Grant Agreement No.: 732638
Call: H2020-ICT-2016-2017
Topic: ICT-13-2016
Type of action: RIA

D3.3: Requirements and specifications for the 2nd cycle

Work package	WP 3
Task	Task 3.1-3.5
Due date	31/12/2018
Submission date	17/03/2020
Deliverable lead	Imec
Version	3
Authors	Brecht Vermeulen (imec), Wim Van der Meerssche (imec), Thijs Walcarius (imec), Radomir Klacza (SU), Albert (Yiu Quan) Su (SU), Pauline Gaudet Chardonnet (SU), Dimitris Dechouniotis (NTUA), Costas Papadakis (NTUA), Aris Dadoukis (CERTH), Donatos Stavropoulos (CERTH), Ana Juan Ferrer (ATOS), Roman Sosa Gonzalez (ATOS), Joaquin Iranzo Yuste (ATOS), Rowshan Jahan Sathi (TUB), Alex Willner (TUB), Lucas Nussbaum (Inria), David Margery (Inria), Cedric Crettaz (MI)
Reviewers	Peter Van Daele (imec)

Abstract	This deliverable gives an overview of the requirements for the developments in WP3 of the second 18 months of Fed4FIRE+. WP2 are normal operations developments (add testbeds, fix bugs, small features, etc). WP3 is focussing on larger new functionality.
Keywords	Requirements second cycle, new functionality

Document Revision History

Version	Date	Description of change	List of contributor(s)
V1	1/10/2018	TOC	Brecht Vermeulen (imec)
V2	1/02/2019	First version	Brecht Vermeulen (imec), Wim Van der Meerssche (imec), Thijs Walcarius (imec), Radomir Klacza (SU), Albert (Yiu Quan) Su (SU), Pauline Gaudet Chardonnet (SU), Dimitris Dechouniotis (NTUA), Costas Papadakis (NTUA), Aris Dadoukis (CERTH), Donatos Stavropoulos (CERTH), Ana Juan Ferrer (ATOS), Roman Sosa Gonzalez (ATOS), Joaquin Iranzo Yuste (ATOS), Rowshan Jahan Sathi (TUB), Alex Willner (TUB), Lucas Nussbaum (Inria), David Margery (Inria), Cedric Crettaz (MI)
V3	1/03/2020	Version for submission	Brecht Vermeulen (imec)

DISCLAIMER

COPYRIGHT NOTICE

ACKNOWLEDGMENT

Co-funded by the European Union

Co-funded by the Swiss Confederation

This deliverable has been written in the context of a Horizon 2020 European research project, which is co-funded by the European Commission and the Swiss State Secretariat for Education, Research and Innovation. The opinions expressed and arguments employed do not engage the supporting parties.

D3.3: Requirements and specifications for the second cycle

Project co-funded by the European Commission in the H2020 Programme		
Nature of the deliverable:		**R**
Dissemination Level		
PU	Public, fully open, e.g. web	✓
CL	Classified, information as referred to in Commission Decision 2001/844/EC	
CO	Confidential to FED4FIRE+ project and Commission Services	

** R: Document, report (excluding the periodic and final reports)*

DEM: Demonstrator, pilot, prototype, plan designs

DEC: Websites, patents filing, press & media actions, videos, etc.

OTHER: Software, technical diagram, etc.

Co-funded by the Horizon 2020
Framework Programme of the European Union

EXECUTIVE SUMMARY

This deliverable gives an overview of the requirements for the developments in WP3 during the second 18 months of the project. WP2 are normal operations developments (add testbeds, fix bugs, small features, etc). WP3 is focussing on larger new functionality.

