

# Implementation of a microservice-based certification platform

Sebastian Copei<sup>1\*</sup> and Manuel Wickert<sup>2\*</sup> and Albert Zündorf<sup>3</sup>

<sup>1</sup> Kassel University, Kassel, Germany,  
sco@uni-kassel.de

<sup>2</sup> Fraunhofer IEE, Kassel, Germany,  
manuel.wickert@iee.fraunhofer.de

<sup>3</sup> Kassel University, Kassel, Germany,  
zuendorf@uni-kassel.de

**Summary.** The digitalization of the energy sector is a transformation process to support the decarbonization of the energy system. One of the big challenges for the digitalization in the energy system is the connectivity to edge devices (e.g. e-mobility, heat pumps, PV, etc) for different vendors. Therefore interoperability between different edge devices, backend-systems, and local energy management components is important. In the Horizon 2020 project interconnect interoperability between such smart devices in the energy sector will be examined. Therefore standardized communication protocols will be used and edge devices will be certified to prove compatibility with the standard. Here we will present a microservice-based certification platform to enable certification in agile software development processes.

**Key words:** microservices, standardization, certification, agile

## 1 Motivation

The transformation of the energy system is becoming a key driver of the digitalization of the energy system. On the one hand, digitalization supports a real-time view of all producers and consumers of the complete energy system. Therefore smart meters will become an important role. On the other hand, will digitalization enable the compensation of fluctuations, caused by wind and photovoltaic (PV) feed-in, by aggregating huge numbers of controllable consumers and energy storages e.g. electrical vehicles (EVs).

One big challenge in digitalization is the connectivity to such edge devices. In contrast to the Endconsumer sector, the energy sector use edge devices with significantly longer planned lifetimes. Furthermore, backend-systems and edge devices are often built from different vendors. Therefore interoperability between these devices and backend-systems is very important. To provide interoperability standardized communication protocols (e.g. IEC 61850 [4], IEC 60870-5-104 [3], OPC UA [6]) are common for such communication scenarios. Edge devices

---

\* These two authors contributed equally

typically will be certified to verify that they are compliant with a certain version of the standard. However, standardization and certification processes are classically very time-consuming processes and therefore not compatible with agile development processes. In [2] we proposed a new standardization and certification approach that can be integrated into agile development processes for edge devices as well as for backend-systems.

In Horizon 2020 project interconnect the communication between edge devices as well as backend-systems should be enabled by SAREF. SAREF is an ontology to enable interoperability between smart devices in the energy sector. The SAREF Standard will be extended and used during the project to provide a connection between EV-charging stations, home energy management systems (including smart meters), and superior operation systems. To enable agile development of the standard extensions as well as all the edge device and backend components, we implemented a certification as a service platform. This platform should be used within the project to verify that each component is compliant with a certain version of the SAREF standard.

In 2 we give a short introduction to our agile certification approach from [1]. The section 3 describes our certification as a service platform. The Evaluation and the next steps are in sections 4 and 5.

## 2 Agile certification

In contrast to the state-of-the-art standardization and certification process, we proposed an agile standardization and certification approach. The basic idea is to make small adaptations to existing versions of a communication standard. The small adaptations lead to newly published versions of the standard. This will increase the number of standards but also decrease the time to integrate new requirements into a communication standard. With a new communication standard version, a few conformance tests should be adapted or created. The conformance tests will be used to verify that components are compliant with the adapted version of the standard. In figure 1 the agile process is shown by two inter-winded cycles. On the left side, the standardization process is shown. The right side shows the implementation of a component like an edge device. The certification of the component builds the bridge between both cycles.

From the perspective of the software development process of edge devices or backend-systems, the certification should be part of a continuous integration or deployment process (CI/CD). Therefore the certification has to be fully automated, including conformance tests, evaluation, and the issuance of the certificate. A certification body may offer all these functionalities by a certification service. The software vendor by themselves may use this service by calling it from a CI / CD Pipeline (see figure 2).

