Revision History

Date	Version	Author	Description
02.06.2022	1.0	Rommel Lamanilao	Test plan for Neptune DXP SAP Edition 22-LTS release

Table of Contents

1. INTRODUCTION

1.1. Overview

Neptune DXP — LTS (long-term support) version offers all the features rolled out in one production-ready release. Prior to its actual release, several tests must be executed to ensure that at least all major functionalities are working. This test plan is created to guide the QA Team for the entire testing process.

Also, this document is intended for the testing of **Neptune DXP — SAP Edition 22-LTS** version.

2. OBJECTIVES AND TASKS

2.1. Objectives

The person in charge of the tests should pay attention to this objective before actual tests are performed. The initial release notes can be used as reference to help meet these objectives.

The objectives of the test should be:

- that all new features are working as expected
- that all bug fixes should be tested and verified
- that all major and minor functionalities should be working
- that all other tests defined should PASS

2.2. Tasks

The activities involve in this test plan are the following:

- Determine the scope of the test — tests to be performed and NOT to be performed
- Document the Test Strategy
- Decide the test criteria — suspension and exit criteria
- Plan the test resources
- Plan the test environment
- Plan when and how to test
- Deliver test results as part of the execution

3. SCOPE AND LIMITATIONS

3.1. Scope

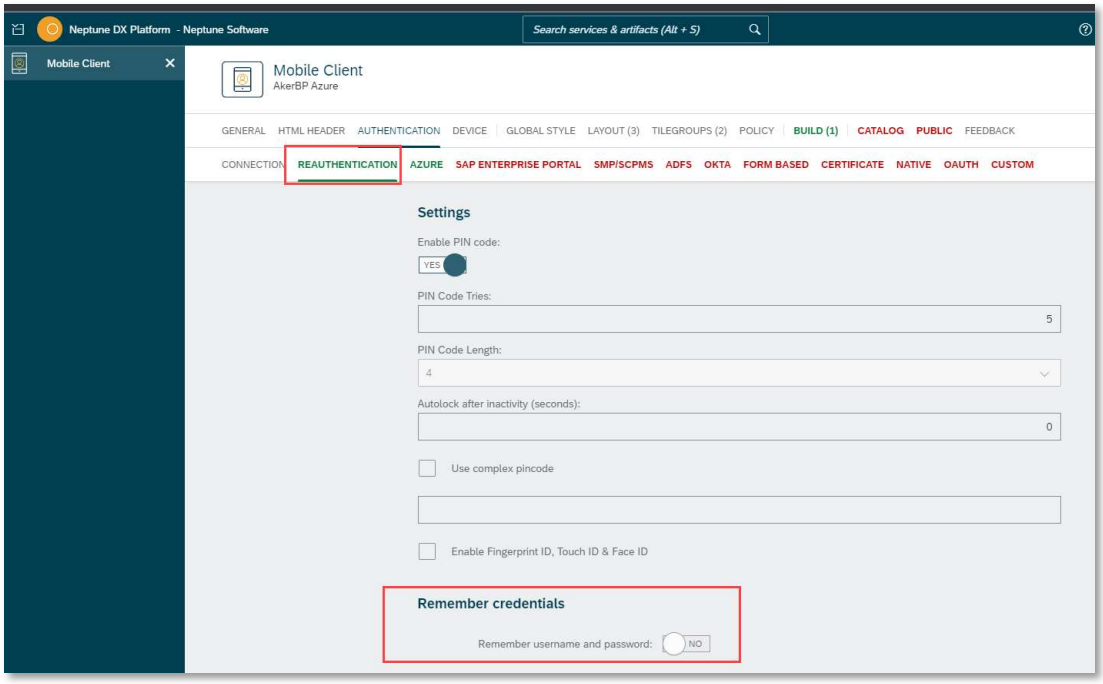
Testing will mostly cover the following:

3.1.1. New Feaures

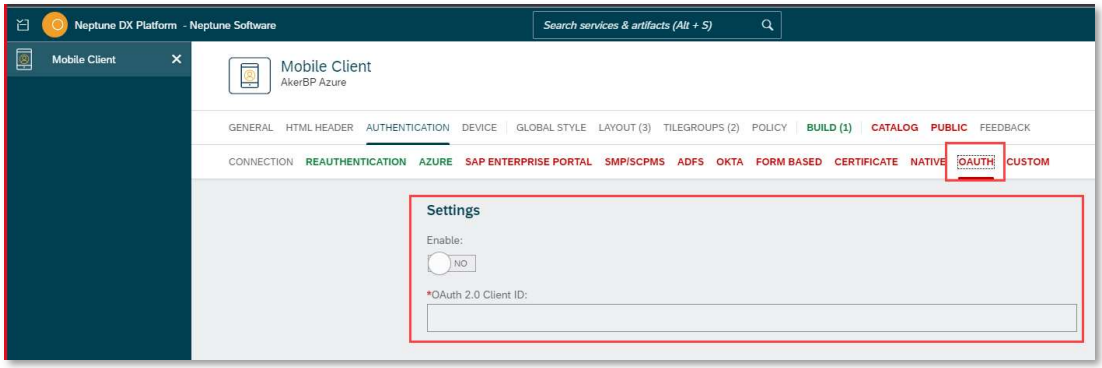
Key contact persons are provided on each feature. Person who is doing the specific feature must refer to the key contact for the procedure on how to test.

