

OVP Portal Requirements

09 Jun 2020

Attendees: [Georg Kunz](#) [Brandon Wick](#) [Lincoln Lavoie](#) [Jim Baker](#)

OVP portal

- repo based development for interfaces to test results
- portal is basically a dynamic representation of test.db results
- Initial plan: One-time development effort by external company, followed by community based support

Open Questions:

- Long term maintenance suggests a hosted project under LFN - ONAP or OPNFV
 - OPNFV/CNTT technically have the alignment to the OVP outcomes
 - ONAP has more development resources and may be a better project owner for the portal?
- Community has been unable to support the current portal, why would the community be better equipped on some new portal?
- How to get to a specific project plan?
 - Have existing portal and a list of improvements
 - UNH is a knowledgeable supplier - can we get confirmation of UNH willingness to build/support

☐ [Lincoln Lavoie](#) to develop SOW for presentation and review **30 Jun 2020**

18 May 2020

Attendees: [Heather Kirksey](#) [Jim Baker](#) [Lincoln Lavoie](#) [Brandon Wick](#) [Georg Kunz](#)

Problem statement:

- Open source development (Dovetail) community dissolved
- Lacking skill set in UI development - Intracomm was contracted
 - Original portal leveraged from OpenStack

28 May 2020

Attendees: [Lincoln Lavoie](#) [Rabi Abdel](#) [Georg Kunz](#) [Brandon Wick](#)

Reviewed requirements set below.

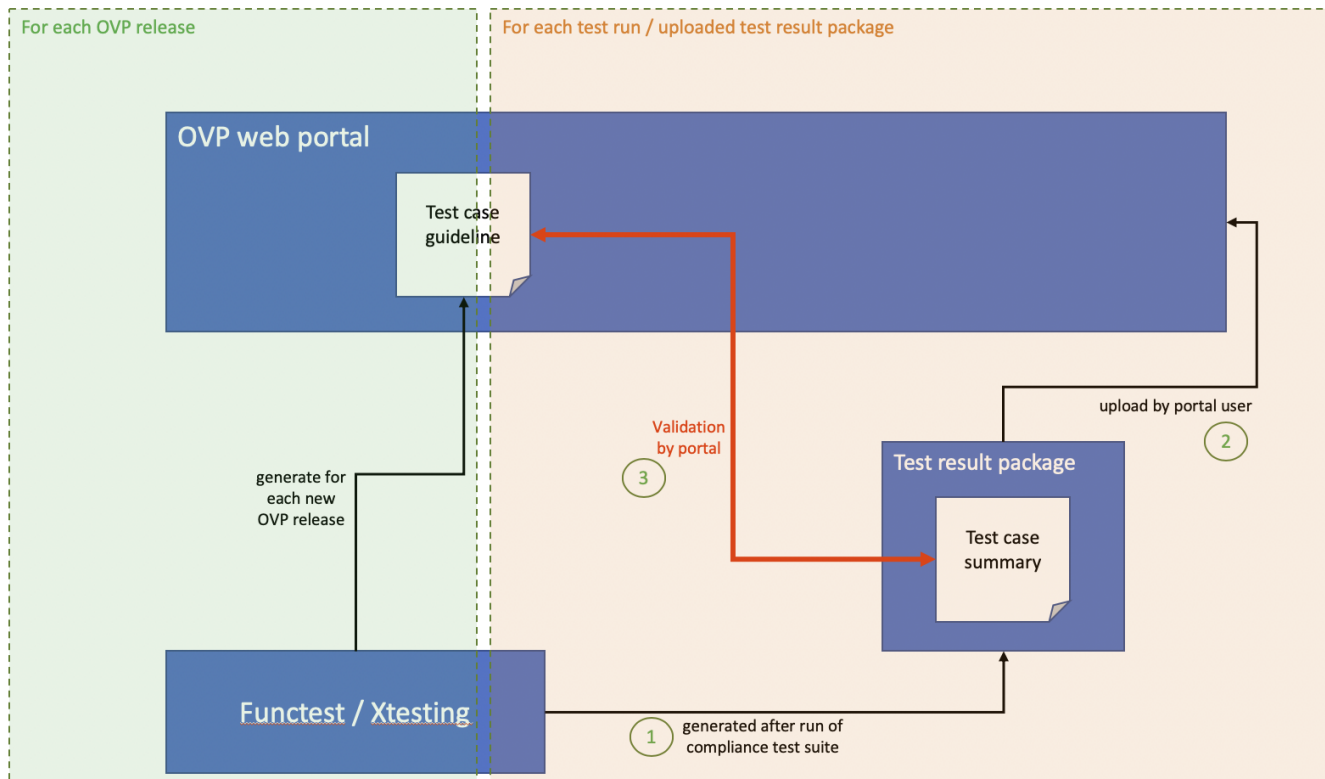
- We reviewed and agreed the requirement below down to the "Results Format" section.
- Group will continue to review that section offline to prepare for next week
- [Lincoln Lavoie](#) will create a flow diagram for the portal workflow.

