*Project*

AAS-Webclient

**MOD05: Support of Bill of Material**

TINF22F, Software Engineering I Praxisproject 2023/24

*Customer*

Markus Rentschler, Ivan Bogicevic

Lerchenstraße 1, 70178 Stuttgart

*Supplier*

Armin Taktar (Project Leader)

Lara Lorke (System Architect)

David Bauer (Product Manager)

Ümmühan Ay (Documentation)

Rafael Sancho Pernas (Developer)

Kyle Zieher (Test Manager)

Rotebühlplatz 41, 70178 Stuttgart

*Author*

Lara Lorke 10.05.2024

**CONTENTS**

[1 Scope 3](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696481)

[2 Glossary 3](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696482)

[3 Module Requirements 3](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696483)

[3.1 User View 3](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696484)

[3.2 Requirements 3](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696485)

[4 Analysis 3](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696491)

[5 Design 4](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696492)

[6 Implementation 4](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696493)

[7 Module Tests 5](file:///C:\Users\sanchora\Downloads\TINF21C_MOD01_Team_1_1v0.docx#_Toc134696494)

# 1 Scope

The module "Support of Bill of Material" is responsible for processing and displaying the Hierarchical Structures enabling Bills of Material concerning a AAS object.

# 2 Glossary

* BoM: Bill of Material

# 3 Module Requirements

## 3.1 User View

The user has selected an AAS server and an AAS object he wants to display. When clicking the “Assetsubmodules” button all available submodules for the AAS object are shown with their names. If the AAS possesses the submodule “Bill of Material” the parts of an AAS are listed

## 3.2 Requirements

REQ/ Select submodules of AAS and retrieve submodule elements if the submodule’s name is “Bill of Material”

# 4 Analysis

The backend already processes the submodules for an AAS object and fills in the submodule list. For each submodule a request to the chosen server needs to be made to receive the kind of submodule as well as it’s elements.

# 5 Design

 When the User clicks on the submodule the button “Assetsubmodels” is available:Ein Bild, das Text, Screenshot, Schrift enthält.

Beschreibung automatisch generiert.

By clicking onto the button further information about the available submodels is displayed. If the submodel is the “Bill of Material” it’s elements are listed below:

Ein Bild, das Text, Screenshot, Schrift, Reihe enthält.

Beschreibung automatisch generiert.

# 6 Implementation

The function "getSubmodel(value)" (getSubmodel(value) in assetBody.js) in the assetBody receives the asset's submodelid and creates an URL witch consist of the chosen server url +"/submodels"+ the submodelid in base64.

Ein Bild, das Text, Screenshot, Schrift, Dokument enthält.

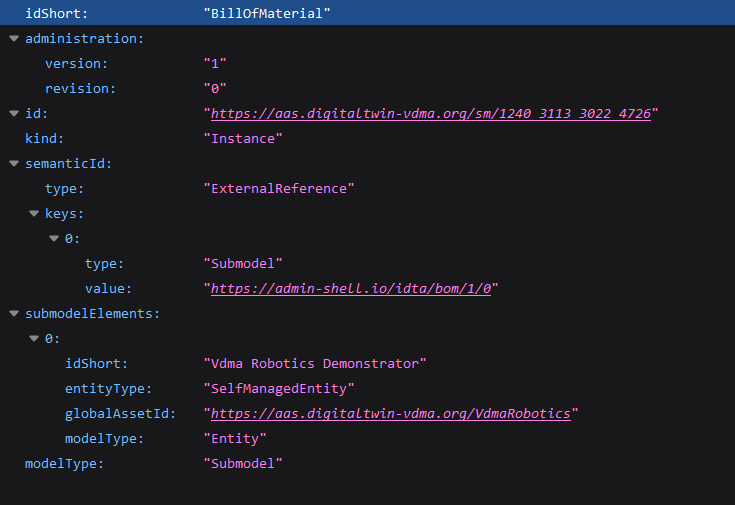
Automatisch generierte BeschreibungThe URL looks like this:



The passed value is the already displayed submodelId value from the HTML.





With a fetch the data is retrieved:

After that the HTML paragraph for the submodel is located and filled with the fitting idShort of the submodel.



If the idShort is a Bill of Material, for each submodel element a paragraph is created and filled with the elements idShort.

Ein Bild, das Text, Schrift, Screenshot enthält.

Automatisch generierte Beschreibung

# 7 Module Tests

Testing for this module is conducted through usability tests. In this process, a server and an AAS object are selected. Furthermore the “Assetsubmodels” button is pressed. The expected outcome is that the data from the selected server reaches the web client without causing a crash and then checks and displays the submodule information.