System Test Plan

(Systemtest Plan)

(TINF18C, SWE I Praxisprojekt 2019/2020)

Project: *DD2AML Converter*

Customer: *Rentschler & Ewertz*
Rotebühlplatz 41
70178 Stuttgart

Supplier: Team 3 – by Antonia Wermerkirch
(Nora Baitinger, Antonia Wermerskirch, ~~Carl Beese~~, Lara Mack, Bastiane Storz)
Rotebühlplatz 41
70178 Stuttgart

Version	Date	Author	Comment
0.1	07.09.2018		created
0.2	03.04.2020	Wermerskirch	First draft
0.3	15.04.2020	Wermerskirch	Second draft
0.4	16.04.2020	Wermerskirch	Changes after Review Updated Features and Test Equipment
0.5	04.05.2020	Wermerskirch	Added Test Cases
1.0	09.05.2020	Wermerskirch	Finalization

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
MC	Multicast
CLI	Command Line Interface
GUI	Graphical User Interfac

3. Product Names and Attributes

The following test objects must be verified:

Ref.-Id.	Product Number	Product Name	Product Description
1	Build v1.0	DD2AML Conversion Library	Library which is responsible for the conversion to AML
2	Build v1.0	DD2AML CLI and GUI	Command Line Interface and Graphical User Interface for conversion

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: Format analysis	Checks which file format was selected	A	TS-001: Conversion Library
LF20: Input validation	Checks whether input file can be serialized	A	TS-001: Conversion Library
LF30: Conversion	– Converting an IODD file to AML – Converting a CSP+ file to AML	A A	TS-001: Conversion Library
LF40: AML Packager	Creates the AMLX package	B	TS-001: Conversion Library

LF50: File conversion library	– Converts a file to an AML string – Converts a file to an AMLX package	B A	TS-001: Conversion Library
LF60: File conversion command line	Conversion using a CLI	A	TS-002: CLI
LF70: File conversion GUI	Conversion using a GUI	A	TS-003: GUI
LF80: Error handling	Provides understandable error messages to the user	B	TS-001: Conversion Library

5. Test Preparation Strategy

The creation of tests will be application case-based. Three main application cases can be identified, the conversion library, the CLI, and the GUI.

The Conversion Library contains the main functions of the converter. The correct conversion of an IODD file and a CSP+ file must be tested. The conversion can only be triggered by using the provided interfaces. The functionality of the library is therefore tested particularly using the CLI, which the library uses for the conversion.

The CLI is another main application case. It is used to pass arguments like the file path of the input file and optionally the output path to the application to start the conversion. As there are several flags that can be passed to the command line tool, the equivalence class method will be used to reduce the number of tests.

The GUI is the last main application case. Unlike the CLI, the Gui provides input fields in which the respective file path is inserted. These fields must be checked.

6. Test Execution Strategy

Since it is a further development 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) Command Line Interface
2) Conversion Library
3) Graphical User Interface

Since the CLI is needed for the conversion so that the conversion library can be tested and used, the CLI is tested first.

To verify that the conversion of an IODD or CSP+ file to an AML file works correctly, which is the main function of the program, it is tested next.

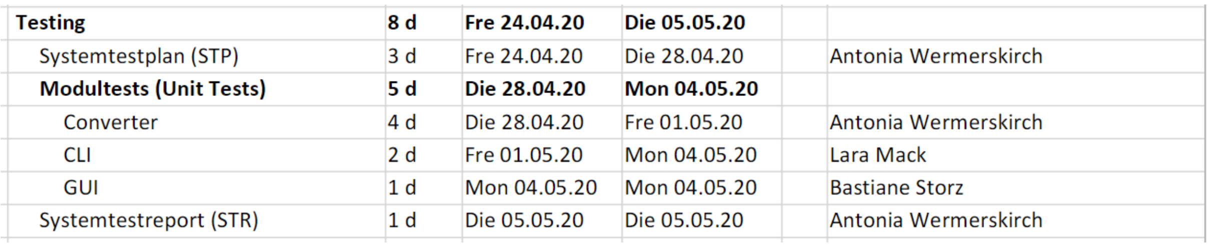
In the end, the GUI is tested.