WP3 consists out of the following tasks, which are also the sequence of sections in this deliverable:

- Task 3.1 is focussing on SLA and reputation for testbed usage
- Task 3.2 is focussing on Experiment-as-a-Service (EaaS), data retention and reproducibility of experiments
- Task 3.3 is targeting Federation monitoring and interconnectivity
- Task 3.4 works on Service orchestration and brokering
- Task 3.5 researches ontologies for the federation of testbeds

The following extra developments have been made based on demands of users, tool developers and testbed owners:

- New user account portal with OAuth
- A tool (for the Fed4FIRE.eu website) to allow to more easily chose testbeds
- An automated setup for openstack with Fed4FIRE tools

Co-funded by the Horizon 2020
Framework Programme of the European Union

TABLE OF CONTENTS

Co-funded by the Horizon 2020
Framework Programme of the European Union

LIST OF FIGURES

Co-funded by the Horizon 2020
Framework Programme of the European Union

LIST OF TABLES

NO TABLE OF FIGURES ENTRIES FOUND.

Co-funded by the Horizon 2020
Framework Programme of the European Union

1 NEW USER ACCOUNT PORTAL

The main requirements that we have put forward for the new user account portal are the following:

- Better Fed4FIRE branding and look&feel
- More user friendly portal (e.g. to invite people to a project or for student classes)
- Better and more clear flow for approval of terms and conditions and GDPR terms
- Possibility to use edugain login for academics (=university home account)
- Extra information is gathered for the user accounts and is put in the user credential to make it possible for testbeds to allow more fine-grained access
- Logging for auditing
- Better statistics are possible on the usage of the testbeds
- OAuth API to make it easy for other (web-based) services to use the same account base

Co-funded by the Horizon 2020
Framework Programme of the European Union

2 TESTBED CHOSER FOR FED4FIRE.EU WEBSITE

When people come to Fed4FIRE and try to find the right testbed for their experiment, or want to check if they can use Fed4FIRE for their experimentation, they use now https://www.fed4fire.eu/testbeds/.

This consists of a map with color-coded testbed labels, based on 7 testbed types (wired, wireless, 5G, IoT, openflow, cloud, big data). Further on, each testbed has a more detailed description with a link to a specific documentation website and contact email.

