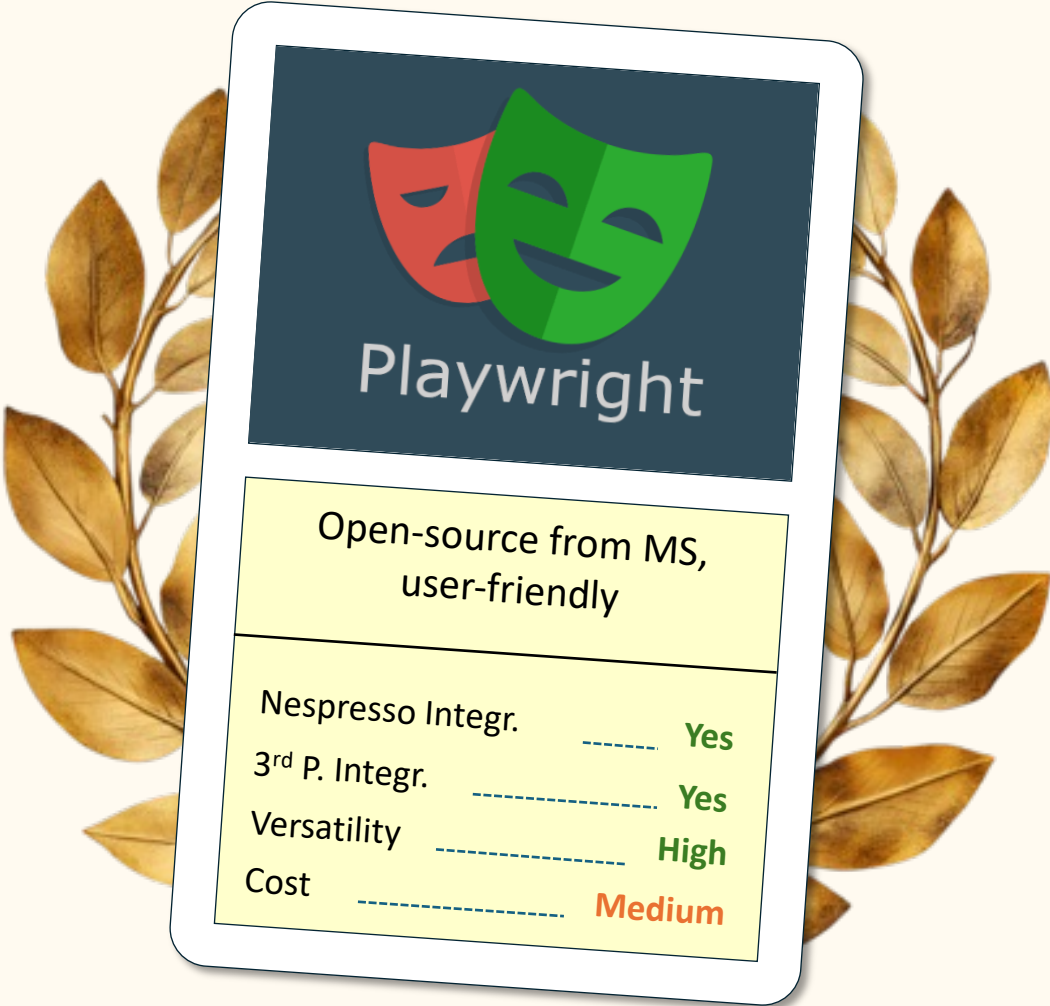
Candidates:

 <p>AFT Se</p> <p>Internal, Selenium, SikuliX, Java based</p> <p>Nespresso Int. ----- Yes 3rd P. Integr. ----- Yes Versatility ----- Medium Cost ----- Medium</p>	 <p>Playwright</p> <p>Open-source from MS, user-friendly</p> <p>Nespresso Integr. ----- Yes 3rd P. Integr. ----- Yes Versatility ----- High Cost ----- Medium</p>	 <p>cypress</p> <p>Open-source, modern approach, JavaScript</p> <p>Nespresso Int. ----- Yes 3rd P. Integr. ----- Yes Versatility ----- High Cost ----- High</p>	 <p>HP UFT</p> <p>Well known, with Object Repository, good with E2E</p> <p>Nespresso Int. ----- Yes 3rd P. Integr. ----- X Versatility ----- Medium Cost ----- Low</p>	 <p>Katalon</p> <p>AI and TestOps integration</p> <p>Nespresso Int. ----- Yes 3rd P. Integr. ----- Yes Versatility ----- Medium Cost ----- Medium</p>
---	---	---	--	---

Choosing the Right Framework: Playwright

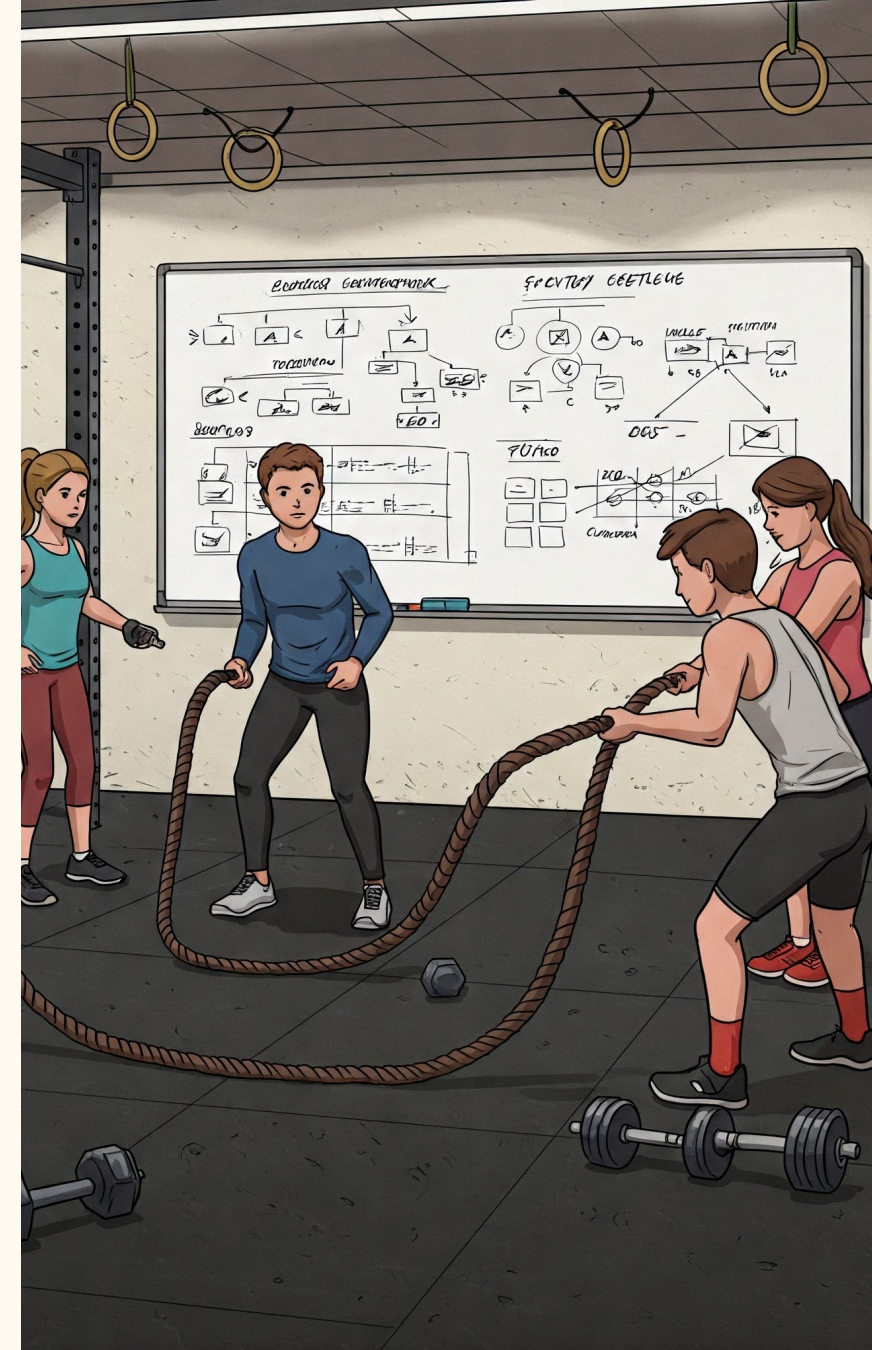
Why we chose Playwright:

- Advanced parallel execution
- Native API & UI testing
- Seamless CI/CD integration
- Simpler maintenance
- Developer support (Microsoft)
- Supports Multimarket thanks to Page Manager fixture
- Low learning curve



Action plan

- Build Playwright framework for Nespresso
- Train existing teams
(business & tools knowledge)
- Introduce AFA strategy & mindset to teams
- Udemy courses: TypeScript & Playwright
- Clear framework documentation
- DevOps Test Lead: support & code reviews



Results - Existing teams

- 3 existing teams (Regression scope)
- Tests rewritten in Gherkin + Automated with Playwright
- Baseline vs. New results:



**50% faster
Automation Time**

**70% faster Execution
Time
→ Quicker Fixes**

Results - New teams

- New development
- First domain fully built with Automation First Approach
- No existing Regression scope

Rollout smooth and fast



All Regression test cases automated before Go-Live



**5 markets validated in 10 minutes
(15 Tests/market)**



**Regression runs easily every sprint,
from Sprint 1**



The Journey Continues: Expanding the Automation First Approach

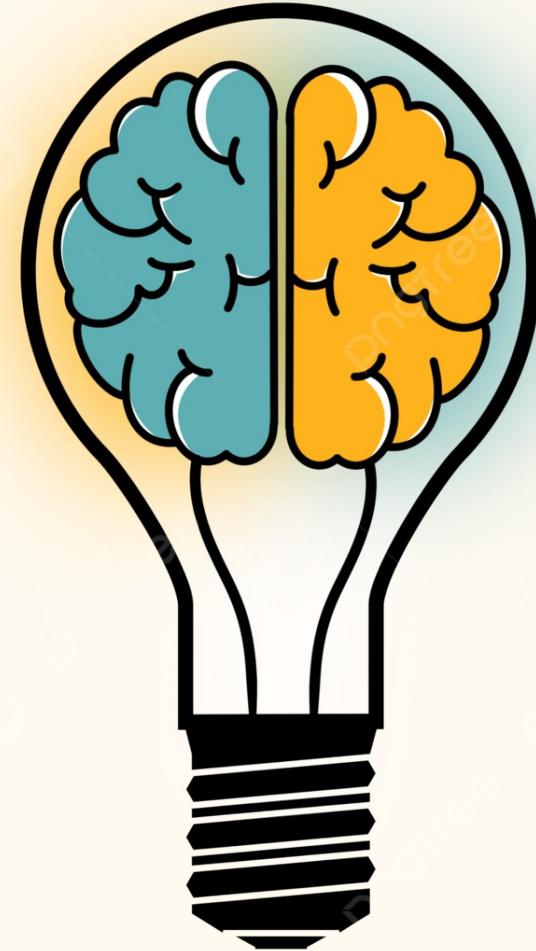
- Strong Results, Rollout Ongoing
- Automation from Day One for New Delivery Teams
- Next Focus: Integrating with Third Parties
(API-based Methods, Contract Testing and Stubs)
- Keep reinforcing mindset shift

Main Takeaways

- Shift from Monolithic to Modular testing
- Standardize with Gherkin
- Start automation early — don't wait for code
- Choose the right tools (Playwright)
- More efficient setup of the teams

It's All About Mindset

"Do you have the right mindset?"



Questions?
Let's keep the conversation going!



expo **QQA** 25

MADRID
May 20th,
21st & 22nd
2025

Thank you for attending

expoqqa.eu