7. Test Equipment

The following equipment must be available for testing:

- A computer with Windows 7 or higher
- Installed AutomationML Editor (Download here)
- The DD2AML software

8. Test Schedule and Budget



Testing	**8 d**	**Fre 24.04.20**	**Die 05.05.20**		
Systemtestplan (STP)	3 d	Fre 24.04.20	Die 28.04.20		Antonia Wermerskirch
Modultests (Unit Tests)	**5 d**	**Die 28.04.20**	**Mon 04.05.20**		
Converter	4 d	Die 28.04.20	Fre 01.05.20		Antonia Wermerskirch
CLI	2 d	Fre 01.05.20	Mon 04.05.20		Lara Mack
GUI	1 d	Mon 04.05.20	Mon 04.05.20		Bastiane Storz
Systemtestreport (STR)	1 d	Die 05.05.20	Die 05.05.20		Antonia Wermerskirch

The testing of the CLI begins as soon as the CLI 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 installer of 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	Conversion Library	Antonia Wermerskirch	Bastiane Storz	Antonia Wermerskirch
TS-002	Command Line Interface	Antonia Wermerskirch	Bastiane Storz	Antonia Wermerskirch
TS-003	Graphical User Interface	Antonia Wermerskirch	Bastiane Storz	Antonia Wermerskirch

10. References / Standards

[1] SRS TINF18C DD2AML

11. Appendix: Testcases

11.1. Testsuite <TS-001 Conversion Library>

11.1.1. <TC-001-001> (File Validation with valid input file)

Testcase ID:	TC-001-001	
Testcase Name:	File Validation with valid Input file	
Req.-ID:	LF20, LF10, LF30, LF40, LF50	
Description:	The test case verifies that it recognizes if a valid file has been selected.	
Test Steps		
Step	**Action**	**Expected result**
1	Install the DD2AML tool and open the CLI by typing cmd in the windows search.	The DD2AML tool is installed on the system. The CLI is open.
2	Select a valid input file for the validation, for example: dd2aml –input /filePathTo/Balluff-BNI_IOL_355_S02_Z013-20170315-IODD1.1.xml	The validation is executed successfully, and the conversion is completed correctly without error message.
3	Then open the logs of the CLI. These can be found under: C:\Users\USERNAME\AppData\Local\DD2AML\Logs\CLI	After replacing the USERNAME tag with the real username, the CLI folder with all logs opens. The most recent log is opened.
4	Find the log message that shows that the file was successfully deserialized. It can be found at the beginning of the log file.	The log message “DD file was deserialized correctly.” should be found approximately in the fourth line of the log.

Testdata:	TD-001-001			
Dataset	**File**	**Validation**	**Permission Input**	**Permission Output**
1	Balluff-BNI_IOL_355_S02_Z013-20170315-IODD1.1.xml	valid	given	given

11.1.2. <TC-001-001> (File Validation with invalid input file)

Testcase ID:	TC-001-001	
Testcase Name:	File Validation with invalid input file	
Req.-ID:	LF20, LF80	
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	Install the DD2AML tool and open the CLI by typing cmd in the windows search.	The DD2AML tool is installed on the system. The CLI is open.
2	Select a valid input file for the validation, for example: dd2aml –input /filePathTo/BrokenBalluff-BNI_IOL_355_S02_Z013-20170315-IODD1.1.xml	The conversion is aborted after the failed validation.
3	Then open the logs of the CLI. These can be found under: C:\Users\USERNAME\AppData\Local\DD2AML\Logs\CLI	After replacing the USERNAME tag with the real username, the CLI folder with all logs opens. The most recent log is open.
4	Look at the first error message in the logs.	The error message can be found approximately in the 5th line. Detailed information about the error, as well as line details are given.

Testdata:	TD-001-002			
Dataset	**File**	**Validation**	**Permission Input**	**Permission Output**
1	BrokenBalluf-BNI_IOL_355_S02_Z013-20170315-IODD1.1.xml	invalid	given	given

11.2. Testsuite <TS-001 Conversion Library>

11.2.1. <TC-002-001> (View CLI help text)

Testcase ID:	TC-001-002
Testcase Name:	View CLI help text
Req.-ID:	LF60
Description:	The test case verifies that the converter displays all possible functions as help text as soon as the passed argument contains "--help". All other passed arguments are ignored.

Test Steps		
Step	**Action**	**Expected result**
1	Install the DD2AML tool and open the CLI by typing cmd in the windows search.	The DD2AML tool is installed on the system. The CLI is open.
2	Run the DD2AML CLI with valid arguments and use the help flag.	Regardless of the other valid arguments, only "--help" is executed and a help text is displayed, showing all possible functions.