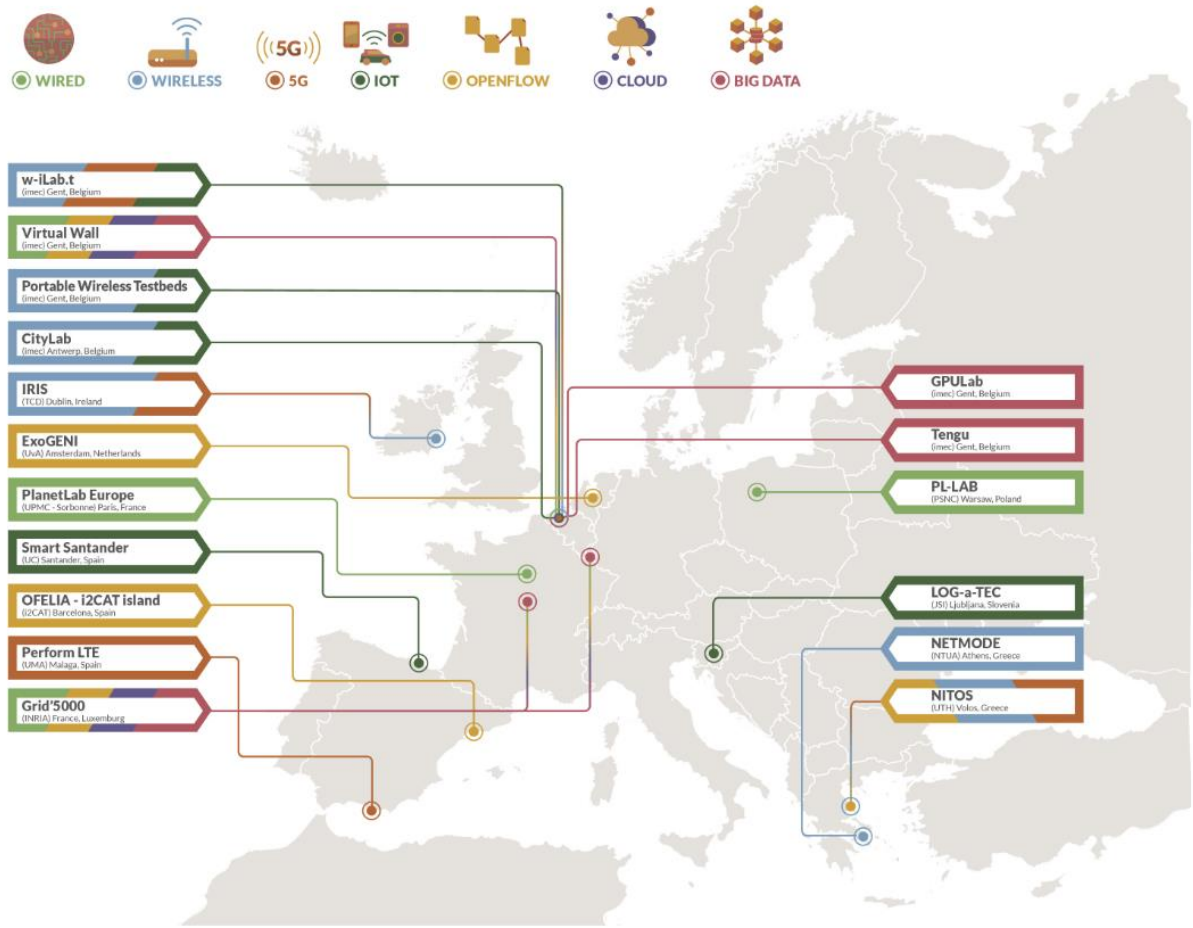

Figure 1: Current testbed map with technology labelling

However, this is too limited for some visitors/possible users. If you are looking e.g. for GPUs or for LTE or for IoT IPv6, you have to basically run through too many testbed descriptions.

Based on this simple question, it seems that ontologies could answer this, but then you need a detailed ontology for all testbeds and the queries need to be quite detailed. So we are looking for something in between: not too complex to set up and maintain (adapt to user questions), helps users, and only needed for the website for now.

Co-funded by the Horizon 2020
Framework Programme of the European Union

3 SLA AND REPUTATION SERVICE

In the Fed4FIRE+ environment, the Service Level Agreement (SLA) and the Reputation Service provide the necessary tools and mechanisms for delivering to the users a quantitative view of the trustworthiness of the federated testbeds. This service facilitates the Fed4FIRE+ users to select the appropriate testbed in the federation according to their experiment's requirements.

The aim of adding SLA within Fed4FIRE+ is to enable testbed providers to create offerings that experimenters can accept establishing an agreement with the testbed owner. We can understand the agreement as a contract between the platform providers and the testbed users. Once the agreement has been created it must be verified that it is being fulfilled. The information related to the execution of an experiment, i.e., if there is an agreement violation, will be send to the other components using a notification / subscription pattern.

The Reputation Service of Fed4FIRE+ aims to enhance and extend the already-developed reputation service of Fed4Fire project. The updated service will leverage Quality of Service (QoS) metrics, such as Availability, Latency etc., Quality of Experience (QoE) metrics, e.g., Usability and Documentation Readability, and SLA data in order to compute the degree of confidence of both experimenters and testbed. At the end of an experiment, the users will be prompted to give their feedback for the reserved testbeds in order to update the reputation score of the testbed and the credibility score of the experimenter. This process mitigates the effect of abnormal or malicious evaluations and guarantees that the testbeds' reputation score is fairly computed.

