

# CASE STUDY

## *“Automating Unit Tests in SAP”*

By Josh J. Jacob (SAP & DevOps Expert @DevOn)



## Customer profile

The customer is one of the largest energy suppliers in Europe with strong footsteps in sustainable and secure energy supplies for over two million customers. Their total revenue is over 4.0 billion euros and manages a well-maintained network of windmills, solar energy projects and biomass plants.

## Customer situation

They have a varied portfolio of energy supplies with a huge IT landscape. In order to manage all these (legacy) systems, the customer has outsourced a part of their software delivery to a third party which uses SAP.

*“Per sprint, we used to spend a full week on testing and fixing bugs”*

At the time, the development teams didn't run any unit tests and were only performing a large end-to-end acceptance test in their development systems. Therefore, they spent a lot of time on testing and fixing bugs. Even minor bugs had to wait until the user acceptance test was done. Because of all the many delays, the customer regularly spent a full week on testing and bug fixing in their 3-week release cycle.

## Requirements

The customer wanted visibility around the SAP processes by building a pipeline where they can orchestrate the development process and the releases. The main goal of this program was to improve quality and reduce the time to market. In here, the first step was the automation of unit tests in SAP.

*“The challenge was to automate the Unit Testing process in SAP”*

Because the SAP landscape was set up and maintained by a third party, the customer needed an external consulting partner with the right expertise and knowledge to help.

They chose DevOn

---

## Approach

In order to get a clear understanding of the situation and the customers pain points, DevOn performed an assessment of their business- and test processes, SAP landscape and their architecture.

Based on the assessment we noticed that, due to the complex landscape and dependencies of the software systems, the orchestration and setup of the CI/CD pipeline was really challenging. However, we saw an opportunity to improve the software quality and speed up the delivery. Therefore, we created a framework where automated unit testing in SAP would be the first priority.

By setting up a pipeline in GitLab, the customer would be able to orchestrate the entire process from there, including automated testing. In order demonstrate this approach we performed a small scale proof of concept where we set up a complete pipeline in GitLab.

After demonstrating the proof of concept, DevOn started training and coaching several teams how to automate unit tests with SAP. At the same time, we taught the CICD department how to roll out this approach with the other teams so they would not be dependent of a third party.

The CICD team of the customer is able to train their **own teams** in the future. So they are **not dependent** of a third-party



92% less time spent on fixing bugs



65% less bugs

## Results

*“We reduced the amount of bug with 65% by automating the unit tests”*

The SAP program resulted in several major improvements. Due to the training of the teams, they are able to perform unit tests in SAP. This created a higher sense of ownership within the teams and reduced the amount of bugs with more than 65%.

*“92% less time spent on fixing bugs”*

Moreover, the developers are not dependent on the end to end acceptance test anymore. As a result, the teams are now able to locate and solve bugs earlier. The bug fixing process has been drastically reduced with 92% .

## End state

Development and Operations personnel are able to make secure decisions before a piece of code is written or deployed.

Security checklists became one of the important artefacts in deciding whether a feature is production ready or not.

Commonly found vulnerability identification was automated to be repeatable tests with custom policies so vulnerabilities are identified right during the build process.

## Key take-aways

**The importance of mindset:** Don't underestimate the importance of the teams' mindset when you would like to make changes. Make sure the team is willing to change, otherwise it won't happen.

**SAP orchestration with SOLMAN:** It is possible to orchestrate the full pipeline as proved in the proof of concept. However, because of third party dependencies, it was not feasible to log all changes and results in GitLab. Therefore we advised the customer to use SAP's own Solutions Manager (SOLMAN). SOLMAN can generate internal tasks (Pipeline) to execute the required checks (Unit Tests, Code Coverage, Code Inspector) and can handle the automated deployments.

**GitLab improvements:** The proof of concept for the GitLab pipeline can be improved by adding GitLab pages or external BI dashboards to publish the results. After close examination, it was concluded that If done within GitLab, might increase the complexity, overhead and can cause performance issues for GitLab.

**Devon framework:** Scale-up the framework program to orchestrate the pipeline processes and to act as a dashboarding tool.

Curious how we can Automate your SAP Unit Tests?  
Contact us and schedule a free call with our SAP Expert!

[Schedule a call](#)