- **Bypass logon screen for multiple logon scenarios**

  Key Contact: Thomas Nor

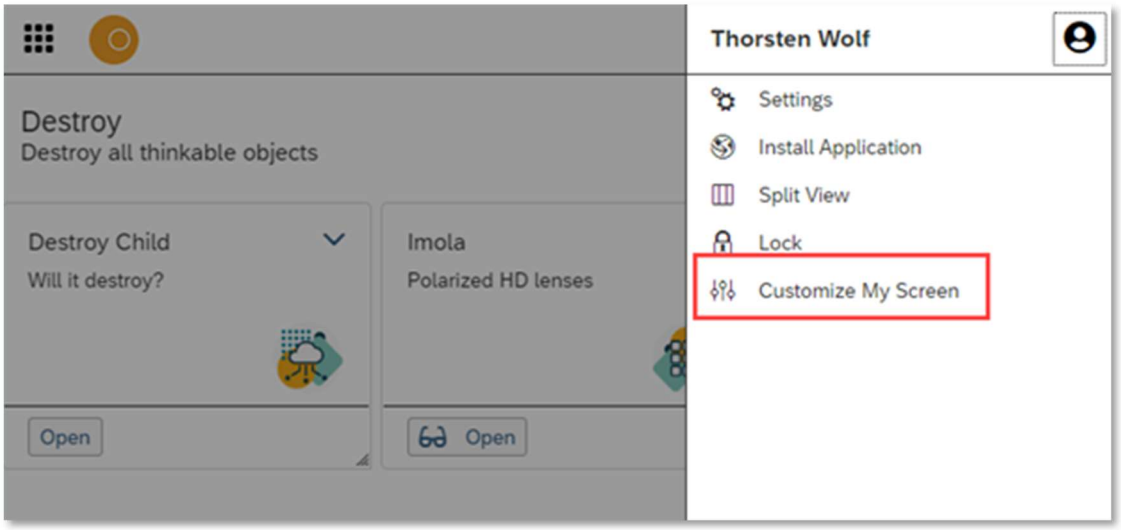


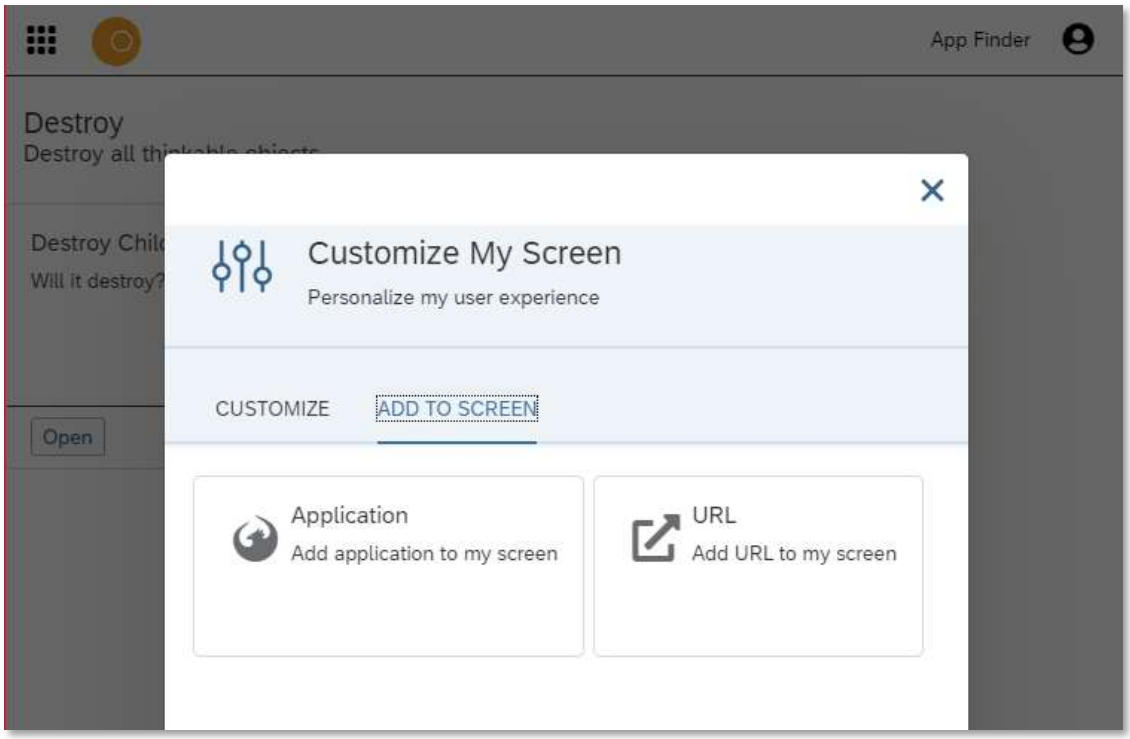
- **Azure AD for P8 — oAuth**
  Key Contact of this new feature: **Thorsten Wolf**

