


Usecase UI Frankfurt Release Planning

The content of this template is expected to be fill out for M1 Release Planning Milestone.

**Info**

Use the "Copy" and "Move" options (available under the ..., top right of this page) to duplicate this template into your project wiki.
Use the Wiki to document the release plan. Don't provide PowerPoint.
Use as much diagrams and flow charts as you need, directly in the wiki, to convey your message.

- 1 [Overview](#)
- 2 [Scope](#)
 - 2.1 [What is this release trying to address?](#)
 - 2.2 [Use Cases](#)
 - 2.3 [Requirements](#)
 - 2.4 [Minimum Viable Product](#)
 - 2.5 [Functionalities](#)
 - 2.5.1 [Epics](#)
 - 2.5.2 [Stories](#)
 - 2.6 [Longer term roadmap](#)
- 3 [Release Deliverables](#)
- 4 [Sub-Components](#)
- 5 [Architecture](#)
 - 5.1 [High level architecture diagram](#)
 - 5.2 [Platform Maturity](#)
 - 5.3 [API Incoming Dependencies](#)
 - 5.4 [API Outgoing Dependencies](#)
 - 5.5 [Third Party Products Dependencies](#)
- 6 [Testing and Integration Plans](#)
- 7 [Gaps](#)
- 8 [Known Defects and Issues](#)
- 9 [Risks](#)
- 10 [Resources](#)
- 11 [Release Milestone](#)
- 12 [Team Internal Milestone](#)
- 13 [Documentation, Training](#)
- 14 [Other Information](#)
 - 14.1 [Vendor Neutral](#)
 - 14.2 [Free and Open Source Software](#)

Overview

Project Name	Enter the name of the project
Target Release Name	Frankfurt
Project Lifecycle State	Incubation
Participating Company	China Mobile, Huawei

Scope

What is this release trying to address?

The Usecase-UI Frankfurt release has following primary objectives:

1. Supporting use cases identified by ONAP and integration:

Multi-domain Optical Network Services
E2E Network Slicing Use Case

2. UUI enhancement and new features:

JDK upgrade evaluate and migrate (Depend on the commit resource)

- Document current upgrade component strategy(TSC must have) [USECASEUI-353 - Getting issue details...](#) [STATUS](#)
- SECCOM Perform Software Composition Analysis - Vulnerability tables(TSC must have) - plan to solve the most of the security issues, but also depends on the commit resource [USECASEUI-354 - Getting issue details...](#) [STATUS](#)
- SECCOM Password removal from OOM HELM charts(TSC must have) [USECASEUI-355 - Getting issue details...](#) [STATUS](#)
- SECCOM HTTPS communication vs. HTTP(TSC must have) - Based on the resource contribution [USECASEUI-356 - Getting issue details...](#) [STATUS](#)

Use Cases

- Multi-domain Optical Network Services
- E2E Network Slicing Use Case

Requirements

Describe the use case this release is targeted for (better if reference to customer requirements).

- Multi-domain Optical Network Services [REQ-141 - Getting issue details...](#) [STATUS](#) [REQ-37 - Getting issue details...](#) [STATUS](#)
- E2E Network Slicing [REQ-146 - Getting issue details...](#) [STATUS](#) [REQ-158 - Getting issue details...](#) [STATUS](#)

Minimum Viable Product

Usecase-UI will include the necessary subcomponents supporting the primary objectives: meeting platform maturity goals and supporting the use cases.

- CSMF BE Component for 5G slicing
- CSMF FE Component for 5G slicing
- UUI BE Component (including NSMF for 5G slicing)
- UUI FE Component (including NSMF for 5G slicing)

Functionalities

List the functionalities that this release is committing to deliver by providing a link to JIRA Epics and Stories. In the JIRA Priority field, specify the priority (either High, Medium, Low). The priority will be used in case de-scoping is required. Don't assign High priority to all functionalities.

Epics

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
USECASEUI-334	Frankfurt release planning milestone		Oct 04, 2019	Nov 08, 2019	Oct 11, 2019	Unassigned	None	=	CLOSED	Done

Stories

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
USECA SEUI-419	Service topology in usecase ui		Mar 25, 2020	Apr 22, 2020		Unassigned	None	=	CLOSED	Done
USECA SEUI-414	Add inter domain link provisioning support for MDONS		Mar 12, 2020	Apr 03, 2020	Mar 13, 2020	Unassigned	None	=	CLOSED	Done

