System Test Report

TINF21C, Software Engineering I

Practical project 2022/23

Project

AAS-Webclient

Customer

Markus Rentschler, Christian Holder

Rotebühlplatz 41, 70178 Stuttgart

Supplier

Project Leader: Samara Dominik (inf21001@lehre.dhbw-stuttgart.de)

Product Manager: Martin Rittmann (inf21157@lehre.dhbw-stuttgart.de)

System Architect: Marcel Hintze (inf21056@lehre.dhbw-stuttgart.de)

Test-Manager: Anja Niedermeier (inf21097@lehre.dhbw-stuttgart.de)

Developer: Severin Helms (21047@lehre.dhbw-stuttgart.de)

Technical Documentation: Tom Engelmann (inf21010@lehre.dhbw-stuttgart.de)

Version	Date	Author	Comment
0v1	04.05.2023	Tom Engelmann	Creation and basic setup
0v2	06.05.2023	Tom Engelmann	Layouting
1v0	10.05.2023	Tom Engelmann	Testing

Inhalt

Inhalt 2
1. Scope 3
2. Definitions 3
3. Test objects 3
4. Test cases 4
4.1. Test suite <TS-Server> 4
4.1.1. <TC-Server-Import-001> (Import server by URL) 4
4.1.2. <TC-Server-Switch-002> (Switch out servers) 5
4.1.3. <TC-Server-Load-003> (Display all assets of server) 6
4.2. Test suite <TS-Asset> 8
4.2.1. <TC-Asset-Browse-001> (Browse through different assets) 8
4.2.2. <TC-Asset-Search-002> (Search for asset by name) 9
4.2.3. <TC-Asset-Filter-003> (Filter for asset by manufacturer name) 12
4.2.4. <TC-Asset-Sort-004> (Sort assets by their manufacturing year) 13

1. Scope

The System Test Report (STR) records the outcomes of tests that originated from the System Test Plan (STP). It comprises the test cases outlined in the STP, along with their corresponding results obtained from carrying out the test plan.

2. Definitions

TC Test case

TS Test suite

STP System test plan

TD Test data

3. Test objects

The following test object will be examined:

Ref.-Id	Product Number	Product Name	Product Description
1	Build v1.0 (commit d5094d5)	AAS-Webclient	User friendly web-tool for displaying AAS-Servers and its contents while providing filter and search functionalities

4. Test cases

4.1. Test suite <TS-Server>

4.1.1. <TC-Server-Import-001> (Import server by URL)

Test case	
ID:	TC-SERVER-IMPORT-001
Name:	Import server by URL
Req.-ID:	REQ10, REQ20, REQ10, REQ40
Description:	The test case verifies the correct functionality of the import of an AAS-server by its URL. It verifies the display of an error message in case the given URL is incorrect. The test case uses the test data from table TD-001. The test set up consists of a computer, with an active internet connection and a working browser.

Test steps			
Step	**Action**	**Expected result**	**Actual result**
1	Click on the button “Server Menu” in the upper right corner	Dropdown server menu opens, providing an input field for a server URL as well as some predefined AAS-server URLs	The dropdown server menu is opening, it provides an input field for a server URL and four predefined AAS-server URLs
2	Click on one of the predefined Server URLs	The selected URL is displayed as the current server and its assets are loaded on the left side of the screen (names and preview images)	The current server is displayed after selection and its assets are shown on the left side with names and preview images
3	Click into the text input field for the Server URL and type in the URLs specified in TD-001	URLs that are marked with as “PASS” are loaded as the current server and their assets are loaded on the left side of the screen. URLs that are marked as “FAIL” produce an error message.	After input of the URLs listed in TD-001 the expected results were equivalent to the actual results

Test Data			
ID:			
Dataset	**URL**	**Expected result**	**Actual result**
1	https://ccae4836-001e-48c2-a4f9-235554f9400b.ma.bw-cloud-instance.org/	**PASS**	**PASS**
2	http://aas.murrelektronik.com:4001/aas/	**PASS**	**PASS**
3	https://www.dhbw-stuttgart.de/	**FAIL**	**FAIL**

Tester: Tom Engelmann
Date: 10.05.2023
Test case result: PASS

4.1.2. <TC-Server-Switch-002> (Switch out servers)

Test case	
ID:	TC-SERVER-SWITCH-002
Name:	Switch out servers
Req.-ID:	REQ10, REQ30
Description:	The test case verifies the correct functionality of switching out a currently used server with another one. It verifies the display of an error message in case the given URL is incorrect. The test case uses the predefined server-URLs that are already provided within the application itself as test data. The test set up consists of a computer, with an active internet connection and a working browser.

