**Customer Requirements Specification**

**(Lastenheft)**

(TINF19C, SWE I Praxisprojekt 2020/2021)

Project: OPC UA Server Farm

Customer: Rentschler & Holder

Rotebühlplatz 41

70178 Stuttgart

Supplier: Team 3 (Niclas Hörber, Kay Knöpfle, Nico Fischer, Daniel Zichler, Niklas Huber, Phillip Förster)

Rotebühlplatz 41

70178 Stuttgart

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comment** |
| 0.1 | 16.10.2020 | Niklas Huber | created |
| 0.2 | 17.10.2020 | Niklas Huber | Copied goal from the task |
| 0.3 | 22.10.2020 | Niklas Huber, Kay Knöpfle | Added Product Environment |
| 0.4 | 04.11.2020 | Niklas Huber | Added Business Process and Use Case |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

#### ****Offene Punkte****

CONTENTS

*1.* Goal 3

2. Product Environment 4

*3.* Product Usage 5

*3.1.* Business Processes 5

3.1.1. <BP.001>: <Name> 5

3.2. Use Cases 5

3.2.1. <UC.001> Use Case Name 6

*3.3.* Features 7

3.3.1. /LF10/ ….. 7

3.3.2. /LF20/ ….. 7

*4.* Product Data 8

4.1. /LD10/ ….. 8

4.2. /LD20/ ….. 8

*5.* Other Product Characteristics 9

5.1. /NF10/ ….. 9

5.2. /NF20/ ….. 9

5.3. System Environment 9

6. References 10

# Goal

The goal of this project is to develop a Server farm which supports the testing of OPC UA Clients. The Server farm should provide multiple virtual OPC UA Server via Network. These virtual OPC UA Server profiles should be parameterizable via an AutomationML configuration file (in CAEX 3.0). In conclusion, the software should simulate multiple OPC UA Servers on one computer for testing OPC UA Clients. The target group are developer and tester of applications with OPC UA Client-Interface. The documentation of the software as well as the documentation of the development and project is also part of the goal.

# Product Environment

OPC UA (Open Platform Communications United) is a standardized and safe exchange of data and information for industry automation between machines, devices, computers and services from different industry sectors. Therefore, it is independent of programming languages, operating systems and system suppliers [1].

The main parts of an OPC UA Environment are an OPC Server and Client. The Server is the foundation of the OPC communication, it implemented the OPC interfaces. The Client is the logical counterpart to the Server and can get data from the UPC Server. An OPC Test Client is a special Client that can test the function and configuration of a UPC Server [2]. In the context of this project, OPC UA Client-Expert [3] will be used as Test Client.

With CEAX 3.0 in AutomationML (AML) an OPC Server can be configured. AML is short for Automation MarkUp Language and it is the connecting element for the seamless automation planning. For that AML describes automation plans as objects and use various standards to describes them. The relevant standard in the context of this project is CEAX 3.0. It is used as to describe hierarchical structures and links. It is also XML based, a meta data format and it is standardized in IEC 62424.

This UPC UA Server Backend will be implemented via open62541-Stack. It is an open source implementation of OPC UA in the programming languages C99 and C++98, it´s platform independent. For that open62541 provides tools to implement OPC UA clients and servers [4].

The server farm will be running via one system. That means that all Server profiles will be simulated via one device. And this Server farm will be tested via the already mentioned Test Client.