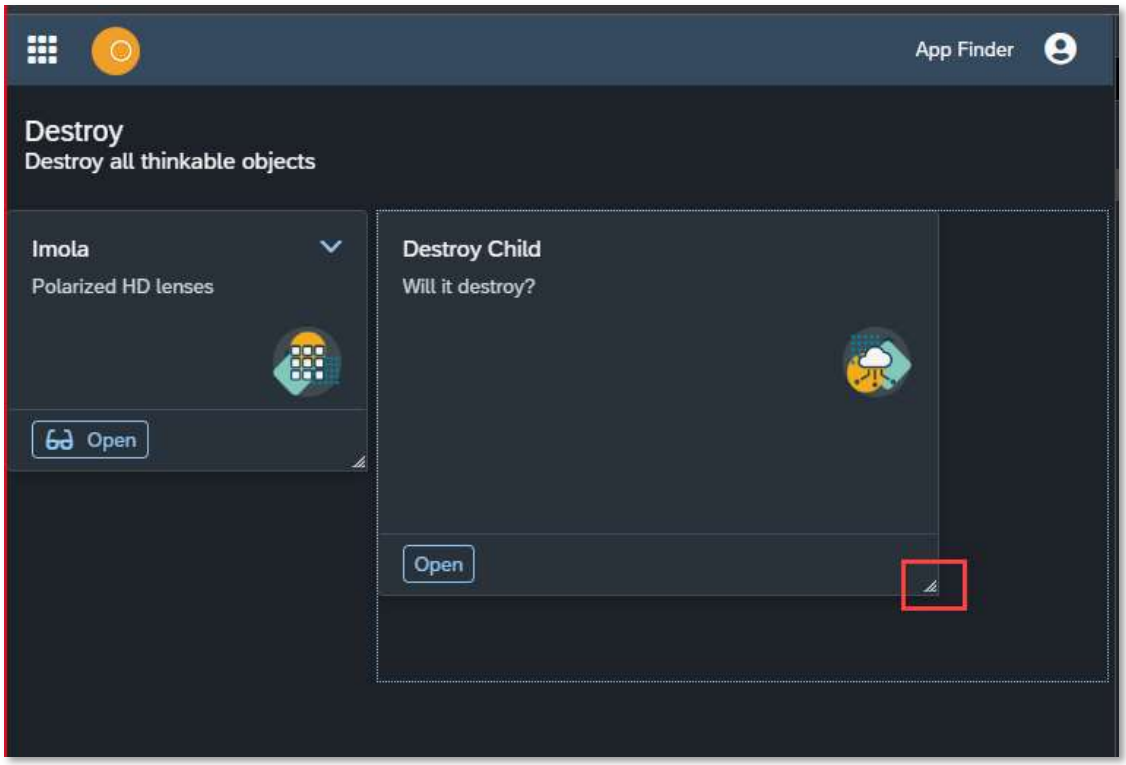


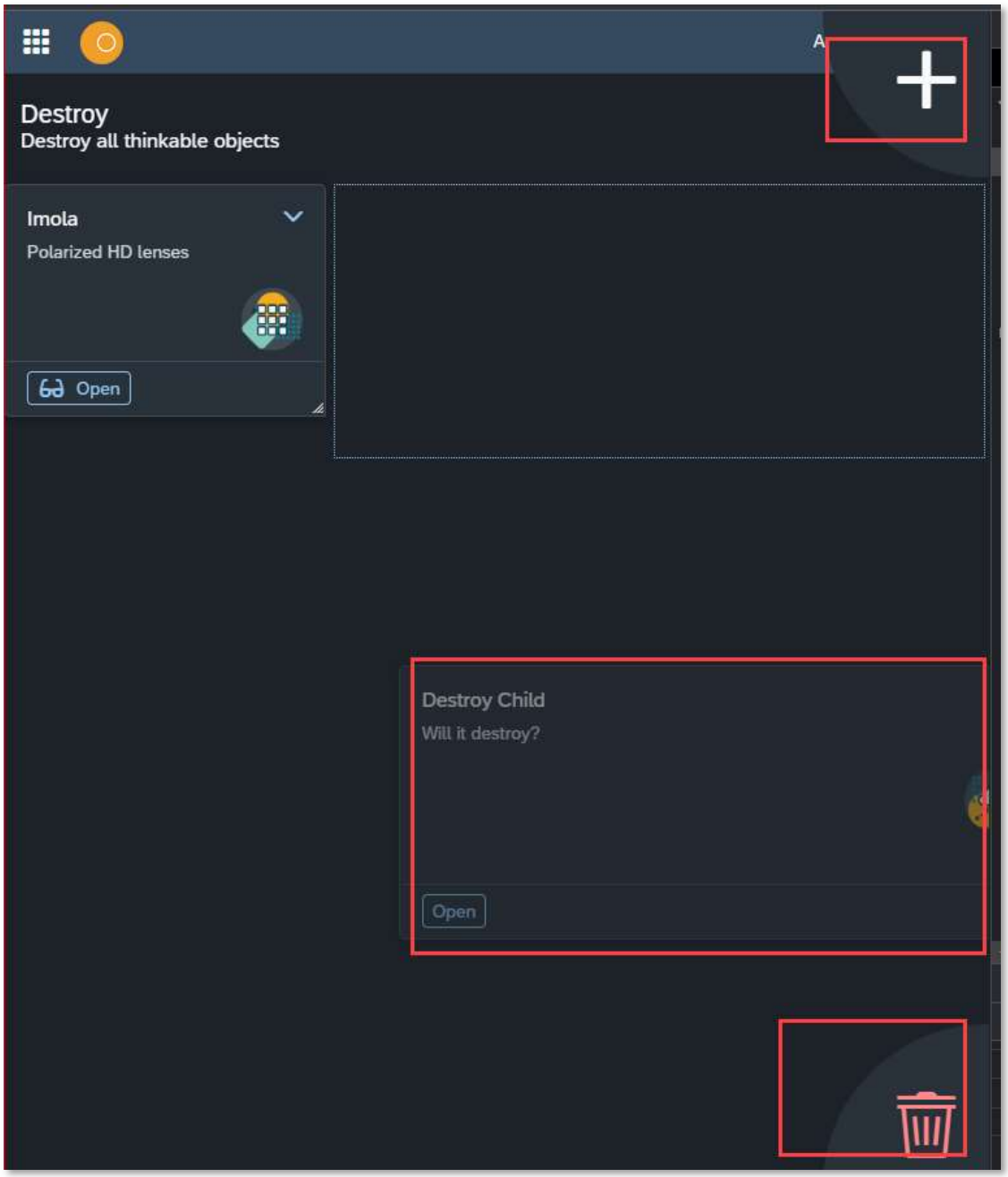
- **Visual tool for admins and end users to create, add, rearrange and resize launchpad tiles**

