System Test Plan

AAS Digital Nameplate Generator

Customer:	Rentschler & Holder
Company address:	Rotebühlplatz 41, 70178 Stuttgart
Supplier:	Team 2

Role	Name	Email Address
Team Lead	Adrian Khairi	Inf21196@lehre.dhbw-stuttgart.de
Test Manager	Janin Ahlemeyer	Inf21006@lehre.dhbw-stuttgart.de
System Architect & Software Developer	Mika Kuge	Inf21059@lehre.dhbw-stuttgart.de
Technical Documentation	Maris Koch	Inf21050 @lehre.dhbw-stuttgart.de
Product Manager	Erika Zhang	Inf21174@lehre.dhbw-stuttgart.de

Version Control

Version	Date	Author	Comment
1.0	26.03.2023	Janin Ahlemeyer	Initialized the STP and created a first version
1.1	31.03.2023	Janin Ahlemeyer	Refined the document with Erika Zhang's comments
1.2	05.04.2023	Janin Ahlemeyer	Improved the document's quality even more by using inspiration of actual enterprise documents
1.3	15.04.2023	Janin Ahlemeyer	Refined the document with Adrian Khairi's comments
1.4	16.04.2023	Janin Ahlemeyer	Adjusting test data in cooperation with Mika Kuge
1.5	23.04.2023	Janin Ahlemeyer	Refined the document with Erika Zhang's comments
2.0	10.05.2023	Janin Ahlemeyer	Made final improvements

Table of Contents

List of figures

1. Introduction

The System Test Plan (STP) of the project “AAS Digital Nameplate Generator” is designed to stipulate the scope, approach, resources, and schedule of all testing activities. Hence, its purpose is to identify the tasks and activities to be performed so that all aspects of the system are adequately tested, and the quality of the software is ensured.

2. Scope

2.1. Quality Objective

The objective of this document is to verify the “AAS Digital Nameplate Generator’s” Functionality. Testing the correct view of assets of an arbitrary “Asset Administration Shell” is of particular significance. Moreover, the responsive and compatible GUI with different environments is needed to ensure a user-friendly front-end application. Besides, the installation and performance of the application is substantiated as well.

Therefore, the testing should focus on guaranteeing a normal user experience from installing as well as searching an asset to downloading a nameplate on different browsers and hardware.

2.2. In Scope

All the functional requirements of the nameplate generator which are defined in the Software Requirements Specification (SRS) of the project need to be tested. The priority of requirements is scaled from 0 to 5. While 5 represents necessary requirements, issues with 0 may be neglected.

Requirement ID	Functionality	Priority	Testsuite ID or TCS Link
DNG.GUI.001	Responsive and compatible GUI	2	TC.CON.001.F TC.CON.002.F TC.NAV.001.F TC.NAV.002.F TC.OV.001.F TC.NP.001.F TC.NP.002.F
DNG.GUI.002	Download menu for SVG and PNG format	4	TC.NP.001.F TC.NP.002.F
DNG.GUI.003	Search functionality	3	TC.CON.001.F TC.CON.002.F

			TC.PERF.001.F
DNG.GUI.004	Navigation buttons	1	TC.NAV.001.F TC.NAV.002.F
DNG.GUI.005	QR-code generator	5	TC.NP.001.F TC.NP.002.F
DNG.GUI.006	Nameplate generator	5	TC.NP.001.F TC.NP.002.F
DNG.GUI.007	Error handling	4	TC.INST.001.F TC.CON.001.F TC.CON.002.F TC.OV.001.F TC.NP.001.F TC.NP.002.F TC.PERF.001.F
DNG.PERF.001	Performance	3	TC.PERF.001.F

2.3. Out of Scope

However, due to the lack of time available for the project, this STP will focus on integration testing procedures. Following features are not tested since they are not included in the functional requirements.

- Application programming interface (API)
- Security
- Usable and easy to understand for its users

3. Reference Material

The sources cited below are utilized as a reference for this document.

- TINF21C_SRS_Team_2_2v0
- IEEE Standard 829-1998, Standard for Software Test Documentation
- Testfallentwurfsverfahren | SpringerLink

4. Definitions and Acronyms

STP	System Test Plan
SRS	Software Requirements Specification
DNG	Digital Nameplate Generator
TS	Test Suite
TC	Testcase

AAS	Asset Administration Shell
STR	System Test Report
API	Application programming interface

5. Document Maintenance

This document will be reviewed and updated as needed, as the project proceeds through each phase of the system development life cycle.