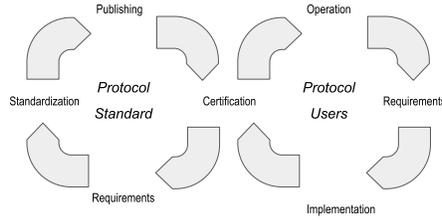


Fig. 1: Agile Standardization Process

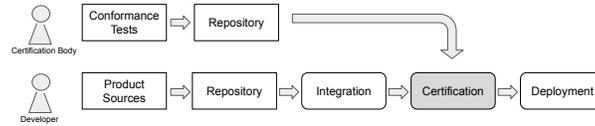


Fig. 2: Continuous Certification Pipeline

### 3 Certification Platform

To support certification bodies, to provide certification services, we developed a certification platform. The platform is implemented based on our proposed microservice architecture in [1]. We implemented five backend-services in *NestJS* and a frontend-service in *Angular*. The platform is developed as open-source and is available under GitHub<sup>2</sup>. For the communication between the backend-services, we used *Apache Kafka*. The user management is realized with *Keycloak*. As mentioned in [1] the certification platform takes a software artifact during the test phase of a build process and performs compliance tests of a standard.

For our implementation, we used *Docker* images to provide an artifact. So a smart energy device e.g. an electrical charging station control software could be delivered as a container to our platform. Therefore we provide a *Docker* image to exchange software artifacts between the CI Pipeline and the Certification platform. During the test phase of the CI Pipeline, the certification platform is called via a REST API. Besides the *Docker* image the version of the communication standard is provided to declare which compliance tests should be performed. The artifact service starts a *Docker* container of the provided image. Afterward, the test-suite service runs the containers for certification. All running and started containers are managed by the job-runner service. If all tests will be passed, the certification services sign the provided *Docker* image from the CI Pipeline with a signature to ensure the correctness of the certificate. Finally, the signed container is provided back to the pipeline. This enables an agile certification during agile software development.

In Figure 3 we summarize the current platform and services. As mentioned before, there are services for managing artifacts, tests, certificates, and runnable test environments. The latter service also provides the API for the CI integration.

<sup>2</sup> <https://github.com/sekassel-research/caas-platform>

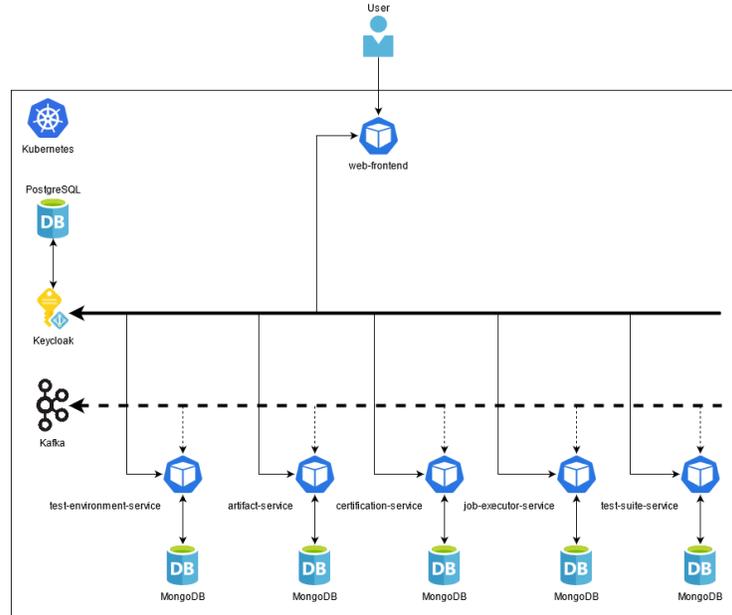


Fig. 3: CaaS Platform

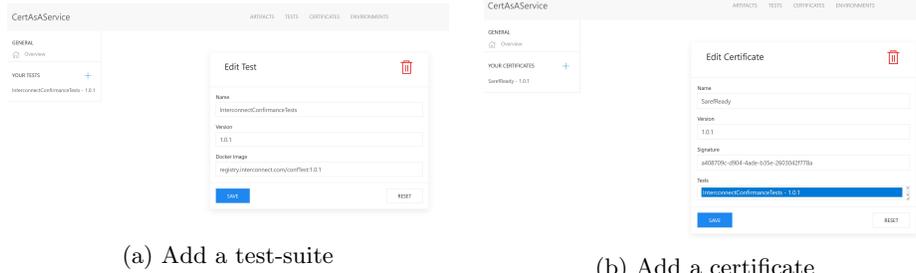
## 4 Evaluation

The current implementation allows providing artifacts, test suites, certificates, and run-able test environments. These features are tested with integration tests. Also, it is possible to start a single artifact with a single test-suite to achieve a certificate. For the evaluation of this workflow, we created a simple scenario from the context of the interconnect project.