Testdata:	TD-002-002			
Dataset	**File**	**Validation**	**Permission Input**	**Permission Output**
1	Balluff-BNI_IOL_355_S02_Z013-20170315-IODD1.1.xml	valid	given	given
2	0x1099_BNI CIE-508-105-Z015_2.0_en.cspp	valid	given	given
3	GSDMLV2.33.xml	valid	given	given

11.2.2. <TC-002-002> (Converting without output flag)

Testcase ID:	TC-001-002	
Testcase Name:	Converting without output flag	
Req.-ID:	LF60	
Description:	The test case verifies that a conversion is also possible without a given output path.	
Test Steps		
Step	**Action**	**Expected result**
1	Install the DD2AML tool and open the CLI by typing cmd in the windows search.	The DD2AML tool is installed on the system. The CLI is open.
2	Run the DD2AML CLI with valid input flag, for example: dd2aml –input /filePathTo/Balluff-BNI_IOL_355_S02_Z013-20170315-IODD1.1.xml	The conversion is executed successfully. Since no output path is given, the output file is saved in the file path of the input file.

Testdata:	TD-002-002			
Dataset	**File**	**Validation**	**Permission Input**	**Permission Output**
1	Balluff-BNI_IOL_355_S02_Z013-20170315-IODD1.1.xml	valid	given	given
2	0x1099_BNI CIE-508-105-Z015_2.0_en.cspp	valid	given	given
3	GSDMLV2.33.xml	valid	given	given

11.3. Testsuite <TS-003 GUI>

11.3.1. <TC-003-001> (GUI Input field verification)

Testcase ID:	TC-003-001	
Testcase Name:	GUI Input field verification	
Req.-ID:	LF70	
Description:	Run converter via a graphical user interface with an empty Input field. The test case ensures that a conversion is not possible without an input file.	
	Test Steps	
Step	**Action**	**Expected result**
1	Install the DD2AML Software and open the GUI.	The software is installed and the GUI window opens.
2	Try to start the conversion by pressing the “Convert” button at the bottom centre.	Conversion not possible, because "Convert" button stays deactivated.

Testdata:	TD-003-001				
Dataset	**Input File**	**Validation**	**Permission Input**	**Permission Output**	**Output File**

11.3.2. <TC-003-002> (GUI Input file selection via file explorer)

Testcase ID:	TC-003-002	
Testcase Name:	GUI Input file selection via file explorer	
Req.-ID:	LF70	
Description:	The test case verifies that only the permitted file formats can be selected as input via file explorer. Permitted file formats: .xml and .cspp	
	Test Steps	
Step	**Action**	**Expected result**
1	Install the DD2AML Software and open the GUI.	The software is installed and the GUI window opens.
2	Click on the "..." button at the end of the input text field.	The file explorer opens in a new window.
3	Click on "Files" in the lower right corner directly above the buttons for open and cancel	A drop-down menu opens showing that only file suffix with .xml or .cspp are allowed.

Testdata:	TD-003-002				
Dataset	**Input File**	**Validation**	**Permission Input**	**Permission Output**	**Output File**

11.3.3. <TC-003-003> (GUI Input file selection via drag and drop)

Testcase ID:	TC-003-003	
Testcase Name:	GUI Input file selection via drag and drop	
Req.-ID:	LF70	
Description:	The test case verifies that only the permitted file formats can be selected as input via drag and drop Permitted file formats: .xml and .cspp	
Test Steps		
Step	**Action**	**Expected result**
1	Install the DD2AML Software and open the GUI.	The software is installed and the GUI window opens.
2	Open the file explorer and select any file. Drag the selected file and drop it into the GUI input text field.	If the selected file has a valid file suffix, its absolute file path will appear in the input field. If it has an invalid suffix, it is not possible to drop the file into the input field.

Testdata:	TD-003-003				
Dataset	**Input File**	**Validation**	**Permission Input**	**Permission Output**	**Output File**

11.3.4. <TC-003-004> (GUI Output file path generation)

Testcase ID:	TC-003-004	
Testcase Name:	GUI Output file path generation	
Req.-ID:	LF70	
Description:	The test case verifies whether an output file is automatically suggested for a given input. This output file should no longer have the file format of the input file in its name.	
Test Steps		
Step	**Action**	**Expected result**
1	Install the DD2AML Software and open the GUI.	The software is installed and the GUI window opens.

2	Select a valid file of IODD, CSP+ or GSD format in the Input text box.	As soon as the file including file path is in the input field, an output file is suggested for the same directory. The output file has the suffix .amlx and does not have the file format of the input file in its name.

Testdata:	TD-003-004	
Dataset	**Input File**	**Output File**
1	.\Balluff-BNI_IOL_355_S02_Z013-20170315-IODD1.1.xml	.\Balluff-BNI_IOL_355_S02_Z013-20170315.amlx
2	. \0x1099_BNI CIE-508-105-Z015_2.0_en.cspp	.\0x1099_BNI CIE-508-105-Z015_2.0_en.amlx
4	.\GSDML-V2.33.xml	.\V2.33.amlx