

VVP Release Planning Honolulu

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

- 1 [Overview](#)
- 2 [Scope](#)
 - 2.1 [What is this release trying to address?](#)
 - 2.2 [Requirements](#)
 - 2.3 [Minimum Viable Product](#)
 - 2.4 [Functionalities](#)
 - 2.4.1 [Epics](#)
 - 2.4.2 [Tasks](#)
 - 2.5 [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	Honolulu
Project Lifecycle State	Either Incubation,
Participating Company	AT&T Ericsson, Nokia, China Mobile, Orange

Scope

What is this release trying to address?

Updates to VNF Heat Template test scripts in alignment with ONAP platform's VNF Requirements

Requirements

VNF developer wants to know whether the HEAT templates for his VNF are conformant with ONAP requirments.

Minimum Viable Product

Test scripts for validating VNF HEAT templates in accordance with numbered VNF requirements

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
No issues found										

Tasks

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
VVP-486	Finalize Documentation	✓	Nov 06, 2020	Nov 19, 2020		Unassigned	None	=	CLOSED	Done
VVP-485	Verify readiness of release artifacts	✓	Nov 06, 2020	Nov 17, 2020		Unassigned	None	=	CLOSED	Done
VVP-484	JIRA Cleanup	✓	Nov 06, 2020	Nov 18, 2020		Unassigned	None	=	CLOSED	Done
VVP-481	Review license scan issues	✓	Oct 22, 2020	Oct 28, 2020		Unassigned	None	=	CLOSED	Done
VVP-480	Verify readiness of release artifacts	✓	Oct 22, 2020	Oct 28, 2020		Unassigned	None	=	CLOSED	Done
VVP-479	JIRA Cleanup	✓	Oct 22, 2020	Oct 28, 2020		Unassigned	None	=	CLOSED	Done
VVP-476	Update integration weather board	✓	Sep 18, 2020	Sep 18, 2020		Unassigned	None	=	CLOSED	Won't Do
VVP-475	Update documented risks	✓	Sep 18, 2020	Sep 18, 2020		Unassigned	None	=	CLOSED	Won't Do
VVP-474	Resolve high/highest priority JIRA issues	✓	Sep 18, 2020	Oct 09, 2020		Unassigned	None	=	CLOSED	Done
VVP-473	Deliver updated container to integration team	✓	Sep 18, 2020	Sep 18, 2020		Unassigned	None	=	CLOSED	Won't Do
VVP-472	Complete project test plan	✓	Sep 18, 2020	Sep 18, 2020		Unassigned	None	=	CLOSED	Won't Do
VVP-471	Review license scan issues	✓	Sep 18, 2020	Oct 06, 2020		Unassigned	None	=	CLOSED	Done
VVP-468	Add deployment properties to service.py	✓	Sep 08, 2020	Sep 15, 2020		Unassigned	None	=	CLOSED	Done
VVP-467	Add update_message for VSP	✓	Aug 27, 2020	Sep 01, 2020		Unassigned	None	=	CLOSED	Done
VVP-466	Add per-instance client configuration overrides	✓	Aug 26, 2020	Aug 27, 2020		Unassigned	None	=	CLOSED	Done
VVP-465	Review and update INFO. yaml	✓	Aug 14, 2020	Sep 01, 2020		Unassigned	None	=	CLOSED	Done
VVP-464	Verify that test coverage goals have been met	✓	Aug 14, 2020	Sep 04, 2020		Unassigned	None	=	CLOSED	Done
VVP-463	Update Release Platform Maturity and CII badging	✓	Aug 14, 2020	Sep 08, 2020		Unassigned	None	=	CLOSED	Done
VVP-462	Resolve high/highest priority JIRA issues	✓	Aug 14, 2020	Sep 04, 2020		Unassigned	None	=	CLOSED	Done
VVP-461	Update the FOSS wiki page	✓	Aug 14, 2020	Sep 01, 2020		Unassigned	None	=	CLOSED	Done

Showing 20 out of [66 issues](#)

Longer term roadmap

Alignment of VNF Heat template tests with ONAP platform requirements as defined in the VNF Requirements project

Release Deliverables

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

Deliverable Name	Deliverable Description
VVP validation scripts	org.onap.vvp.validation-scripts
documentation	https://docs.onap.org/en/latest/guides/onap-provider/vnvalidator.html

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: [Honolulu Release Platform Maturity](#)

VVP is a standalone testing tool - it is not an ONAP platform component, and the platform release maturity is not applicable.

• API Incoming Dependencies

List the API this project is expecting from other projects.

NONE.

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)
N/A	N/A	N/A	N/A	N/A

• API Outgoing Dependencies

API this project is delivering to other projects.

NONE

API Name	API Description	API Definition Date	API Delivery date	API Definition link (i.e.swagger)
N/A	N/A	N/A	N/A	N/A

• Third Party Products Dependencies

Third Party Products mean products that are mandatory to provide services for your components. Development of new functionality in third party product may or not be expected.

List the Third Party Products (OpenStack, ODL, RabbitMQ, ElasticSearch, Crystal Reports, ...).

Name	Description	Version
To fill out	To fill out	To fill out

In case there are specific dependencies (Centos 7 vs Ubuntu 16. Etc.) list them as well.

The VVP scripts use python 3 and other opensource software. View the FOSS section for the third party software that's needed for VVP.

• 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.

The VVP scripts are marked with a requirement # from the VNFRQTs project, and the mapping table [here](#) is updated dynamically and can be used to verify to requirement to test mapping is up to date. For each commit to the vvp project, automated tests are run to ensure code quality before a commit can be merged to the project.

Because VVP is not deployed as a runtime component with the full ONAP platform, there are no integration tests that need to be performed.

• 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
None	None

• Known Defects and Issues

None

• 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).

Please update any risk on the centralized wiki page - [Honolulu Risks](#)

• Resources

Please see the INFO.yaml files associated with each repo as the authoritative sources of information. <https://gerrit.onap.org/r/admin/repos/q/filter:vvp>

• Release Milestone

The milestones are defined at the [Release Planning: Honolulu](#) 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
To fill out	To fill out	To fill out

refer to the [Frankfurt Release timeline](#)

• Documentation, Training

Please update the following centralized wiki: [Honolulu Documentation](#)

That includes

- Team contributions to the specific document related to the project (Config guide, installation guide...).
- 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](#).