System Test Plan

(Systemtest Plan)

(TINF20C, SWE I Praxisprojekt 2021/2022)

Project: *Standalone Modelling Wizard for Devices*

Customer: *Rentschler & Holder*
Rotebühlplatz 41
70178 Stuttgart

Supplier: Team 1
Florian Kaiser, Florian Kellermann, Linus Eickhoff, Lukas Ernst, Malte Horst
Rotebühlplatz 41
70178 Stuttgart

Version	Date	Author	Comment
0.1	11.03.2022	Linus Eickhoff	created
0.2	22.04.2022	Malte Horst	Corrected errors
0.3	24.04.2022	Linus Eickhoff	Added missing data

Contents

1. Scope

The STP (System Test Plan) specifies the test strategy and test planning. It references tests to be performed to verify the accordance of the demanded features given by the SRS (System Requirements Specification) to the implemented features. The document derived from the STP is the STR (System Test Report) where additionally the results are given.

2. Definitions

TC Testcase
TS Testsuite
GUI Graphical User Interface

3. Product Names and Attributes

The following test objects must be verified:

Ref.-Id.	Product Number	Product Name	Product Description
1	Build v1.0	Standalone Modelling Wizard for Devices GUI	Windows standalone application with a GUI

4. Features

The following requirements must be verified, as long as they are not classified as “not to be tested”. This table shows the test coverage between functionality and test suites or test cases.

Req. - ID	Functionality	Priority	Testsuite ID
LF10: Import	Imports file by absolute path	A	TS-001
LF20: File validation	Checks whether input file is in a valid format	B	TS-001
LF30: Error handling	Application throws errors on expected shutdowns and wrong formatting	B	TS-002
LF40: GUI	Draws GUI for user	A	TS-002
LF50: Display device in a readable way	Displays loaded device in GUI in a readable way for user	A	TS-002
LF60: Edit device	Every attribute of devices should be editable	A	TS-002
LF70: Create device	Creates a new and empty device	A	TS-002
LF80: Export device	Loaded device is saved as to file	A	TS-001, TS-002

5. Test Preparation Strategy

The creation of tests will be application case based. Two main application cases can be identified, the file operations and the GUI.

File operations represent the first main application case. Device files need to be loaded, validated and saved to ensure full functionality of the application for the user.

The GUI is the second main application case. Unlike the previous plugin for the AML Editor, the GUI provides a view of the loaded device with input fields in which the respective device data is displayed. These fields must be checked and features to edit and save device must be validated.

6. Test Execution Strategy

Since it is a re-implementation of an already existing software, a complete test is not necessary, but it is still useful. The test should be divided into the following phases:

1) File operations
2) Graphical User Interface

Since the file operations are needed for the application to work, these have to be tested first.

Then the GUI functionality can be tested. This includes the start of the program and the execution of the main features of the application in the GUI.

7. Test Equipment

The following equipment must be available for testing:

- A computer with Windows 10 or higher
- The standalone Device Modelling Wizard software

8. Test Schedule and Budget

The testing of the application begins as soon as the application is completed. This makes it possible to make the necessary corrections quickly. The conversion library can only be tested once the rules for one input format, but preferably both input formats, have been established. Since only minimal changes are made in the GUI, the GUI can be tested as soon as all adjustments intended for the GUI have been made.

No budget is needed for the tests, as they are all performed by hand.

9. Test Planning

Testsuite	Test objective	Testplan Creator	Testplan Reviewer	Tester
TS-001	File operations	Linus Eickhoff	Florian Kaiser	Linus Eickhoff
TS-002	Graphical User Interface	Linus Eickhoff	Florian Kaiser	Linus Eickhoff

10. References / Standards

[1] SRS TINF20C Device Modelling Wizard

11. Appendix: Testcases

11.1. Testsuite <TS-001 File operations>

11.1.1. <TC-001-001> (Loading of a valid file with validation)

Testcase ID:	TC-001-001
Testcase Name:	Loading of a valid file with validation
Req.-ID:	LF10, LF20, LF30
Description:	The test case verifies that it recognizes if a valid file has been loaded.

Test Steps		
Step	**Action**	**Expected result**
1	Open Application.exe from Binary Folder	Application starts without problems.
2	Select a valid input file for the validation, by selecting “File” and then “Open” and choose file in explorer	The validation is executed successfully, and the conversion is completed correctly without error message.
3	Check if Data was interpreted correctly in “Attributes”, “Generic Information”, “Interfaces”	Should have all valid data in readable format.

Testdata:	TD-001-001			
Dataset	**File**	**Validation**	**Permission Input**	**Permission Output**
1	Balluff_ProductLibrary_CAEX3_221020.amlx	valid	given	given
2	Balluff-BNI_PNT-507-005-Z040-20201208.amlx	valid	given	given

11.1.2. <TC-001-002> (Loading of an invalid file with validation)

Testcase ID:	TC-001-002
Testcase Name:	Loading of an invalid file with validation
Req.-ID:	LF10, LF20, LF30
Description:	The test case verifies that errors are detected during the validation of the input file and a corresponding error message is displayed with a description of the error and line details in the log.

Test Steps

Step	Action	Expected result
1	Open Application.exe from Binary Folder	Application starts without problems.
2	Select an invalid input file for the validation, by selecting “File” and then “Open” and choose file in explorer	The validation is executed successfully, without crashing.
3	Check if error message is displayed to the user.	The Application displays the error when the file is invalid.

