

CASE STUDY / ECOHUB AG

# How EcoHub cut regression testing time by 75% and eliminated production surprises with Autom

---

EcoHub transformed its release validation from manual coordination and scattered reporting into a centralized testing process that gives QA, product owners and technical stakeholders a shared view of release readiness.

“Autom has made our release process more efficient by providing us with a structured and transparent way to coordinate testing across systems. The biggest benefit for us is the time savings: less manual effort, fewer coordination rounds, and faster access to the information needed for release decisions. By integrating Autom into our infrastructure, we have been able to scale our release activities more effectively while maintaining the level of quality and trust that our teams expect.”

— Steffen Lenz, CIO, EcoHub AG

#### AT A GLANCE

---

**75%**

reduction in API  
regression coverage time

**50%**

faster testing on EcoHub  
Essentials

**~1 week**

per module to migrate  
existing frontend test  
cases

## Testing was fragmented, visibility was poor, and gaps kept reaching production

---

### No centralized testing layer

Automation existed in separate individual setups, maintained by a single QA on a local machine. Results were fragmented, and there was no shared results platform for QAs, Product Owners and technical leadership. Adding new products created recurring coverage risk, because additional validation still depended heavily on manual effort.

### Status required a meeting

Developers received results late, slowing fixes and forcing rework weeks after implementation. Product Owners had to coordinate directly with QAs to understand pass or fail status, and reports had to be manually compiled each time. Every update cost someone time.

### Coverage gaps surfaced at the worst moment

With five environments to cover and limited QA bandwidth, prioritization was challenging. Product Owners could begin user acceptance testing and hit a broken flow on the first click. The problem was not the testers. It was the volume of manual work relative to the time available to do it.

“We wanted a test automation tool where we have centralized reports, a URL where everybody has the same transparency over test cases, without somebody needing to create manual reports.”

## A testing setup built around EcoHub's reality, not the other way around

---

Steffen had used Selenium-based frameworks across previous roles and knew the ecosystem well. Selenium is widely adopted, has strong community support and provides a proven foundation for automation. For EcoHub, however, Selenium alone was not enough. The requirement was a structured framework around that foundation, with centralized reporting, controlled versioning and a clear path toward automated deployments across multiple environments.

Autom was already available as an off-the-shelf platform. The decisive advantage was its flexibility: Autom could be extended for EcoHub's unique systems without forcing EcoHub to build and maintain a separate custom framework. This mattered because EcoHub's SAF is a proprietary framework developed by EcoHub. Its testing requirements are highly specific and not supported natively by standard tools.

EcoHub's SAF testing sits on a Kafka-based, encrypted messaging standard. Every alternative would have required a custom extension that EcoHub would own and maintain indefinitely. With Autom, the extension became part of the platform approach, developed together with the Autom team and aligned with EcoHub's actual infrastructure.

## A tailored integration for a highly specific infrastructure

EcoHub's backend testing requirements were not standard. The SAF framework handles encrypted secure data exchange and enables file exchange between brokers and insurers. Built on Kafka, it follows protocol requirements specific to the Swiss insurance market.

Autom worked directly with EcoHub's development team to support SAF message handling, encryption and decryption flows, and the backend API layer that sits underneath the platform. This allowed QAs to write and run test cases against the actual infrastructure instead of testing around it.

The result was an integration built into Autom, not a workaround bolted onto a tool designed for something else. Test cases can be executed against EcoHub's real environment, and results flow into centralized reporting that is accessible to QA, Product Owners and technical stakeholders.

## RESULTS

# Measurable improvement across every active workstream

---

75%

**Reduction in API regression  
coverage time**

From 4 hours to 1 hour per sprint  
deployment on SAF and Kafka-  
based testing

50%

**Faster testing on EcoHub  
Essentials**

Since 50% of modules are now  
covered and running in Autom

Migration of existing test cases to Autom now runs at approximately one week per module.

---

**Production surprises eliminated.** Since Autom went live on the SAF layer, post-deployment issues that used to surface during Product Owner testing have stopped. Every deployment is validated against the same coverage baseline, on every environment, every time.

## Less coordination overhead. Faster feedback. More time on what matters.

---

### **Status updates became a URL**

What previously required coordinating a meeting and waiting for a manually compiled report now takes a single link. Product Owners, the CIO and QAs look at the same dashboard. The quality score is visible without anyone having to compile or send it.

---

### **Developers fix faster**

Test results now reach developers while they still have the context of what they built. A bug found the same day it was introduced is faster to resolve than the same bug found two weeks later, when the developer has moved on to another topic.

---

### **QAs focus on new coverage, not repetition**

With regression testing automated and consolidated, the QA team's capacity has shifted. Repeating the same manual test across five environments on every deployment is no longer where their time goes. They write new test cases, extend coverage and improve depth instead of cycling through what is already known to work.

Autom also helped reduce tool fragmentation for QA and release validation. Instead of relying on multiple disconnected sources, such as spreadsheets, Azure DevOps and M365 Loop, the team can use Autom as a central source of truth for test execution and reporting.

With Autom, the QA team gained more capacity to expand test coverage, design stronger automation flows, define custom parameters and support quality gates in CI/CD pipelines.

## One case, multiple stakeholder outcomes

---

### C-level and business decision makers

A clearer release decision basis, less reliance on manually compiled status, and better transparency across critical workflows before release.

---

### Heads of Engineering and Product

Earlier feedback, fewer coordination loops, and a repeatable quality gate that can grow into CI/CD and release automation.

---

### QA and Engineering teams

Less repetitive regression work, more time for new coverage, and a central platform for test execution, reporting and test-suite improvement.

**See what Autom can do for your release process.**

Book a demo to understand where your biggest release-readiness gaps are today.





EcoHub AG is a Swiss technology and platform provider for the insurance and broker ecosystem. The company operates a secure data-exchange platform connecting brokers, insurers and other market participants. EcoHub reaches approximately 65–70% of the Swiss broker market, with many of Switzerland's major insurers connected to the platform.

EcoHub's technical organization is responsible for the development, operation, testing and security of its platform landscape, including API-based and event-driven services that enable secure and standardized data exchange between market participants.