Co-funded by the Horizon 2020
Framework Programme of the European Union

3.1 SLA AND REPUTATION SERVICE ARCHITECTURE

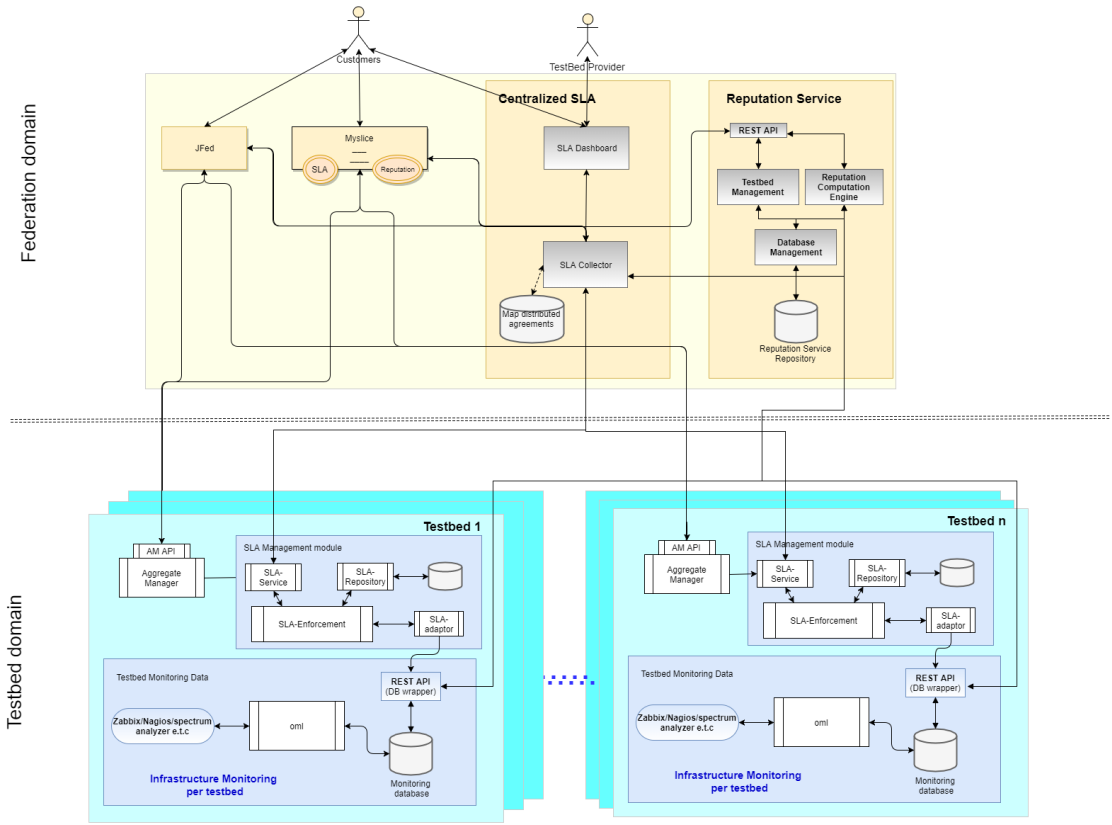

Figure 2: SLA & Reputation Service Architecture

Figure 2 shows the architecture of the SLA and Reputation service within the Fed4FIRE+ project. In the *Federation domain* we have the components that will be centralized and only instantiated once in Fed4FIRE+. In the *Testbed domain* we have the software components that will be installed at each platform and therefore instantiated multiple times in the Fed4FIRE+ project.

3.2 REQUIREMENTS FOR REPUTATION FOR SECOND CYCLE

During the first cycle, the testing and execution of the experiment evaluation lifecycle was conducted through REST API calls with the Reputation Service frontend tools. One of the key features of the HRS is the credibility mechanism which allows the Reputation Service to reduce the impact of malicious users in the computation of the reputation score of each testbed. Through the testing phase, the credibility mechanism was considered inadequate in some cases. In particular, when the monitoring data value was below the SLA constraints, if the user's rating was far below the monitoring data, the credibility value was modified and in some case heightened irrationally, since the SLA agreement was breached, and the user has every right to be negative about the testbed's performance. Such misconceptions of the original credibility algorithm have to be corrected, and in the second cycle a new credibility algorithm will be developed to replace the previous one in HRS.

Co-funded by the Horizon 2020 Framework Programme of the European Union

3.3 REQUIREMENTS FOR SLA FOR SECOND CYCLE

The above architecture and the underlying components should fulfil the functional requirements listed below.