This document contains a revision history log. When changes occur, the document’s revision history log will reflect an updated version number as well as the date, the owner making the change, and change description will be recorded in the revision history log of the document.

6. Participants Roles and Responsibilities in Test and Evaluation

In charge of Test and Evaluation is the Test Manager. First, they are responsible for overseeing the Test and Evaluation process, including creating test plans, execution, review, and coordinating acceptance. They may assign tasks to the developer team to assist the testing.

Furthermore, the document is created after the development phase. Its intended audience is the Product manager and Test Manager. Some portions of the document may on occasion be shared with the customer whose input and approval of the testing process is needed.

7. Approach

In this project, requirements-based testing should be conducted. Test cases are designed based on test objectives as well as conditions derived from the SRS.

The requirements-based method used in this document for creating test cases is black box testing. It is a method of software testing that is used to evaluate the functionality of a system or application without any knowledge of its internal workings. Instead, the tester interacts with the system through its user interface, inputs various test cases, and observes the output to ensure that the system behaves as expected. Additionally, since black box testing focuses on the system's behavior, it may not be suitable for testing non-functional requirements such as performance, scalability, and security.

There are several techniques that can be used in black box testing, including equivalence partitioning, boundary value analysis, and pairwise testing. Equivalence partitioning involves dividing the input space into groups of equivalent inputs, while boundary value analysis focuses on testing the boundaries of the input space. The pairwise testing approach involves selecting pairs of input parameters or factors that are most likely to interact with each other and create defects. To reduce decision tables the open-source tool ALLPAIRS by James Bach is used. These three methods are used in conjunction to ensure thorough testing of the system.

8. Test Data

Test data:	TD.001				
Dataset	Device	IP address	Default gateway	Net mask	Expected result
01	Windows	192.168.0.254	192.168.0.1	192.168.0.1	PASS
02	Mac	192.168.0.254	192.168.0.1	192.168.0.1	PASS
03	IPhone	192.168.0.254	192.168.0.1	192.168.0.1	PASS
04	Windows	192.168.0.254	192.168.0.0	192.168.0.1	FAIL
05	Mac	192.168.0.254	192.168.0.0	192.168.0.1	FAIL
06	IPhone	192.168.0.254	192.168.0.0	192.168.0.1	FAIL

Test data:	TD.002			
Dataset	Browser	Screen size	Operating system	Expected result
01	Google Chrome 111.0	1920 x 1080	Windows 10	PASS
02	Google Chrome 111.0	3840 x 2160	Windows 10	PASS
03	Google Chrome 111.0	3088 x 1440	macOS	PASS
04	Microsoft Edge 110.0	1920 x 1080	Windows 10	PASS
05	Microsoft Edge 110.0	3840 x 2160	Windows 10	PASS
06	Microsoft Edge 110.0	3088 x 1440	macOS	PASS
07	Firefox 111.0	1920 x 1080	macOS	PASS
08	Firefox 111.0	3840 x 2160	macOS	PASS
09	Firefox 111.0	3088 x 1440	Windows 10	PASS
10	Safari 16.1	1920 x 1080	macOS	PASS
11	Safari 16.1	2340 x 1080	iOS	PASS

Test data:	TD.003

Dataset	AAS server link	API	Expected result
01	https://admin-shell-io.com/5001	V1 API	PASS
02	https://ccae4836-001e-48c2-a4f9-235554f9400b.ma.bw-cloud-instance.org/	V3 API	PASS
03	http://aas.murrelektronik.com:4001/aas	V3 API	PASS
04	ccae4836-001e-48c2-a4f9-235554f9400b.ma.bw-cloud-instance.org/	V3 API	FAIL

The test server can be accessed with the link *https://ccae4836-001e-48c2-a4f9-235554f9400b.ma.bw-cloud-instance.org/*.