Test steps			
Step	**Action**	**Expected result**	**Actual result**
1	Click on the button “Server Menu” in the upper right corner	Dropdown server menu opens, providing an input field for a server URL as well as some predefined AAS-server URLs	The dropdown server menu is opening, it provides an input field for a server URL and four predefined AAS-server URLs
2	Click on one of the predefined Server-URLs	The selected URL is displayed as the current server and its assets are loaded on the left side of the screen (names and preview images)	The current server is displayed after selection and its assets are shown on the left side with names and preview images
3	Click on a different predefined Server-URL	The current server information in the top right corner switches to the newly selected server. The assets of the old server disappear from the left side of the window and only the assets of the new URL are displayed.	The new server is displayed after selection and its assets are shown on the left side with names and preview images. The old assets disappeared.

Tester: Tom Engelmann
Date: 10.05.2023
Test case result: PASS

4.1.3. <TC-Server-Load-003> (Display all assets of server)

Test case			
ID:	TC-SERVER-LOAD-003		
Name:	Display all assets of server		
Req.-ID:	REQ10, REQ30		
Description:	The test case verifies the correct functionality of displaying all assets which are contained in the currently used server. The test case uses the server-URL specified in TD-002 The test set up consists of a computer, with an active internet connection and a working browser.		
Test steps			
Step	**Action**	**Expected result**	**Actual result**
1	Click on the button “Server Menu” in the upper right corner	Dropdown server menu opens, providing an input field for a server URL as well as some predefined AAS-server URLs	The dropdown server menu is opening, it provides an input field for a server URL and four predefined AAS-server URLs
2	Click on the text input field for the server URL and type in the server-URL specified in TD-002. Then click on the “Add Server” button	The selected URL is displayed as the current server and its assets are loaded on the left side of the screen. (names and preview images) The exact same assets that are specified as expected results in TD-002 are displayed.	After input of the URL listed in TD-002 the expected result is equivalent to the actual result

Test Data			
ID:	TD-002		
Dataset	**URL**	**Expected result**	**Actual result**
1	https://ccae4836-001e-48c2-a4f9-235554f9400b.ma.bw-cloud-instance.org/	[REGISTRY, Norgren_B84G_4GK_AP3_RME, Norgren_ISOLine_PRA_802032_M_100, ARGO-HYTOS Filter Element EXAPOR MAX3, ARGO_HYTOS_Return_Filter_ES075, Parker_D1FPE50MB9NB0_ISDE8HU,	[REGISTRY, Norgren_B84G_4GK_AP3_RME, Norgren_ISOLine_PRA_802032_M_100, ARGO-HYTOS Filter Element EXAPOR MAX3, ARGO_HYTOS_Return_Filter_ES075, Parker_D1FPE50MB9NB0_ISDE8HU,

		Parker_HMI-2203250342054_HGDK8HU, Parker_PV046R2L1T1NMMC_JU64L8HZ, AAS_Type_CD55B20_50, AAS_Type_JSY205220, AAS_Demo_4WRPEH6, AAS_Demo_CytroPac, AAS_R412026837, AAS_R481712899, AAS_Type_SPAU-P10R-T-G18M-L-PNLK-PNVBA-M8D, AAS_Type_VUVS-L25-M52-AD-G14-F8-1C1, AAS_Type_DSBC-63-125-PPVA-N3, ExampleMotor]	Parker_HMI-2203250342054_HGDK8HU, Parker_PV046R2L1T1NMMC_JU64L8HZ, AAS_Type_CD55B20_50, AAS_Type_JSY205220, AAS_Demo_4WRPEH6, AAS_Demo_CytroPac, AAS_R412026837, AAS_R481712899, AAS_Type_SPAU-P10R-T-G18M-L-PNLK-PNVBA-M8D, AAS_Type_VUVS-L25-M52-AD-G14-F8-1C1, AAS_Type_DSBC-63-125-PPVA-N3, ExampleMotor] PASS
Tester: Tom Engelmann			
Date: 10.05.2023			
Test case result: PASS			