ID	Title	Coverage in Second cycle
SLA_01	SLA solution must cover the whole lifecycle specified in WS-Agreement	An initial workflow for the whole lifecycle has been proposed. After discussing with testbed owners, an improved workflow has come out, which will be developed in the third cycle.
SLA_02	SLA solution Subscription mechanism	Delayed to 3rd cycle
SLA_03	SLA Dashboard	Dashboard implemented. Pending integration with federation tools.
SLA_04	Agreement creation and enactment	
SLA_05	Include node information in violation	
REP_01	Reputation Service access to monitoring and SLA data	An initial unified solution is discussed using the new federation portal. An early integration of Imec's testbeds is ongoing.

Co-funded by the Horizon 2020
Framework Programme of the European Union

4 REPRODUCIBILITY OF EXPERIMENTS

The Experiment Specification (ESpec) was developed in cycle 1 as a new standard for setting up experiments. It combines various existing industry standards and leverages them to make it easier to fully setup an experiment: from requesting and provisioning the necessary testbed resources to installing software, doing the configuration management and the application deployment. This is also documented towards the users at https://jfed.ilabt.imec.be/espec .

In this way, the ESpec can be used as a base for creating “Experiments-as-a-Service”, where we provide experimenters with fully automated experiments that provide an excellent starting point for doing their scientific research or education activities. See e.g. automated openstack deployment in this deliverable and other examples in D3.2.

This ESpec helps in provisioning an experiment, but is not meant/usable for the experimentation orchestration itself.

In Fed4FIRE we had OMF[1] (https://www.rubydoc.info/github/mytestbed/omf/file/README.mkd) as an experiment orchestration tool, but it is currently not supported anymore (e.g. the original website does not respond for the last couple of years: http://omf.mytestbed.net/projects/omf/wiki/Introduction.

However, there is a demand for an experiment orchestration tool, especially for wireless experiments at scale. Experimenters want to be able to turn on and off things, change parameters at fixed times, for a large number of nodes. We will start development of a lightweight tool for this in the second cycle.

[1] Thierry Rakotoarivelo, Maximilian Ott, Guillaume Jourjon, and Ivan Seskar. 2010. OMF: a control and management framework for networking testbeds. SIGOPS Oper. Syst. Rev. 43, 4 (January 2010), 54–59. DOI:https://doi.org/10.1145/1713254.1713267

Co-funded by the Horizon 2020
Framework Programme of the European Union

5 YOUREPM – SERVICE ORCHESTRATION

The objective of the Fed4FIRE+ YourEPM is to provide a tool that allows experimenters to design and execute business processes modeled in BPMN, so experimenters can design cross-testbed processes or execute 3rd parties' services. For a complete description of the orchestration solution, refer to D3.1.

The list of developments for the second cycle is:

- Integrated version with latest improvements, including enhanced multitenancy;
- Support for F4F OAuth2 authentication, instead of a certificate-based mechanism. Regular and admin users are supported. Additionally, we will define a super admin user, capable of reviewing all defined workflows in the platform.
- To install YourEPM in the federation.

5.1 REQUIREMENTS

The requirements defined for the component are presented in the table below.

ID	Title	Coverage in Second cycle
YourEPM-03	Assistant in the selection of the exposed services.	Work in progress
YourEPM-08	Integration with the authentication and authorization of Fed4FIRE+	YourEPM authorizes user by using OAuth2 service

Co-funded by the Horizon 2020
Framework Programme of the European Union

6 AUTHENTICATION PROXY SERVICE

The Fed4FIRE authentication proxy (see also D3.1 and D3.2) allows service providers, authenticated by Fed4FIRE+ credentials, to declare new HTTP or HTTPS endpoints (services or testbeds) to be made accessible to any authenticated Fed4FIRE+ user. It is a self-service service for testbed or service providers.

The service or testbed provider must configure his endpoint so that the proxy can forward requests to it. This can be done by different means: by configuring his firewall at the endpoint or by uploading credentials to the proxy. In effect, this implies trusting the proxy to only transmit requests originating from authenticated Fed4FIRE+ users.