Test data	TD.004			
Dataset	Asset	AAS server	Product image	Expected result
01	Norgren_B84G_4GK_AP3_RME	Test server	Width: 400px Height: 600px	PASS
02	TestAsset_PINone_MI_None	Test server	None	PASS
03	TestAsset_PIBroken_MI_Broken	Test server	Broken	PASS
04	TestAsset_PiSmaller	Test server	Smaller than max size	PASS
05	TestAsset_PiBigger	Test server	Bigger than max size	PASS
06	TestAsset_Many	Test server	More than one picture	PASS

Test data:	TD.005			
Dataset	Asset	Server	Marking image	Expected result
01	Norgren_B84G_4GK_AP3_RME	Test server	None	PASS
02	AAS_Type_VUVS-L25-M52-AD-G14-F8-1C1	Test server	One	PASS
03	AAS_Demo_CytroPac	Test server	Many	PASS

Test data:	TD.006			
Dataset	Asset	Server	Nameplate	Expected result
01	Norgren_B84G_4GK_AP3_RME	Test server	Adequate	PASS
02	ExampleMotor	Test server	None	PASS

9. Test Items

9.1. Testsuite <TS.INST: Access via Link>

Test case		
ID:	<TC.INST.001.F>	
Name:	Access front end on local machine	
Req. -ID:	DNG.GUI.007	
Description:	*The test case verifies the correct functionality of the hyperlink of the web application as well as the device's configuration and internet connection. It verifies the forwarding from the project's GitHub page. The test case verifies compatibility with relevant browsers, screen sizes and operating systems.* *The test case uses the test data from table TD.001 and TD.002.* *The test set up consists of a computer, which is connected to the internet.*	
Test steps		
Step	Action	Expected result
1	Go to project's GitHub	Display project's GitHub page
2	Click on the Link for the Digital Nameplate Generator in the ReadMe file	Display home page; no connection to the Internet or a misconfigured network lead to no access

9.2. Testsuite <TS.CON: Connect to AAS-Server>

Test case	
ID:	<TC.CON.001.F>
Name:	Connect to AAS-Server via search bar
Req. -ID:	DNG.GUI.001; DNG.GUI.003; DNG.GUI.007
Description:	*The test case verifies the correct functionality of the search bar. It verifies the connection to AAS-Servers. Links missing https:\\*

	are not supported. The test case verifies compatibility with relevant browsers, screen sizes and operating systems. *The test case uses the test data from table TD.002 and TD.003.* *The test set up consists of a computer, which is connected to the internet.*	
Test steps		
Step	Action	Expected result
1	Go to home page	Display home page
2	Click on search bar	Search bar is selected. A drop-down list of search suggestions is opened.
3	Enter a string into search bar	For each character the suggestion of the drop-down list is filtered.
4	Click on search button	All assets from the server are loaded. The asset list is displayed; missing images are replaced with default image; invalid parameters produce the display of an error message.

Test case		
ID:	<TC.CON.002.F>	
Name:	Connect to suggested AAS-Server via search bar	
Req. -ID:	DNG.GUI.001; DNG.GUI.003; DNG.GUI.007	
Description:	*The test case verifies the correct functionality of the search bar and its suggestions. It verifies the connection to AAS-Servers. The test case verifies compatibility with relevant browsers, screen sizes and operating systems.* *The test case uses the test data from table TD.002 and TD.003.* *The test set up consists of a computer, which is connected to the internet.*	
Test steps		
Step	Action	Expected result
1	Go to home page	Display home page
2	Click on search bar	Search bar is selected. A drop-down list of search suggestions is opened.
3	Enter a string into search bar	For each character the suggestion of the drop-down list is filtered.
4	Click on one of the suggested server addresses	All assets from the server are loaded. The asset list is displayed; missing

		images are replaced with default image; invalid parameters produce the display of an error message.

9.3. Testsuite <TS.NAV: Navigation>

Test case		
ID:	<TC.NAV.001.F>	
Name:	Navigation bar	
Req. -ID:	DNG.GUI.001; DNG.GUI.004	
Description:	*The test case verifies the correct forwarding of the navigation bar. It verifies compatibility with relevant browsers, screen sizes and operating systems.* *The test case uses test data from table TD.002.* *The test set up consists of a computer, which is connected to the internet.*	
Test steps		
Step	Action	Expected result
1	Go to home page	Display home page
2	Click on menu button	Open menu
3	Click on “About” row	Forwarding to about page
4	Click on menu button	Open menu
5	Click on “Home” row	Forwarding to home page
6	Click on menu button	Open menu
7	Click on “GitHub” row	Forwarding to project’s GitHub page