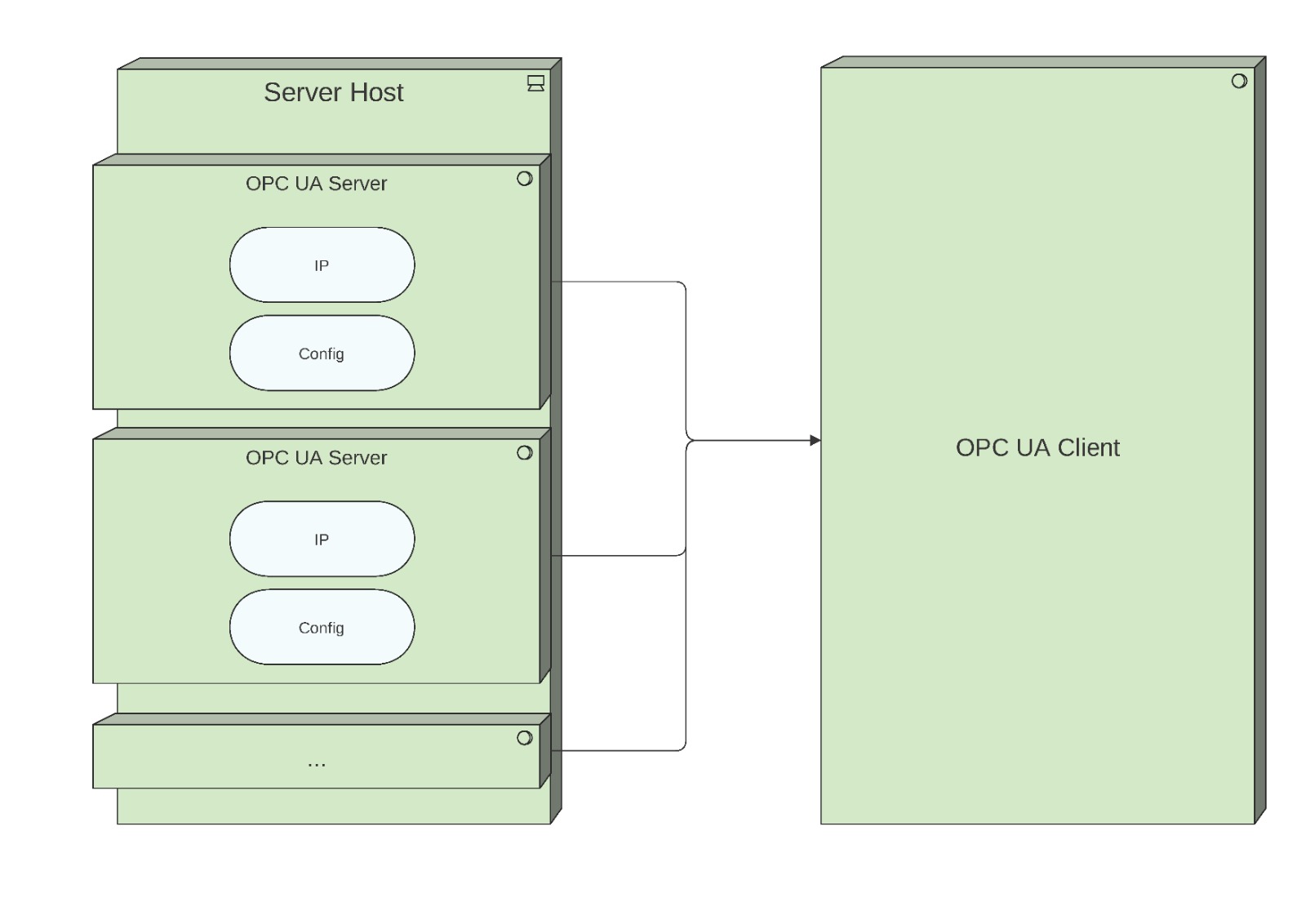


Figure 1 Product Enviornment

.

# Product Usage

The main purpose of the software will be the simulation and testing of multiple virtual OPC UA Servers which are simulated on one system. Each Server can be configured via sperate profiles and will start after finishing the configuration. From a user perspective, the main use case of this software is the testing of the simulated Servers. Clients testen ist der hauptusecase

The following business processes use cases and features shall be supported by the software.

## Business Processes

This section will handle the starting of the Server Farm, respectively of one Server.

### <BP.001>: Server interaction

|  |  |
| --- | --- |
| Triggering Event: | User want to start the UPC UA Server farm, respectively one Server with a profile. |
| Result: | The Sever starts with the profile. |
| Involved Roles: | User and UPC UA Server. |

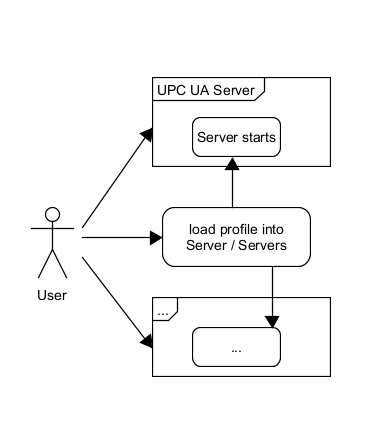


Figure 2: <BP.001> Server start (Unfertiges Diagramm, nur ob Prinzip richtig Kasten mit UPC Server Farm drum rum machen)

## Use Cases

This Project will implement a UPC UA Server farm, which runs on one device. The virtual OPC UA Servers will be parameterizable with the Server profiles. Each virtual Server will be able to be tested via OPC UA Client UA-Expert from the user.