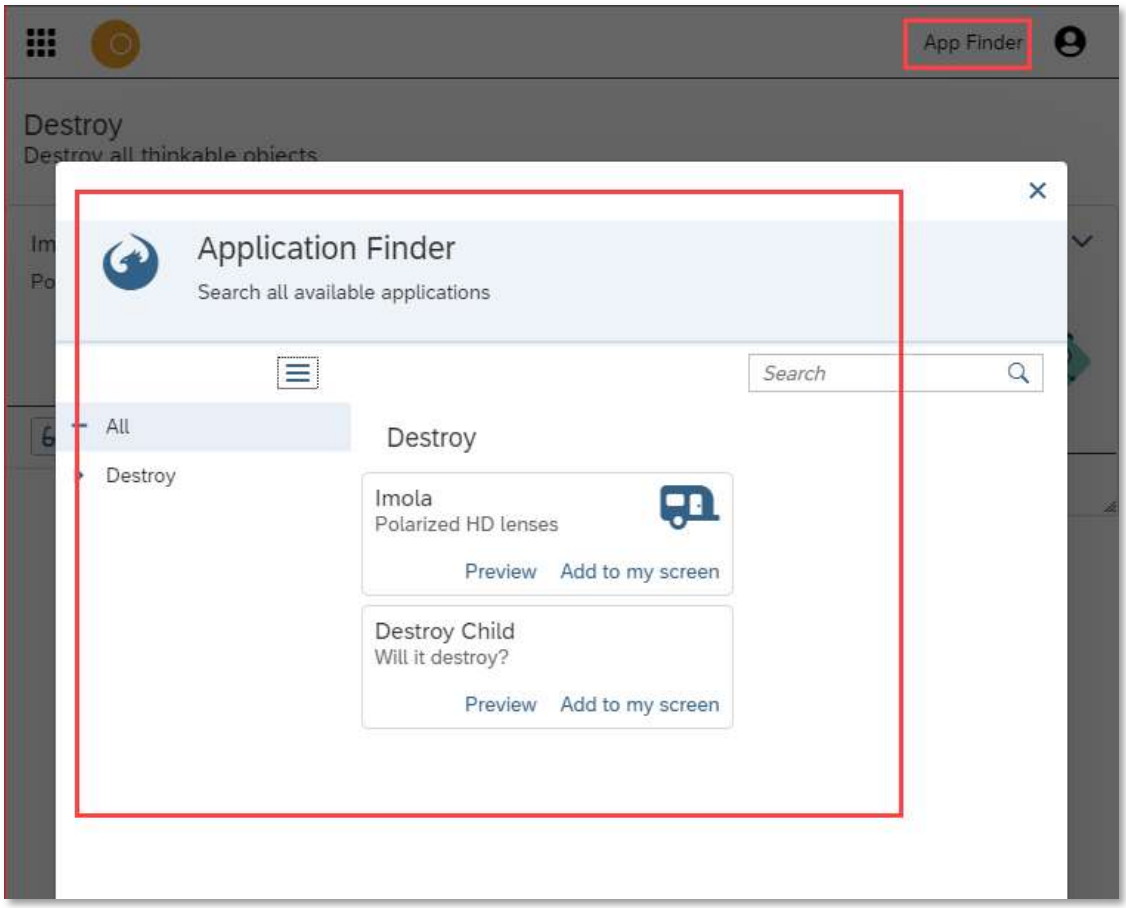
  Key Contact: Morten Prom





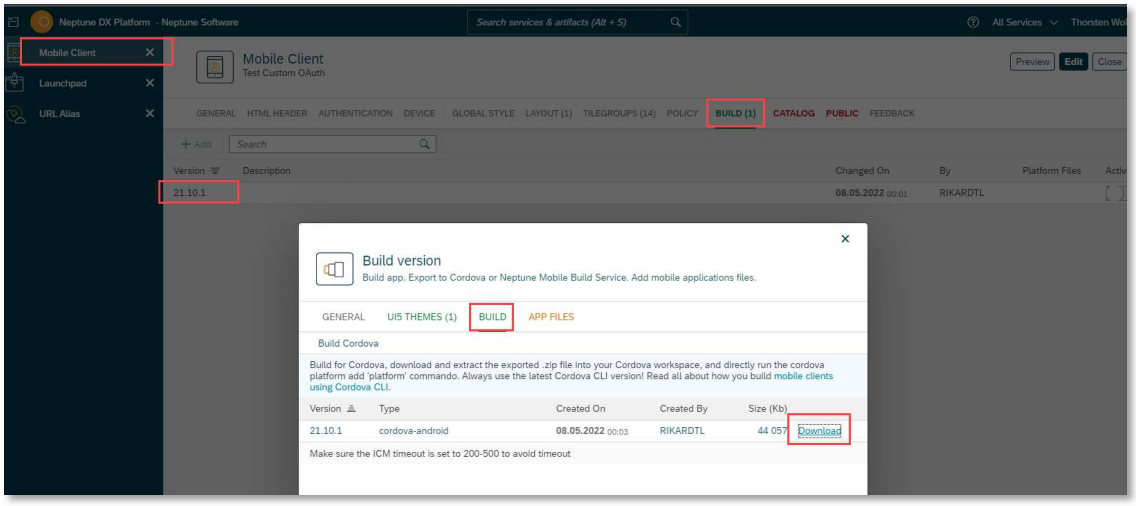

App Finder
Destroy
Destroy all thinkable objects
Imola
Polarized HD lenses
Open
Destroy Child
Will it destroy?
Open


Destroy
Destroy all thinkable objects
Imola
Polarized HD lenses
Open
Destroy Child
Will it destroy?
Open



- **Capacitor including Build Service**