Test case		
ID:	<TC.NAV.002.F>	
Name:	Browser’s navigation buttons	
Req. -ID:	DNG.GUI.001; DNG.GUI.004	
Description:	*The test case verifies the correct functionality of the browser’s navigation buttons. It verifies compatibility with relevant browsers, screen sizes and operating systems.* *The test case uses test data from table TD.002.* *The test set up consists of a computer, which is connected to the internet.*	
Test steps		
Step	Action	Expected result
1	Go to an asset view	Display asset view

2	Click on back button	Back to asset list
3	Click on back button	Back to home page
4	Click on “about”	Display about page
5	Click on back button	Back to home page

9.4. Testsuite <TS.OV: Overview>

Test case		
ID:	<TC.OV.001.F>	
Name:	Choose asset of asset list	
Req. -ID:	DNG.GUI.001; DNG.GUI.007	
Description:	*The test case verifies the correct functionality of the asset list. It verifies that the images of the product and markings are adjusted.* *The test case verifies compatibility with relevant browsers, screen sizes and operating systems.* *The test case uses the test data from table TD.002, TD.004, TD.005 and TD.006.* *The test set up consists of a computer, which is connected to the internet.*	
Test steps		
Step	Action	Expected result
1	Go to asset list	Display asset list
3	Click on search bar with search box text “Asset Name”	Search bar is selected
	Enter a string into search bar	For each character the assets are filtered matching the string.
2	Hover on a table row	Currently selected asset appears in another color.
3	Click on table row of specific asset	The asset view of the specific asset is opened. Product image, nameplate and asset details are displayed; missing submodel details produce a corrupted overview
4	Click on collapse item	Further details of category are expanded

9.5. Testsuite <TS.NP: Nameplate>

Test case

<table>
<tr><td>ID:</td><td colspan="2">&lt;TC.NP.001.F&gt;</td></tr>
<tr><td>Name:</td><td colspan="2">Download nameplate in SVG</td></tr>
<tr><td>Req. -ID:</td><td colspan="2">DNG.GUI.001; DNG.GUI.002; DNG.GUI.005; DNG.GUI.006; DNG.GUI.007</td></tr>
<tr><td>Description:</td><td colspan="2"><i>The test case verifies the correct functionality of downloading a nameplate in SVG format and scanning its QR code with a smartphone. It verifies the decryption of the complete nameplate submodel details as a string. The test case verifies compatibility with relevant browsers, screen sizes and operating systems.
The test case uses the test data from table TD.002 and TD.006.
The test set up consists of a smartphone with scanning function and a computer, which is connected to the internet.</i></td></tr>
<tr><td colspan="3">Test steps</td></tr>
<tr><td>Step</td><td>Action</td><td>Expected result</td></tr>
<tr><td>1</td><td>Go to asset view</td><td>Asset view is displayed</td></tr>
<tr><td>2</td><td>Click on “Download SVG” button</td><td>Nameplate in SVG format is downloaded</td></tr>
<tr><td>3</td><td>Open the file</td><td>Nameplate in SVG format is displayed</td></tr>
<tr><td>4</td><td>Open scan application on device</td><td>Camera to scan a QR-Code is opened</td></tr>
<tr><td>5</td><td>Scan QR code</td><td>Response with a string containing all nameplate submodel information is displayed; failed scanning produces the display of an error message.</td></tr>
<tr><td colspan="3"></td></tr>
</table>

<table>
<tr><td colspan="3">Test case</td></tr>
<tr><td>ID:</td><td colspan="2">&lt;TC.NP.002.F&gt;</td></tr>
<tr><td>Name:</td><td colspan="2">Download nameplate in PNG</td></tr>
<tr><td>Req. -ID:</td><td colspan="2">DNG.GUI.001; DNG.GUI.002; DNG.GUI.005; DNG.GUI.006; DNG.GUI.007</td></tr>
<tr><td>Description:</td><td colspan="2"><i>The test case verifies the correct functionality of downloading a nameplate inf PNG format and scanning its QR code with a smartphone. It verifies the decryption to the complete nameplate submodel details as a string. The test case verifies compatibility with relevant browsers, screen sizes and operating systems.
The test case uses the test data from table TD.002 and TD.006.
The test set up consists of a smartphone with scanning function and a computer, which is connected to the internet.</i></td></tr>
<tr><td colspan="3">Test steps</td></tr>
<tr><td>Step</td><td>Action</td><td>Expected result</td></tr>
</table>

1	Go to asset view	Asset view is displayed
2	Click on “Download PNG” button	Nameplate in PNG format is downloaded
3	Open the file	Nameplate in PNG format is displayed
4	Open scan application on device	Camera to scan a QR-Code is opened
5	Scan QR code	Response with a string containing all nameplate submodel information is displayed; failed scanning produces the display of an error message.