4.2. Test suite <TS-Asset>

4.2.1. <TC-Asset-Browse-001> (Browse through different assets)

Test case			
ID:	TC-ASSET-BROWSE-001		
Name:	Browse through different assets		
Req.-ID:	REQ40, REQ50		
Description:	The test case verifies the correct functionality of selecting one of the displayed assets to see more information about it. The test set up consists of a computer, with an active internet connection and a working browser. Additionally, the application must already have a working connection to an AAS-Server, meaning that its different assets are already loaded on to the left side of the screen.		
Test steps			
Step	**Action**	**Expected result**	**Actual result**
1	Move cursor to the left side of the screen onto the asset view and start scrolling up and down	The asset view scrolls up and down and reveals all assets that are available.	The preview of the assets does scroll up and down with the mouse wheel
2	Click on one of the assets	The space on the right side fills up with a more detailed view of the selected asset, providing an image (if available), its name, as well as information about all the sub models, which can be expanded and collapsed.	Detailed view with its information appeared. Image appeared. Expand and collapse functionality worked.
3	Click on a different asset	The information about the old asset is replaced by the detailed view about the new asset. All information is replaced.	The detailed view changed to the new asset. Information was updated
Tester: Tom Engelmann			
Date: 10.05.2023			
Test case result: PASS			

4.2.2. <TC-Asset-Search-002> (Search for asset by name)

Test case			
ID:	TC-ASSET-SEARCH-002		
Name:	Search for asset by name		
Req.-ID:	REQ80, REQ30		
Description:	The test case verifies the correct functionality of searching for specific assets by their name or by a substring of their name. For testing, the inputs and expected results from the TD-003 table are used. As not all possible search inputs can be tested, those are categorized by several equivalence classes that are supposed to cover and represent all major groups of input possibilities. The test set up consists of a computer, with an active internet connection and a working browser. Additionally, the application must already have a working connection to an AAS-Server, meaning that its different assets are already loaded on to the left side of the screen. In this case the server URL provided in TD-003 must be used to verify all the results.		
Test steps			
Step	**Action**	**Expected result**	**Actual result**
1	Click on the search text input field within the filter bar above the assets and start typing in a name.	Below the search field an autocomplete field appears with all assets that start with the same characters as the input.	Input into text input field worked. Autocomplete functionality worked
2	Select one of the autocomplete suggestions	All assets except the chosen one disappear.	All assets in the preview disappeared except for the chosen one
3	Click on the cross next to the search input field	The search field is empty and all assets are displayed again.	Text input field was cleared. All assets are shown again in the preview
4	Click on the search text input field again	/	
5	Type in the different search strings that are provided in TD-003 and click on the search icon	The assets that are specified in the expected results section of TD-003 for each input string are displayed. In case of an error, next to the search field, a error message appears.	After input of the URL listed in TD-003 the expected result is equivalent to the actual result