  Download the mobile client zip file and upload it to MBS and see if Capacitor mobile client would work

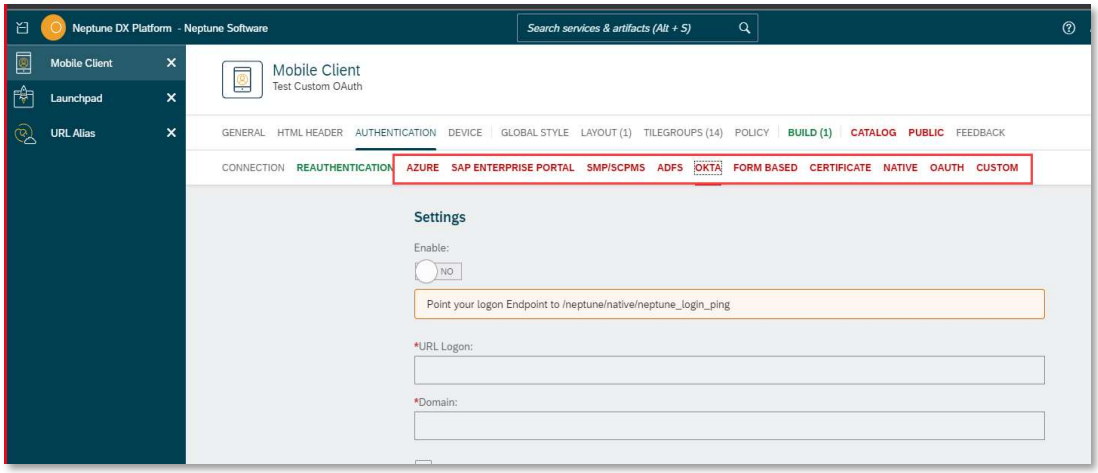
Key Contact: **Jens-Uwe Groß, Thomas Nor and Branislav Djuric**

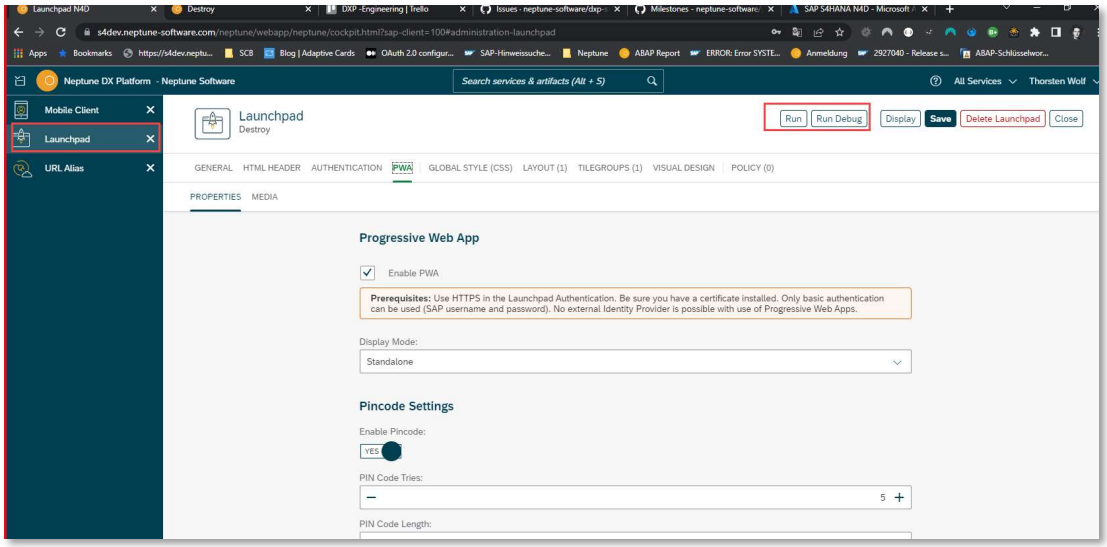


- **Testing multiple mobile clients (especially multiple authentication types) as well as Launchpad Variations (PWA)**