9.6. Testsuite <TS.PERF: Performance>

Test case		
ID:	<TC.PERF.001.F>	
Name:	Performant under expected or peak loads	
Req. -ID:	DNG.GUI.003; DNG.GUI.007; DNG.PERF.001	
Description:	*The test case verifies fast website loads below seven seconds. It verifies the time fetching data from the server and displaying it.* *The test case uses the test data from table TD.003.* *The test set up consists of a timer and a computer, which is connected to the internet.*	
Test steps		
Step	Action	Expected result
1	Go to home page	Display home page
2	Click on search bar	Search bar is selected. A drop-down list of search suggestions is opened.
3	Enter a string into search bar	For each character the suggestion of the drop-down list is filtered.
4	Click on search button	All assets from the server are loaded. The asset list is displayed; missing images are replaced with default image; invalid parameters produce the display of an error message.
5	Click on one of the suggested server addresses and measure time	All assets from the selected server are loaded below seven seconds. The asset list is displayed; invalid parameters produce the display of an error message.

10. Requirements Traceability Matrix

Requirement Identifiers	Reqs Tested	DNG.GUI.001	DNG.GUI.002	DNG.GUI.003	DNG.GUI.004	DNG.GUI.005	DNG.GUI.006	DNG.GUI.007	DNG.PERF.001
Test Cases	26	7	2	3	2	2	2	7	1
TC.INST.001.F	1							X	
TC.CON.001.F	3	X		X				X	
TC.CON.002.F	3	X		X				X	
TC.NAV.001.F	2	X			X				
TC.NAV.002.F	2	X			X				
TC.OV.001.F	2	X						X	
TC.NP.001.F	5	X	X			X	X	X	
TC.NP.002.F	5	X	X			X	X	X	
TC.PERF.001.F	3			X				X	X

Figure 10.1: Requirements Traceability Matrix

11. Incident Management

GitHub will be used as an incident tracking log. Issues will be used to capture the details of each incident. In particular, the error message and the possible cause are described. Furthermore, these issues receive Tags, which indicated the priorities. These will be determined as follows.

- **Critical**: the application isn’t available or doesn’t work, preventing any further testing
- **Major**: the product does not function as expected, or the results don’t meet the acceptance criteria
- **Medium**: the problem conflicts with business logic, tested parts work incorrectly, or additional features don’t work as designed
- **Low**: bugs don’t contradict the product’s logic and can be easily fixed

Moreover, defects will be reported by the Test Manager to the development team directly via arbitrary communication channel. While the development team is responsible for fixing upcoming issues, the test manager will do the retesting. The following process description gives a global view of testing activities, in particular the defect reporting process.