*The GoodEnergy AG wants to implement a new service for the interconnect project. The be interoperable with the other system components our SAREF communication extension has to be implemented. After the developer of the GoodEnergy AG provided a version of their service, they want to use the certification service to test and certify their service for usage in the interconnect project.*

The certification service is provided by a project internal certification body. The certification body will use our platform and a test suite with all conformance tests to provide this service. In figure 4a you can see the newly created test-suite. To use this test-suite in the further process, the certification body needs to map the standard version number to the first created test-suite. This is shown in figure 4b.

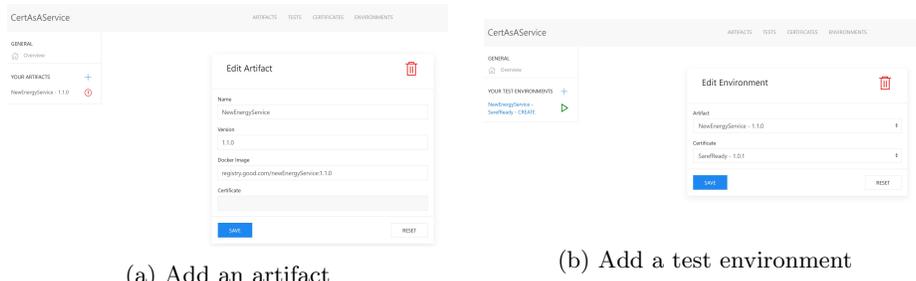
After the certification body created a test suite and maps it to a version number, the GoodEnergy AG can use the certification service. Figure 5a shows how software can be uploaded via UI, instead of using the REST API directly from the CI/CD Pipeline. In the next step, you choose the version number, for which certification has to be performed. Figure 5b shows a created test environment for



(a) Add a test-suite (b) Add a certificate

Fig. 4: Actions as a standardization committee

the uploaded software and version number. Finally, the conformance tests can be performed, tests will be evaluated and the certificate could be issued. Here a certification corresponds to a signed Docker-Image. The certification service signs the image, so it is possible to verify that a current container content is compliant with a certain standard version.



(a) Add an artifact (b) Add a test environment

Fig. 5: Actions as GoodEnergy AG

The UI certificate download is currently a work in progress. Therefore the development of a process can not completely be evaluated for the charging station. However, the scenario shows how the developed platform will support agile certification of communication standards by charging stations as an example.

## 5 Future Work

The current implementation does not contain a relevant number of test cases for the charging station certification. Therefore we plan to implement these tests and evaluate the certification platform during the further development of smart components. Furthermore, we want to implement the possibility to provide a system or a composition of multiple services as an environment to test a service

interaction within a system. Finally, we want to evaluate the tool within an industry case study from the energy domain.

## 6 Acknowledgement and Disclaimer

This Publication is part of a project [5] that has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement N<sup>o</sup>857237. The sole responsibility of this publication lies with the author. The European Union is not responsible for any use that may be made of the information contained therein.



## References

1. Sebastian Copei, Manuel Wickert, and Albert Zündorf. Certification as a service. In Maria Paasivaara and Philippe Kruchten, editors, *Agile Processes in Software Engineering and Extreme Programming – Workshops*, pages 203–210, Cham, 2020. Springer International Publishing.
2. Laura Daniele, Frank den Hartog, and Jasper Roes. Created in close interaction with the industry: The smart appliances reference (saref) ontology. In Roberta Cuel and Robert Young, editors, *Formal Ontologies Meet Industry*, pages 100–112, Cham, 2015. Springer International Publishing.
3. IEC 60870-5-104 - Telecontrol equipment and systems. Standard, International Electrotechnical Commission, Geneva, CH, 2006.
4. IEC 61850 Standard Series, Communication networks and systems in substations. Standard, International Electrotechnical Commission, Geneva, CH, 2020.
5. Interconnect project - homepage. Last viewed 20.04.2021. URL: <https://interconnectproject.eu/>.
6. OPC Unified Architecture, IEC 62541, Standard Series. Standard, OPC Foundation, International Electrotechnical Commission, Scottsdale, USA, 2008.