Requirements

This list of requirements should be expanded in level of detail to support an RFP:

- General requirements
 - The portal must at least provide the same functionality as today's portals (see <https://nfvi-verified.lfnetworking.org/#/> AND <https://vnf-verified.lfnetworking.org/#/>)
 - One portal should support multiple programs, that can be searched / filtered by program/badge type on the public listing page.
 - The public lists should be searchable, and allow filtering by the program type, company, and other columns displayed on the main page.
 - High-level use cases
 1. Support upload, validation, display, sharing, and manage test results and application by "user"
 2. Support a review workflow of test results by "reviewers"
 3. Publicly list companies and products which have obtained a badge in a "marketplace", as marked by "admin"
- Test Result Management
 - authenticated users (role "user") must be able to
 - upload test results
 - edit meta data (application) of a test result set (product name, etc.)
 - view, delete, and edit only their own test results
 - change status of a test result between "private" and "for review"
- review management
 - authenticated reviewers (role "reviewer") must be able to:
 - access only to test results set to state "for review" (not all uploaded results)

- cast a vote (-1, 0, 1) on instance of test results submitted to review
 - Add comment along with there vote (i.e. why they voted -1, etc.)
 - OVP release management
 - Management of releases of OVP (create new, edit, delete) must be runtime operations, i.e., not requiring new versions of the portal
 - a OVP release comprises
 - a unique identifier (e.g. OVP020.09)
 - links to documentation
 - a list of test cases for each program type that are mandatory or optional
 - This list is used to validate if a set of submitted results meets the requirements for the OVP release.
 - portal lifecycle management
 - all management operations on test results, market place entries, users, and new releases of OVP must be runtime operations not requiring new builds of the web portal
 - separation of LCM of the portal instance (responsibility of LF IT) and content (responsibility of OVP admins)
 - Public List Management
 - "admins" (user role) must be able to manage entries of the marketplace (create, edit, delete)
 - all entries of the marketplace must be stored in persistent storage
 - entries must include support to display a company logo (provided by the user submitting the application)
 - market place data items per entry: see current fields + <add more if needed, Brandon?>
 - A full list of fields for the existing NFVI and VNF programs will be provided by the LFN CVC
 - LFN CVC will provide guidance on which fields will display on the top level list (main page) or only in a detailed listing (linked to from the main page)
 - User Management
 - Users log in through a Linux Foundation Open ID
 - A user logging in for the first time is automatically assigned the "user" role.
 - User Roles
 - "user"
 - Can upload and manage test results
 - Can only see own test results
 - Can create an application to submit their results to review
 - "reviewer"
 - can see all test results marked as "for review" by its "user" (the user is the owner of the results / application)
 - "admin"
 - can manage assigned user roles
 - can manage (create, update, delete) entries to the marketplace
 - A portal user can have multiple assigned roles, i.e. Jo can be assigned the roles of "admin" and "user"
 - Results Format
 - Should be as flexible as possible.
 - Results are uploaded as a zip or tar.gz file, other formats will be rejected
 - Must include a "test result summary" in the archive file root
 - The "test results summary" will include:
 - Version of the tool used (i.e. what version of functest was running)
 - Date & time of the test run
 - validation and display of test results (see also terminology below)
 - the web portal must validate uploaded test results by comparing the "test result summary" to a "test result guideline"
 - "test result guideline": source of truth
 - list of all test cases which are part of a given OVP release
 - use case: detect if test cases are missing from uploaded test results
 - the expected result for passing each test case (functional tests: "pass", non-functional: "value")
 - stored in web portal only
 - "test result summary"
 - part of the "test result package" generated by test tool
 - json formatted
 - should include in addition to today (**AP on test tooling team**)
 - OVP release ID (e.g. 2020.10)
 - OVP program type (e.g. NVFI, VNF, ...)
 - example of a "test result package" currently generated by test tooling:
 - <http://artifacts.opnfv.org/functest/DUXGBAW88H7Q.zip>
 - optional requirements, requires close collaboration with and input from test tooling team
 - define a schema for formal validation of test result summary
 - define a schema for formal validation of test result guide
- Validation of Results
 - The portal should be capable of validating the submitted results.
 - Validation checks the results contain the correct test cases (minimum set) and those test cases pass
 - The set of test cases (minimum set) should be controlled by the portal admin.
 - An OVP release may include multiple "minimum sets" that apply to different releases of Functest and OpenStack



- Terminology
 - "test result package": archive containing "test result summary" file and individual logs
 - "test result summary": json formatted file containing a summary of all test cases / one run of the compliance test tool
 - "test result guideline": json formatted file containing all tests which are part of an OVP release + expected result for passing a test

Requirements (as noted during the call on 18 May 2020):

- Development
 - Represent the workflow of the respective participants
 - xtesting results uploaded - schema for uploads
 - portal to validate/accept inputs - version checking
 - Allow authorized set of people to manage the badging administration
 - No regression of functionality from Dovetail implementation
 - Alignment of results formats from ONAP/OPNFV
 - ?Allow all versions to be uploaded - deprecate older versions?
 - Bring forward existing badging - unlikely to support old schema/results
 - Minimum: current xtesting and ONAP results - schemas
 - Converged portal (VNF/NFVIs/CNF)
 - Built on LF infra (shared vs. dedicated)
 - Desire portal to be managed without LF IT interactions
 - Naming changes?
 - Define that early
 - User management
 - integrated with LF SSO
 - Privileged users for management
 - 3rd party OVP lab integration
 - Use existing portal as a basis for MVP definition
 - Timeline?
 - Objective: full MVP implementation - Oct 2020 (ONES Sept 28)
 - Public availability
 - Migrate existing data
 - Internal Go-Live – 01 Sep 2020
 - Development time – start 01 Jul 2020
 - Review submissions to RFP

- RFP open time – 18 Jun 2020
 - RFP definition complete – 05 Jun 2020
 - Budget setting/approval – LF GB 17 Jun 2020
 - Vendor qualification - at least 3 vendors
- Support for incoming data sets and badging processes
- Hosting
- Maintenance

☑ Georg Kunzto expound on requirements by 25 May 2020