USECA SEUI-375	Support CCVPN-E-LINE over OTN Inter Domain Links		Jan 20, 2020	Apr 27, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-374	NSMF portal Development		Jan 19, 2020	Feb 03, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-373	CSMF Portal Development		Jan 19, 2020	Feb 03, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-371	Add OTN/L1 Service Creation Support		Dec 19, 2019	Apr 03, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-370	NSMF slicing resource monitor component		Dec 05, 2019	Jan 19, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-369	NSMF slicing resource management		Dec 05, 2019	Jan 19, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-368	NSMF slicing order management		Dec 05, 2019	Jan 19, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-367	UUI committer removal		Nov 26, 2019	Nov 26, 2019	Unassigned	None	=	CLOSED	Done
USECA SEUI-356	SECCOM HTTPS communication vs. HTTP (TSC must have)		Nov 14, 2019	Feb 26, 2020	Keguang He	None	=	CLOSED	Done
USECA SEUI-355	SECCOM Password removal from OOM HELM charts(TSC must have)		Nov 14, 2019	Apr 24, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-354	SECCOM Perform Software Composition Analysis - Vulnerability tables(TSC must have)		Nov 14, 2019	Feb 20, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-353	Document current upgrade component strategy(TSC must have)		Nov 14, 2019	Jan 20, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-352	Create NSMF functional Pages and Components		Nov 07, 2019	Jan 15, 2020	Unassigned	None	=	CLOSED	Done
USECA SEUI-351	Create CSMF Portal		Nov 07, 2019	Jan 14, 2020	Unassigned	None	=	CLOSED	Done

16 issues

Longer term roadmap

Indicate at a high level the longer term roadmap. This is to put things into the big perspective.

Release Deliverables

Indicate the outcome (Executable, Source Code, Library, API description, Tool, Documentation, Release Note, etc) of this release.

Deliverable Name	Deliverable Description
Source Code	Source code for all UUI components
Maven Artifacts	Maven Artifacts for all UUI components
Docker Containers	Docker container associated with UUI components

Sub-Components

List all sub-components part of this release.

Activities related to sub-components must be in sync with the overall release.

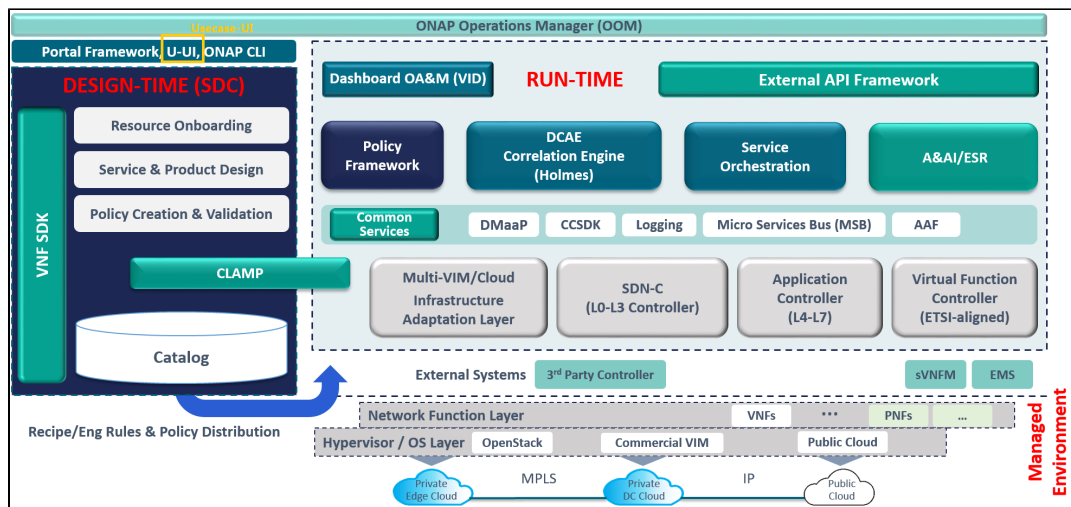
Sub-components are repositories and are consolidated in a single centralized place. Edit the [Release Components name for your project](#) in the centralized page.