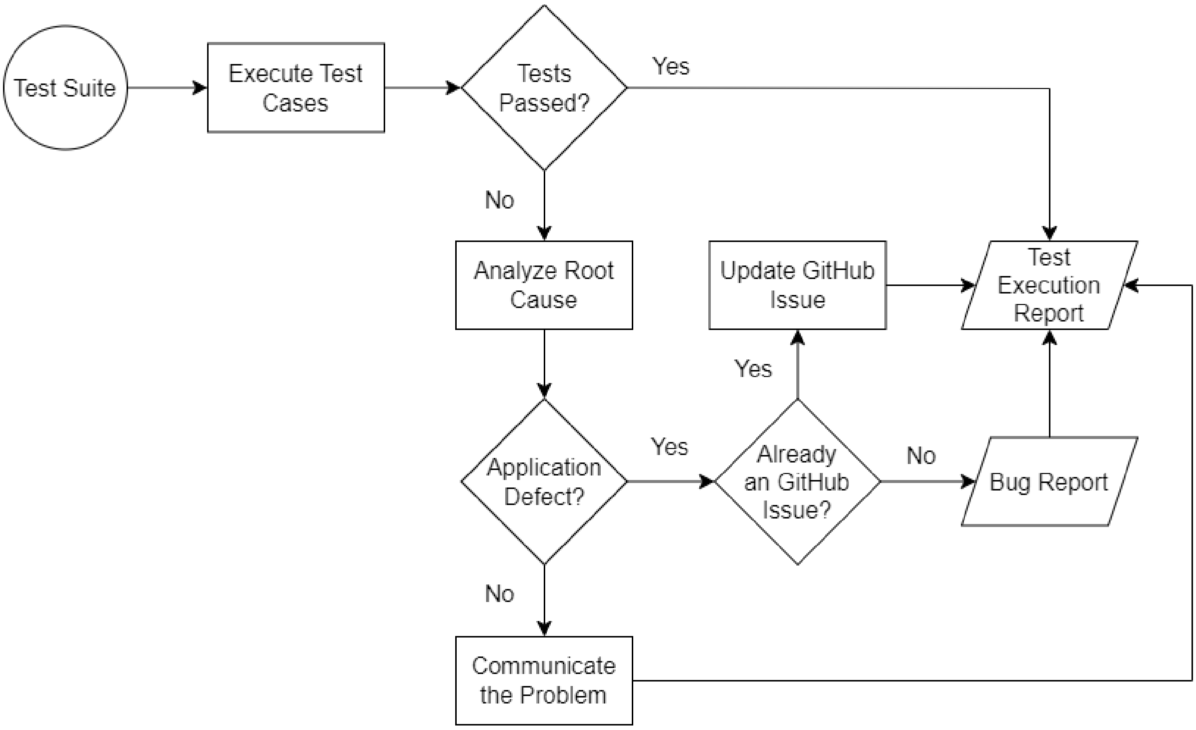

Figure 11.1: Process description of testing activities

12. Pass/Fail Criteria

To pass the system integration test, every feature must function properly. Therefore, the main functionality to receive a nameplate of any asset from an AAS-Server is guaranteed. Furthermore, the QR code of its nameplate must be scannable. All points of integration within the system work as defined in SRS. Therefore, the criteria that denote a successful completion of a test phase is a mandatory run rate at 100% unless a clear reason is given.

13. Suspension Criteria and Resumption Requirements

13.1. Normal Criteria

At the end of each day (20:00 p.m.) testing will be suspended. At that time, all test cases executed during the day should be marked as such. The Test Manager will initiate a backup routine to save the day’s updated test files.

When all test cases of a test suite have been executed, the test will be suspended, and the results are documented for the System Test Report (STR).

13.2. Abnormal Criteria

If the team members report that there are 40% of test cases failed, testing will be suspended until the development team fixes all the failed cases. When a change is being migrated to the test environment, the Test Manager must be notified in advance to schedule a time for the move. After the move has been completed, a retest of previously tested functions should be performed. If a critical incident is reported, as defined by the defect reporting process, testing should be suspended until the defects have been fixed. When the fixed issue is moved back into the test environment, any previously performed tests that are affected by the incident should be performed again to ensure new defects were not created as a result of the fix.

14. Test Environment

It mentions the minimum hardware and software requirements that will be used to test the application.

Testing Tools		
No.	Resources	Descriptions
01	Network	Setup a connection to the internet with a speed of at least 1 mb/s. Firewalls and proxies may block access to the site
02	Computer	Computers with MacOS and Windows 10 that fulfills the minimum requirements for a current version of a modern browser that supports at least ECMAScript 2018 (Chrome, Edge, Safari, Firefox, ...)
03	Smartphone	At least one smart phone with iOS
04	Server	Need an AAS server where assets can be added

The following software are required in addition to client-specific software.

- Firefox 111.0
- Microsoft Edge 110.0
- Google Chrome 111.0
- Application to scan the QR code
- Application to measure time

15. Test Schedule and Budget

Attempted test suite	Date	Responsible
<TS.INST: Local installation>	03.05.2023	Janin Ahlemeyer
<TS.CON: Connect to AssetServer>	03.05.2023	Janin Ahlemeyer
<TS.NAV: Navigation>	04.05.2023	Janin Ahlemeyer
<TS.OV: Overview>	04.05.2023	Janin Ahlemeyer
<TS.NP: Nameplate>	06.05.2023	Janin Ahlemeyer
<TS.PERF: Performance>	06.05.2023	Janin Ahlemeyer

No budget is required as all tests are conducted manually.

16. Plan Approvals

Role	Name and Title	Signature	Date
Team Lead	Adrian Khairi	A. Khairi	25.04.2023
Product Manager	Erika Zhang	E. Zhang	25.04.2023
Test Manager	Ahlemeyer, Janin	Janin Ahlemeyer	25.04.2023