Indeed, Fed4FIRE+'s architecture is based on the SFA framework, where users identify themselves with certificates and authenticate themselves by using these certificates to secure an TLS link to service providers. While this architecture has a lot of good properties, it is not a good fit to consume services implemented using a REST or SOAP architecture, very common in today's Internet. Inria's work for this component has consisted in the implementation of a gateway, the Fed4FIRE authentication proxy, to bridge the gap between Fed4FIRE+ credentials and services.

6.1 REQUIREMENTS

The requirements defined for the component are presented in the table below.

ID	Title
1	Expose HTTP servers (exposing APIs or websites) only to authenticated Fed4FIRE+ users through a self-service authentication proxy
2	Allow HTTP server owners to control who can access their server through the authentication proxy
3	Transmit information extracted from the user certificate (urn, authority urn and email) to HTTP server accessed through the authentication proxy through HTTP headers
4	Allow HTTP server owner to share control of the configuration of their service as declared in the authentication proxy to members of a Fed4FIRE group
5	Allow HTTP server owners to provide a login URL to be used by the authentication proxy if a login process is required to access parts of their server
6	Allow HTTP server owners to secure the link between the authentication proxy and their server using a login and a password
7	Allow HTTP server owners to secure the link between the authentication proxy and their server using a TLS client certificate
8	Allow HTTP server owners to secure the link between the authentication proxy and their server by declaring the authority of their server's certificate

Co-funded by the Horizon 2020
Framework Programme of the European Union

9	Allow declaration of a new HTTP server accessible through the authentication proxy through an API
10	Allow complete configuration of the proxy service to an HTTP server accessible through the authentication proxy through an API

Co-funded by the Horizon 2020
Framework Programme of the European Union

7 CENTRAL BROKER

The Central Broker (see also D3.1 and D3.2) acts as an overarching service that can be utilized by the experimenters to discover resources that span the federation and fulfil their experimentation requirements. More specifically, the Central Broker receives as input general specifications from the users related to date and time they want to run their experiment, technologies involved like Wired, Wireless, 5G, IoT, OpenFlow, Big Data, hardware specifications of the nodes like number of CPUs, amount of RAM and storage and finally number of nodes that denote the scale of the experiment willing to conduct in Fed4FIRE. Upon receiving these requests, the Central Broker filters the available resources of the federation and maps the experimenters' requests to the actual testbed resources, providing as output a proposed set of resources to the experimenter, who in his turn decides whether to request/reserve these resources for his experiment.

To this end, the Central Broker needs to keep an up-to-date inventory of the available resources of all the testbeds that are part of the federation. The updated inventory allows the service to apply its mapping algorithm that take into consideration the user's requirements along with the characteristics of the available resources and provide the results without any significant delays.

7.1 REQUIREMENTS

The requirements defined for the Central Broker are presented in the table below.

ID	Title
1	Retrieve resource availability of all the testbeds of the federation.
2	Keep an up-to-date inventory of the available resources of all the testbeds that are part of the federation.
3	Provide a method that enables experimenters to express their resource requirements in a future proof extensible way.
4	Support all the different testbed technologies like Wireless, Wired, 5G, Cloud, IoT, Big Data and OpenFlow in the resource mapping algorithm.
5	Respond with a set of available resources that meet the experimenter's requirements.

Co-funded by the Horizon 2020
Framework Programme of the European Union

8 ONTOLOGIES

For this second cycle we schedule to develop an Aggregate Manager based on ontologies. The Aggregate Manager is the component which talks to a testbed to provision resources and which has an API that be used by experimenter tools.

Together with this Aggregate Manager, we will also define a semantic based resource description to describe the resources.