  Thomas Nor will create a sample test mobile client that can be reused in the future to test a specific use case

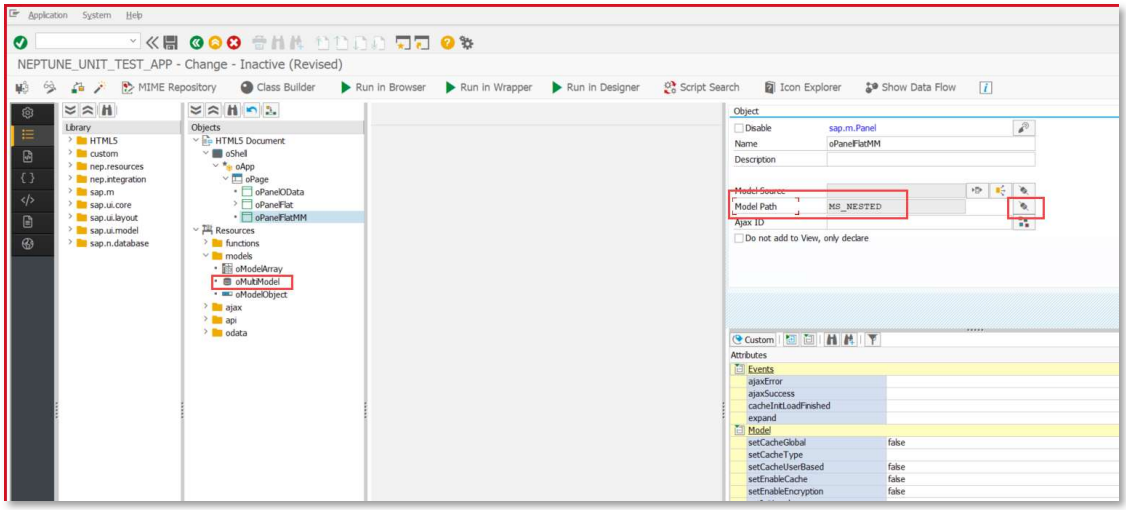
  Key Contact: Thomas Nor

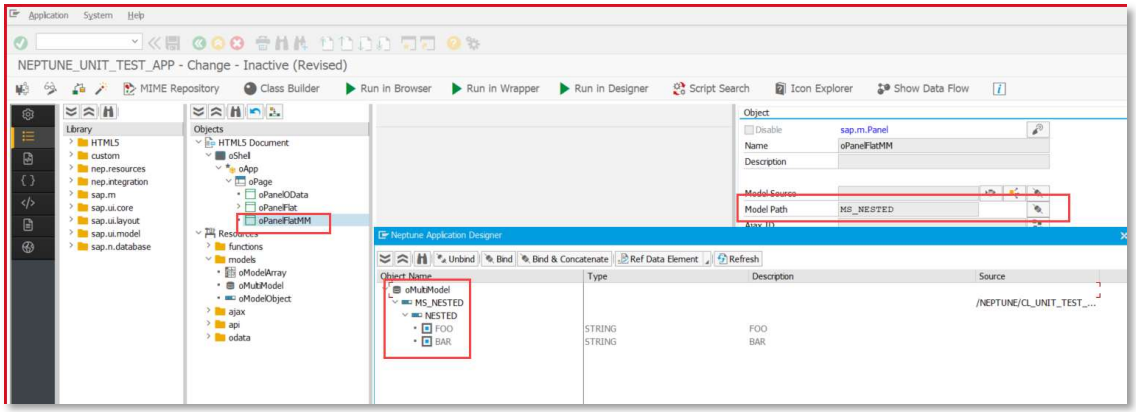




- **Multi-Model/Generic Models for P8**

  Key Contact: Thorsten Wolf





3.1.2. Re-verification of bug fixes

Bug fixes from GitHub Issues must be re-verified to ensure that the provided fixes are still carried out and that they do not introduce new issues.