Architecture

High level architecture diagram

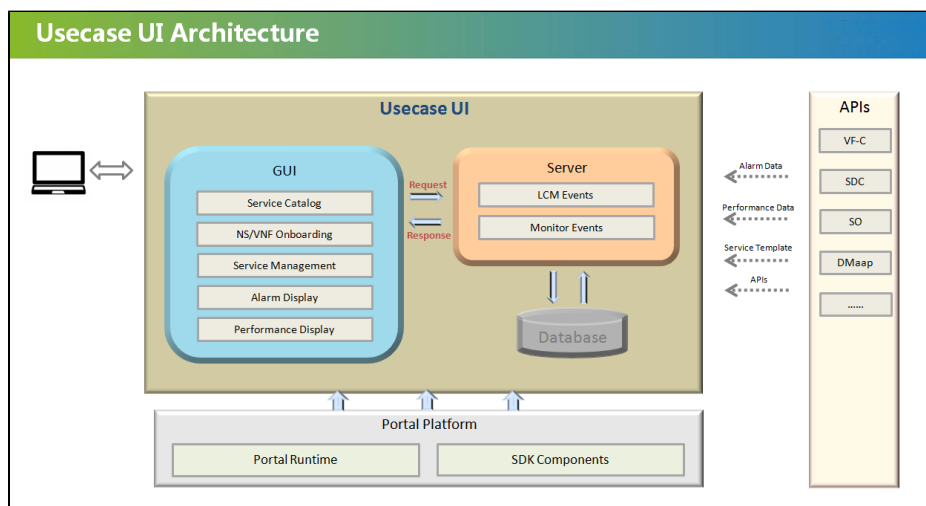
At that stage within the Release, the team is expected to provide more Architecture details describing how the functional modules are interacting.

Indicate where your project fit within the [ONAP Architecture diagram](#).



Block and sequence diagrams showing relation within the project as well as relation with external components are expected.

Anyone reading this section should have a good understanding of all the interacting modules.



Platform Maturity

Please fill out the centralized wiki page: [Frankfurt Release Platform Maturity](#)

API Incoming Dependencies

List the API this project is expecting from other projects.
Prior to Release Planning review, Team Leads must agreed on the date by which the API will be fully defined. The API Delivery date must not be later than the [release API Freeze date](#).

Prior to the delivery date, it is a good practice to organize an API review with the API consumers.

API Name	API Description	API Definition Date	API Delivery date	API Definition link (i.e.swagger)
Portal Platform API	API for integration of portal applications			
Catalog API (SDC)	API to read the NS and VNF Catalog			
SO API(Modeling)	API for NS instantiation and termination			
MSB API	API for registration and use of micro-services bus			
A&AI API	API for getting inventory and image management			
VF-C API	API for NS instantiation and termination			
DataLake API	API for FCAPS (VNF/VM monitoring)			
External API	API for external data			

API Outgoing Dependencies

None

Third Party Products Dependencies

None.

Testing and Integration Plans

Provide a description of the testing activities (unit test, functional test, automation,...) that will be performed by the team within the scope of this release.

Describe the plan to integrate and test the release deliverables within the overall ONAP system.
Confirm that resources have been allocated to perform such activities.

Gaps

This section is used to document a limitation on a functionality or platform support. We are currently aware of this limitation and it will be delivered in a future Release.
List identified release gaps (if any), and its impact.

Gaps identified	Impact

Known Defects and Issues

Provide a link toward the list of all known project bugs.

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
No issues found										

Risks

List the risks identified for this release along with the plan to prevent the risk to occur (mitigation) and the plan of action in the case the risk would materialized (contingency).

Risk identified	Mitigation Plan	Contingency Plan
To fill out	To fill out	To fill out

Resources

Fill out [the Resources Committed to the Release](#) centralized page.

Release Milestone

The milestones are defined at the [Release Level](#) and all the supporting project agreed to comply with these dates.

Team Internal Milestone

This section is optional and may be used to document internal milestones within a project team or multiple project teams. For instance, in the case the team has made agreement with other team to deliver some artifacts on a certain date that are not in the release milestone, it is recommended to provide these agreements and dates in this section.

It is not expected to have a detailed project plan.

Date	Project	Deliverable

Documentation, Training

- Highlight the team contributions to the specific document related to the project (Config guide, installation guide...).
- Highlight the team contributions to the overall Release Documentation and training asset
- High level list of documentation, training and tutorials necessary to understand the release capabilities, configuration and operation.
- Documentation includes items such as:
 - Installation instructions
 - Configuration instructions
 - Developer guide
 - End User guide
 - Admin guide
 - ...



Note

The Documentation project will provide the Documentation Tool Chain to edit, configure, store and publish all Documentation asset.

Other Information

- Vendor Neutral

If this project is coming from an existing proprietary codebase, ensure that all proprietary trademarks, logos, product names, etc. have been removed. All ONAP deliverables must comply with this rule and be agnostic of any proprietary symbols.

- Free and Open Source Software

FOSS activities are critical to the delivery of the whole ONAP initiative. The information may not be fully available at Release Planning, however to avoid late refactoring, it is critical to accomplish this task as early as possible.

List all third party Free and Open Source Software used within the release and provide License type (BSD, MIT, Apache, GNU GPL,...).

In the case non Apache License are found inform immediately the TSC and the Release Manager and document your reasoning on why you believe we can use a non Apache version 2 license.

Each project must edit its project table available at [Project FOSS](#).

Charter Compliance

The project team comply with the [ONAP Charter](#).