8.1.1 Functional Requirements

Semantic Based Resource Descriptions Requirements

ID	SRD_01
Title	**Semantic Modelling of Experiment Management Concepts**
Short description	All concepts participating in the experiment lifecycle should be modelled and linked using the appropriate semantics. These concepts include F4F+ wireless testbeds and their respective resources utilizing the already developed ontology of OMN suite. The aforementioned heterogeneous concepts should be described in a formalized manner to build a basis for their management based on their semantics, i.e. their underlying meaning and relations.
Relevant Use Case(s)	Discover available resources leveraging semantic descriptions; Discover available resources based on criteria; Discover available resources based on inferred knowledge; Retrieve information about some testbed resources & reservations; Create semantic descriptions representing testbed's resources; Validate experiment scenarios based on semantic resource descriptions
Type	“Data”
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	SRD_02
Title	**Semantic Aware Support of Experiment Lifecycle**
Short description	The system should be able to support the experiment’s lifecycle while leveraging semantically enriched resource descriptions. In this context, users should be able to discover, reserve, provision and release the testbed’s semantically modelled resources. These are basic functionalities upon which more intelligence can be built. This requirement is of significant importance as it will facilitate the federation-wide interoperability.
Relevant Use Case(s)	Discover available resources leveraging semantic descriptions; Discover available resources based on criteria; Discover available resources based on inferred knowledge; Book resources suitable for the experiment's requirements; Book resources in advance; Check the status a pending/ongoing reservation; Renew an ongoing reservation; Cancel an ongoing reservation and release the respective resources; Access historical data of past reservations
Type	“Functional”

Co-funded by the Horizon 2020 Framework Programme of the European Union

Priority Level	Mandatory
Identified by Partner(s)	NTUA
ID	SRD_03
Title	Semantic Aware Support of Administrative Tasks
Short description	The system should be able to support the administrative tasks while leveraging semantically enriched resource descriptions. In this context, administrators should be able to retrieve, create, update and delete the testbed's semantically modelled resources.
Relevant Use Case(s)	Retrieve information about some testbed resources & reservations; Retrieve information based on criteria; Create semantic descriptions representing testbed's resources; Modify descriptions and ongoing reservations; Delete descriptions; Access historical data of past reservations
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	SRD_04
Title	**Reasoning & Knowledge Inference**
Short description	The system should be able to apply logical rules to the stored knowledge base and deduce new (inferred) knowledge. The integration of a reasoning engine, capable of the aforementioned behaviour is essential for the F4F+ project to exploit the benefits of semantically modelled data.
Relevant Use Case(s)	Discover available resources based on inferred knowledge; Validate experiment scenarios based on semantic resource descriptions
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

Co-funded by the Horizon 2020
Framework Programme of the European Union

SFA Based Testbed Management & Federation Requirements

ID	TMF_01
Title	**SFA Based User Authentication & Authorization**
Short description	Access to resources within a testbed should not be provided to unauthenticated users. Based on the trusted identity of a user and other attributes that will be provided, specific authorization policies should take effect. Adopting SFA, unified authentication and authorization should be based on X509 certificates. The federation Registry (and corresponding APIs) shall be able to authenticate users and provide them with the necessary credentials to book resources.
Relevant Use Case(s)	Authenticate using x509 certificates
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	TMF_02
Title	**SFA Based Support of Experiment Lifecycle**
Short description	The semantic aggregate manager should expose the appropriate API, providing methods that facilitate resource discovery, booking and reservation, resource provisioning and release, thus maintaining compatibility and enabling interoperability with existing SFA enabled infrastructures.
Relevant Use Case(s)	Discover available resources using SFA enabled tools; Book resources suitable for the experiment's requirements; Book resources in advance; Access historical data of past reservations; Check the status a pending/ongoing reservation; Renew an ongoing reservation; Cancel an ongoing reservation & release the respective resources
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	TMF_03
Title	**Support of XML based RSpec documents**
Short description	In order to successfully integrate SFA, the Resource Specification (RSpec) XML documents should be adopted, alongside semantics, as a common language for describing resources, resource requests and reservations.
Relevant Use Case(s)	Discover available resources using SFA enabled tools; Book resources suitable for the experiment's requirements; Check the status a pending/ongoing reservation; Renew an ongoing reservation; Cancel an ongoing reservation and release the respective resources
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

