Moduldokumentation

(MOD)

(TINF18C, SWE I Praxisprojekt 2019/2020)

Modul

GUI

Project: Profinet DCP Client als WEB-Applikation

Customer: *Rentschler & Ewertz*
Rotebühlplatz 41
70178 Stuttgart

Supplier: Team 2 (Jannik Schwarz, Sinan Yurttadur, Noah Broß, Nicolas Breuninger, Marvin Sonntag, Rene Scholz)
Rotebühlplatz 41
70178 Stuttgart

Version	Date	Author	Comment
0.1	28.04.2020	Nicolas Breuninger	First draft
1.0	06.05.2020	Nicolas Breuninger	Added information
1.1	14.05.2020	Nicolas Breuninger	Added Test information, final version

1. Content

2. History

Version	Datum	Autor(en)	Kommentare
0.1	28.04.2020	Nicolas Breuninger	First draft
1.0	10.04.2020	Nicolas Breuninger	First version
1.1	14.05.2020	Nicolas Breuninger	Added Test information, final version

3. Scope

The Module Documentation (MOD) describes the architecture, the interfaces and the main features of the module. It also describes the module/component test including the results. It can also serve as a programming or integration manual for the module. If there are some risks related to the module itself, they shall be noted and commented within this document.

Die Moduldokumentation beschreibt die Architektur, die Schnittstellen und die Hauptmerkmale des Moduls. Außerdem werden die Modul bzw. Komponententests einschließlich der Ergebnisse beschrieben und dokumentiert. Die MOD dient bei Bedarf auch als Programmier- oder Integrationshandbuch für das Modul. Wenn bestimmte Risiken direkt mit der Verwendung des Moduls verknüpft sind, so sind sie in diesem Dokument zu benennen und zu kommentieren.

4. Module Requirements

4.1. User View

This module is the frontend view. It is used as a source for input by the user as well giving information to the user. It allows for the application to have interactions with the user.

4.2. Module Context

This module requires the NetworkService-module for holding the network information.

5. Analysis

This module has the purpose to display the information retrieved by the backend.
The only dependency is on the NetworkService as it holds all the displayed information.

6. Design

To display all the information about profinet-devices mockups were created. The realization of these mockups tries to solve this task.

The architecture is determined by Angular. A angular-module consists of 3 separate file types:
.html
.scss
.ts

Link to the *source files* (https://github.com/nicolasbreuni/Tinf18C_Team_2_Profinet_DCP_Client/blob/master/SOURCE/dcp-client-frontend/src/app/app.component.ts)

Because of how angular connects views and variables this module does not have any APIs or connections other than the values of the variables in the NetworkService.

6.1. Risks

Risks include the failing to start the frontend modules.
Tasks of this module are network requests over the NetworkService module.

Risk reduction is not possible. Though the risk of this module failing is practically zero.

7. Implementation

The implementation first started with the mockup of the UI.
Then a recreation of the created mockup was developed. Finally an integration of the NetworkService was implemented.

8. Module Test

8.1. Module Testreport

Test-ID	Pass/Fail	If failed: Test Observation	Date	Tester
1	Pass		14.05.2020	Rene Scholz
2	Pass		14.05.2020	Rene Scholz
3	Pass		14.05.2020	Rene Scholz
4	Pass		14.05.2020	Rene Scholz
5	Pass		14.05.2020	Rene Scholz
6	Pass		14.05.2020	Rene Scholz
7	Pass		14.05.2020	Rene Scholz
8	Pass		14.05.2020	Rene Scholz
9	Pass		14.05.2020	Rene Scholz
10	Pass		14.05.2020	Rene Scholz
11	Pass		14.05.2020	Rene Scholz

9. Summary

This module displays the data retrieved by the backend.

Streghts:
This module doesn’t use any third-party-software. Everything is developed in native Angular