Since not all have been granted access to GitHub, all fixed issues are copied and temporarily stored in this launchpad application.

3.2. Limitations

- Performance Testing
- Stress Testing
- Database Testing
- GitHub changes and commits not testable from tester's perspective and only developers can verify such as server configuration files

**Not on regular basis but will be performed if required*

4. TESTING STRATEGY

A separate document is created for this section which you can find it here:

Neptune DXP - Test Strategy

5. TEST CRITERIA

5.1. Suspension Criteria

If there is at least one major functionality that is not working, for example, unable to create a mobile build then suspend the release of the build until the development team fixes it.

5.2. Exit Criteria

All test cases are performed with a mandatory of at least 80% PASS rate

6. RESOURCE PLANNING

6.1. System Resource

Resources	Description
Server	The server where the Neptune DXP — SAP Edition is installed
GitHub access	GitHub is used for bug tracking. If necessary, an access must be provided to all members of the QA Team

6.2. Human Resource

Resources	Tasks
Test Manager • Rommel Lamanilao	• Manages the entire testing process • Creates and maintains the test plan and test strategy • Performs the testing tasks together with the rest of the testers
Software Tester • Luis Gonzales • Carlos Daniel Silva Reis Pires • André Carrilho • David Carrilho • Lloyd Trevarthen	• Performs feature testing • Re-verifies the bug fixes in GitHub • Reports bugs/issues found to GitHub

6.2.1. Test Responsible

Here we define the list of testers who will be assigned to perform the testing for Neptune DXP — SAP Edition

- David Carrilho
- Luis Gonzales
- TBD

7. TEST CASE DESIGN

The project of designing the test cases for Neptune DXP — SAP Edition is still in progress.

You can find the project here:
https://dxp-sap-test.neptune-software.cloud/cockpit.html#development-testsunit

username: neptune.software
password: planet8

8. TEST ENVIRONMENT

The following should be setup to ensure software testing process success.

8.1. Test Server Setup

Most recent updates can be found in N4D SAP Test System

Cockpit URL:
https://s4dev.neptune-software.com/neptune/webapp/neptune/cockpit.html?sap-client=100

8.2. Device

Tests will be performed on the following devices:

- Desktop
- Tablet
- Phone (Android and iOS)

8.3. Browser

The following browsers should be installed:

- Google Chrome
- Mozilla Firefox
- MS Edge
- Safari

9. SCHEDULE AND ESTIMATION

9.1. Test Estimation

Testing	Estimate Effort (%)
Smoke Test	5
Feature Test	40
Re-verification	20
System Test	35
TOTAL	100

9.2. Test Schedule

	Planned Date	Actual Date
Test started date	08-June-2022	
Test completed date		

10. RISKS AND MITIGATIONS

Risks	Mitigations
Lack of test resource availability	Facilitate hiring or loan people from another department
Shortage of time to cover all the scope	• Focus on the new features and major changes • Focus on critical areas
Defects are found at a late stage and consumes time to resolve	• Defect management plan is in place to ensure prompt communication and fixing of issues

11. TOOLS

Tools	Usage
GitHub	For reporting of issues found during the testing and for requesting for a new feature
SAP GUI	For testing in the actual SAP environment

12. BUG REPORTING

All encountered issues and feature requests should be reported to GitHub.

12.1. Bug Priority

Bug priority refers to how urgently a bug needs to be fixed. When creating a bug report, one must provide a priority on it. Although GitHub does not have a field for this priority, labels have defined instead.

Levels of bug priority defined as labels in GitHub:

- **Low:** Bug can be fixed at a later date. Other more serious bugs take priority
- **Medium:** Bug can be fixed in the normal course of development and testing
- **High:** Bug must be resolved at the earliest as it affects the system and renders it unusable until it is resolved

12.2. Bug Category

Like in bug priority, GitHub does not have category field as well so we will use labels instead. Category refers to the module or component where the bug is found. This is not a mandatory field but it is nice to have for easy filtering of the issue list.

These are the commonly used labels for categories:

- Use this label for general category particularly for smaller components/applications
    - Cockpit:

- For bigger components/applications:
    - Launchpad
    - App Designer
    - App Editor
    - Script Editor
    - Adaptive Designer
    - Deployment
    - Custom Component