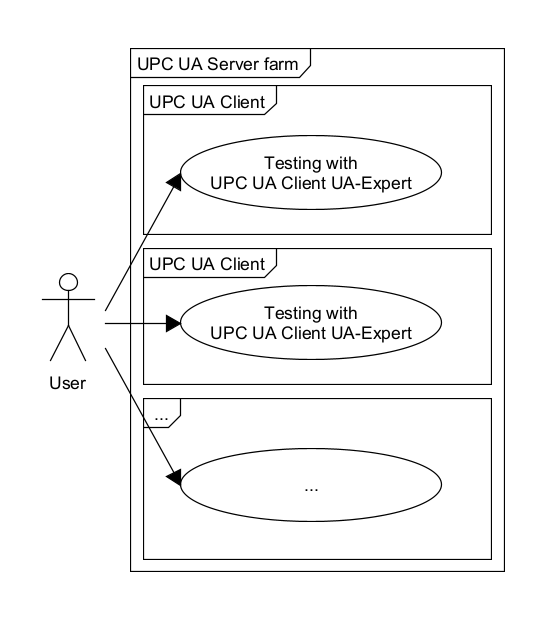
**

Figure 3: Use Case Overview Diagram

### <UC.001> Testing UPC UA Client(s)

|  |  |
| --- | --- |
| **Related Business Process:** | <BP.001>: Server interaction |
| **Use Cases Objective:** | User want to test multiple UPC UA Clients. |
| **System Boundary:** | The Server farm is the system boundary. |
| **Precondition:** | * OPC UA Servers must be started with the fitting profiles * Installation of Test Client |
| **Postcondition on success:** | The Servers must run without errors |
| **Involved roles:** | User, Test Client and UPC UA Servers (Server farm) |
| **Triggering Event:** | Start of the testing with the Test Client. + start der Serverfarm/Servers |

Aktivitätsdiagramm

User opens server

Users config files

…

## Features

In diesem Abschnitt sollen die bereits definierbaren funktionalen Features in „Balzert-Notation“ aufgelistet werden und nach den Regeln der Anforderungsschablone ausformuliert werden. à (<https://www.sophist.de/fileadmin/SOPHIST/Puplikationen/Broschueren/SOPHIST_Broschuere_MASTeR.pdf>)

Im Gibhub -> Server Ordner -> Features die der Open Stack bereitstellt

### /LF10/ …..

The [Component] shall {[adverb]} check if [Condition] and, {in case of violation trigger the appropriate reaction as specified in the fault database} {perform the following reactions:[List of Reactions]}

### /LF20/ …..

The [Component] shall {[adverb]} check if [Condition] and, {in case of violation trigger the appropriate reaction as specified in the fault database} {perform the following reactions:[List of Reactions]}

# Product Data

In diesem Abschnitt werden die Hauptdaten und Datenschnittstellen beschrieben, mit denen das Softwareprodukt arbeiten soll und die bereits identifizierbar sind (siehe Abb. 1). Im Allgemeinen werden diese Hauptdaten eines Programms auch nonvolatil gespeichert.

Mögliche Product Data?

Konfigurationdatei

Open62541-Stack

10 Server Profile

Test Client

System mit CAEX 3.0 Parser

## /LD10/ …..

…

## /LD20/ …..

….

# Other Product Characteristics

This section describes the already known non-functional requirements for the product.

…

Die Aufgabe dieses Abschnittes ist die Beschreibung der nicht-funktionalen Anforderungen. Dabei handelt es sich um Charakteristiken oder Qualitäten, die das Produkt attraktiv machen und es von vergleichbaren Produkten unterscheiden.

In diesem Abschnitt werden die wesentlichen Eigenschaften des zu entwickelnden Produktes beschrieben, die nicht direkt die zu leistende Funktionalität betreffen.

In diesem Abschnitt sollen diese bereits definierbaren Anforderungen in „Balzert-Notation“ aufgelistet werden und nach den Regeln der Anforderungsschablone ([*https://www.sophist.de/fileadmin/SOPHIST/Puplikationen/Broschueren/SOPHIST\_Broschuere\_MASTeR.pdf*](https://www.sophist.de/fileadmin/SOPHIST/Puplikationen/Broschueren/SOPHIST_Broschuere_MASTeR.pdf)) ausformuliert werden.

## /NF10/ …..

The software/system shall support …

## /NF20/ …..

….

## System Environment

This section describes the system environment required to operate the product.

…

Hier sollten alle wesentlichen und notwendigen Parameter der Systemumgebung (Hardware, Software) beschrieben werden, soweit diese bereits festlegbar ist.

# References

[1]ihttps://www.industry-of-things.de/was-ist-opc-ua-definition-architektur-und-anwendung-a-727188/

[2] <https://www.opc-router.de/was-ist-opc-ua/#OPC-Foundation-Video>

[3] https://www.unified-automation.com/products/development-tools/uaexpert.html)

[4] <https://open62541.org/doc/open62541-current.pdf>