Test Data	

ID:	TD-003			
Dataset	**Equivalence class**	**Search strings**	**Expected result**	**Actual Result**
1	String at beginning of asset	“aas”	[AAS_Type_CD55B20_50, AAS_Type_JSY205220, AAS_Demo_4WRPEH6, AAS_Demo_CytroPac, AAS_R412026837, AAS_R481712899, AAS_Type_SPAU-P10R-T-G18M-L-PNLK-PNVBA-M8D, AAS_Type_VUVS-L25-M52-AD-G14-F8-1C1, AAS_Type_DSBC-63-125-PPVA-N3]	[AAS_Type_CD55B20_50, AAS_Type_JSY205220, AAS_Demo_4WRPEH6, AAS_Demo_CytroPac, AAS_R412026837, AAS_R481712899, AAS_Type_SPAU-P10R-T-G18M-L-PNLK-PNVBA-M8D, AAS_Type_VUVS-L25-M52-AD-G14-F8-1C1, AAS_Type_DSBC-63-125-PPVA-N3] PASS
2	Substring in the middle of asset	“filter”	[ARGO-HYTOS Filter Element EXAPOR MAX3, ARGO_HYTOS_Return_Filter_ES075]	[ARGO-HYTOS Filter Element EXAPOR MAX3, ARGO_HYTOS_Return_Filter_ES075] PASS
3	Numbers	“7”	[ARGO_HYTOS_Return_Filter_ES075, AAS_R412026837, AAS_R481712899]	[ARGO_HYTOS_Return_Filter_ES075, AAS_R412026837, AAS_R481712899] PASS
4	Special Characters	“-”	[ARGO-HYTOS Filter Element EXAPOR MAX3, Parker_HMI-2203250342054_HGDK8HU, AAS_Type_SPAU-P10R-T-G18M-L-PNLK-PNVBA-M8D, AAS_Type_VUVS-L25-M52-AD-G14-F8-1C1, AAS_Type_DSBC-63-125-PPVA-N3]	[ARGO-HYTOS Filter Element EXAPOR MAX3, Parker_HMI-2203250342054_HGDK8HU, AAS_Type_SPAU-P10R-T-G18M-L-PNLK-PNVBA-M8D, AAS_Type_VUVS-L25-M52-AD-G14-F8-1C1, AAS_Type_DSBC-63-125-PPVA-N3] PASS
5	Exact name of asset	“ARGO-HYTOS Filter Element EXAPOR MAX3”	[ARGO-HYTOS Filter Element EXAPOR MAX3]	[ARGO-HYTOS Filter Element EXAPOR MAX3] PASS
6	Search string with no results	“abcd”	[] + Error warning	[] + “No entries found”-warning

				Pass
7	Empty string	“”	*All assets* (compare TD-002, dataset 1, expected result)	All assets shown PASS
Tester: Tom Engelmann				
Date: 10.05.2023				
Test case result: PASS				

4.2.3. <TC-Asset-Filter-003> (Filter for asset by manufacturer name)

Test case			
ID:	TC-ASSET-FILTER-003		
Name:	Filter for asset by manufacturer name		
Req.-ID:	REQ70		
Description:	The test case verifies the correct functionality of filtering for specific assets by the name of their manufacturer. The test set up consists of a computer, with an active internet connection and a working browser. Additionally, the application must already have a working connection to an AAS-Server, meaning that its different assets are already loaded on to the left side of the screen.		
Test steps			
Step	**Action**	**Expected result**	**Actual result**
1	Click on the manufacturer button within the filter bar above the assets	A drop-down menu appears with all the manufacturers that can be found in the assets of the connected server.	A drop-down menu appeared listing all manufacturers and an “All”-option
2	Select one of the drop-down options	All assets are displayed that are from the selected manufacturer.	All assets were shown correctly from the chosen manufacturer. “All”-optionen worked as well.
3	Click on the cross next to the search input field	All assets are displayed again.	Filter was removed successfully.
Tester: Tom Engelmann			
Date: 10.05.2023			
Test case result: PASS			

4.2.4. <TC-Asset-Sort-004> (Sort assets by their manufacturing year)

Test case			
ID:	TC-ASSET-SORT-004		
Name:	Sort assets by their manufacturing year		
Req.-ID:	REQ60		
Description:	The test case verifies the correct functionality of sorting the assets by the year of their manufacturing. The test set up consists of a computer, with an active internet connection and a working browser. Additionally, the application must already have a working connection to an AAS-Server, meaning that its different assets are already loaded on to the left side of the screen.		
Test steps			
Step	**Action**	**Expected result**	**Actual result**
1	Click on the “Year” - button within the filter bar above the assets	A drop-down menu appears with the options “oldest first” and “newest first”	The drop-down menu appeared with the two mentioned options
2	Select the option “oldest first”	All assets are sorted from oldest to newest asset	The assets are sorted correctly beginning with the oldest asset
3	Select the option “newest first”	All assets are sorted from newest to oldest now	The assets are sorted correctly beginning with the newest
Tester: Tom Engelmann			
Date: 10.05.2023			
Test case result: PASS			