Co-funded by the Horizon 2020 Framework Programme of the European Union

Interoperability Requirements

ID	INT_01
Title	**Interoperability with SFA enabled Provisioning Tools**
Short description	The semantic aggregate manager should expose the appropriate version of the GENI API in order to facilitate interoperability with the well-known SFA enabled provisioning tools (e.g. MySlice, jFED, omni).
Relevant Use Case(s)	Authenticate using x509 certificates; Discover available resources using SFA enabled tools; Book resources suitable for the experiment's requirements; Check the status a pending/ongoing reservation; Renew an ongoing reservation; Cancel an ongoing reservation and release the respective resources
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	INT_02
Title	**Parallel Use of Semantic & SFA Based Resource Descriptions**
Short description	The F4F+ testbeds should be able to become federated with existing testbeds that support SFA and use XML based RSpecs, while at the same time, the federation should exploit the benefits of a semantic web approach, (i.e data semantics). Translation mechanisms should be adopted in order to facilitate alternation between those data formats.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

Co-funded by the Horizon 2020
Framework Programme of the European Union

General Requirements

ID	GEN_01
Title	**Scheduling of Bookings**
Short description	The system should enable users to book resources in advance. Each available F4F+ wireless resource should be able to become reserved for a period of time defined in a lease tag during the request.
Relevant Use Case(s)	Book resources in advance
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	GEN_02
Title	**Storage of Historical Data**
Short description	Users should be able to access historical data of their entire past executed actions and experiments.
Relevant Use Case(s)	Access historical data of past reservations
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	GEN_03
Title	**Asynchronous Request Processing**
Short description	The system should support asynchronous request processing for the faster completion of time-consuming tasks and the concurrent serving of multiple users.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	GEN_04
Title	**Data Consistency**
Short description	Presented data to the users must be valid and correspond to the current status of their resources and experiments.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	"Functional"
Priority Level	Mandatory
Identified by Partner(s)	NTUA

Co-funded by the Horizon 2020 Framework Programme of the European Union

8.1.2 NON-Functional Requirements

Usability

ID	US_01
Title	**Consistent User Experience**
Short description	The semantic aggregate manager should provide consistent user experience.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	“Usability”
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	US_02
Title	**User-friendly Installation**
Short description	Minimal and easy installation of the semantic aggregate manager on top of the testbed’s existing infrastructure.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	“Usability”
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	US_03
Title	**User-friendly Configuration and Usage**
Short description	The configuration process and use of the semantic aggregate manager should be easy, intuitive and reliable.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	“Usability”
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	US_04
Title	**Comprehensive Platform and API Documentation**
Short description	The semantic aggregate manager and API documentation should be available, comprehensive and consistent with current functionality.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	“Usability”
Priority Level	Mandatory
Identified by Partner(s)	NTUA

Co-funded by the Horizon 2020 Framework Programme of the European Union

Reliability

ID	REL_01
Title	**Reliable Service Infrastructure**
Short description	Ensure that the semantic aggregate manager and the relevant services are available at all times.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	“Reliability”
Priority Level	Mandatory
Identified by Partner(s)	NTUA

ID	REL_02
Title	**Reliable data, context and content**
Short description	Ensure that context information is as accurate as possible.
Relevant Use Case(s)	This requirement is implied in most use cases where user actors are involved.
Type	“Reliability”
Priority Level	Mandatory
Identified by Partner(s)	NTUA

Co-funded by the Horizon 2020
Framework Programme of the European Union

9 AUTOMATED OPENSTACK DEPLOYMENT

Because of a high user demand for an easy to set up Openstack environment, we added this requirement for the 2nd cycle. Development of the necessary scripts to deploy automatically a flexible openstack environment in the Fed4FIRE environment.

We can use existing frameworks, such as EnOS (https://enos.readthedocs.io/en/stable/) but need to adapt this to the Fed4FIRE environment and tools.

Co-funded by the Horizon 2020
Framework Programme of the European Union

10 CONCLUSIONS

In this deliverable we identified the (user) requirements for the developments in WP3.

Co-funded by the Horizon 2020
Framework Programme of the European Union