Testdata:	TD-001-002			
Dataset	**File**	**Validation**	**Permission Input**	**Permission Output**
1	Balluff_ProductLi-brary_CAEX3_221020.amlx (manipulated to be invalid)	invalid	given	given

11.1.3. <TC-001-003> (Export of a valid device to file with validation)

Testcase ID:	TC-001-003

Testcase Name:	Export of a valid device to file with validation
Req.-ID:	LF20, LF30, LF80
Description:	The test case verifies that a correctly formatted device can be validated and exported to a file

Test Steps		
Step	**Action**	**Expected result**
1	Open Application.exe from Binary Folder.	Application starts without problems.
2	Select a valid input file for editing, by selecting “File” and then “Open” and choose file in explorer.	The validation is executed successfully without crashing and the data is read and displayed correctly.
3	Edit the File, by changing its attributes and adding new data to empty attribute fields.	Attributes are changed correctly.
4	Click on “File” and select “Save”, select location in the file explorer and save file.	Valid file can be saved without errors and filename is generated automatically.
5	Open File again in Application and check if Changes were applied and file is still valid.	Changes like changes in Attributes are displayed correctly.

Testdata:	TD-001-003			
Dataset	**File**	**Validation**	**Permission Input**	**Permission Output**
1	Balluff_ProductLibrary_CAEX3_221020.amlx	valid	given	given

11.1.4. <TC-001-004> (Export of an invalid device to file with validation)

Testcase ID:	TC-001-001
Testcase Name:	Export of an invalid device with validation
Req.-ID:	LF10, LF20, LF30
Description:	The test case verifies that errors are detected during the validation of the exported device.

Test Steps

Step	Action	Expected result
1	Open Application.exe from Binary Folder.	Application starts without problems.
2	Select a valid input file for editing, by selecting “File” and then “Open” and choose file in explorer.	The validation is executed successfully, without crashing and the data is read and displayed correctly.
3	Edit the File, by changing its attributes with invalid data.	Invalid Inputs to attributes are recognized when exporting the device. Error message is displayed.

Testdata:	TD-001-004			
Dataset	**File**	**Validation**	**Permission Input**	**Permission Output**
1	Balluff-BNI_PNT-507-005-Z040-20201208.amlx	Valid (before editing)	given	given

11.2. Testsuite <TS-002 GUI>

11.2.1. <TC-002-001> (Add interface and attachment to device)

Testcase ID:	TC-002-001
Testcase Name:	Add interface and attachment to device
Req.-ID:	LF40, LF60
Description:	Run application and try if interfaces and attachments can be added to an device

Test Steps		
Step	**Action**	**Expected result**
1	Open Application.exe from Binary Folder.	Application starts without problems.
2	Drag Interface from "Interface Class Library" to "Interfaces" Window	The dragged Interface should be added to the device, indexed in order.
3	Add Attachment (e.g. "ComponentPicture") and click "select File"	File Explorer should open to select the right file.
4	Select File from explorer.	See if path is displayed in "Attachments" correctly.

Testdata:	TD-002-001				
Dataset	**Input File**	**Validation**	**Permission Input**	**Permission Output**	**Output File**

11.2.2. <TC-002-002> (GUI Load file via file explorer)

Testcase ID:	TC-002-002
Testcase Name:	GUI Input file selection via file explorer
Req.-ID:	LF10, LF20, LF40, LF50
Description:	The test case verifies that only the permitted file formats can be selected as input via file explorer. Afterwards, the device from the file has to be displayed correctly in the GUI. Permitted file formats: .xml

Test Steps		
Step	**Action**	**Expected result**
1	Open Application.exe from Binary Folder.	Application starts without problems.
2	Click on "File" and "Open"	The file explorer opens in a new window.

3	Search for file to load	A drop-down menu opens showing that only .amlx Files are allowed.
4	Double click on the file to select and load it.	File is loaded correctly if valid

Testdata:	TD-002-002				
Dataset	**Input File**	**Validation**	**Permission Input**	**Permission Output**	**Output File**
1	Balluff-BNI_PNT-507-005-Z040-20201208.amlx	valid	given	given	-

11.2.3. <TC-002-003> (GUI Creation and editing of a new device)

Testcase ID:	TC-002-003
Testcase Name:	GUI Creation and editing of a new device
Req.-ID:	LF40, LF60, LF70
Description:	The test case verifies whether a new, empty device can be created and edited in the editor.

Test Steps		
Step	**Action**	**Expected result**
1	Open Application.exe from Binary Folder.	Application starts without problems.
2	Go to “File” and select “new” to start creating new device.	All Fields are reset correctly.
3	Edit and fill attribute and interface data	All fields are edited correctly.

11.2.4. <TC-002-004> (GUI Export of a loaded device)

Testcase ID:	TC-002-004
Testcase Name:	GUI Export of a loaded device
Req.-ID:	LF40, LF80
Description:	The test case verifies whether a loaded device in the application can be exported and saved as a file.

Test Steps		
Step	**Action**	**Expected result**
1	Open Application.exe from Binary Folder.	Application starts without problems.
2	Select a valid input file for editing, by selecting “File” and then “Open” and choose file in explorer.	Device is loaded correctly and file validation is successful.
3	Change data, like attributes and click on “File” and then on “Save”.	File Explorer is opened and Filename autogenerated but editable.
4	Choose saving location for file and save.	File is saved and exported correctly without errors

Testdata:	TD-002-004	
Dataset	**Input File**	**Output File**
1	Balluff-BNI_PNT-507-005-Z040-20201208.amlx	Balluff-BNI_PNT-507-005-Z040-20201208.amlx
2	Balluff-BNI_PNT-508-105-Z015-CAEX3-20201207.amlx	Balluff-BNI_PNT-508-105-Z015-CAEX3-20201207.amlx