SEVENTH FRAMEWORK PROGRAME

Specific Targeted Research Project

Project Number:	**FP7–SMARTCITIES–2013(ICT)**
Project Acronym:	**VITAL**
Project Number:	**608682**
Project Title:	**Virtualized programmable InTerfAces for innovative cost-effective IoT depLoyments in smart cities**

D5.3.1 Smart Governance Toolkit V1

Document Id:	VITAL-D531-251015-Draft
File Name:	VITAL-D531-251015-Draft.pdf
Document reference:	Deliverable 5.3.1
Version :	Draft
Editor :	Fotis Stamatelopoulos, Angelos Lenis
Organisation :	SILO
Date :	25 / 10 / 2015
Document type:	Deliverable
Security:	PU (Public)

DOCUMENT HISTORY

Rev.	Author(s)	Organisation(s)	Date	Comments
V01	A.Lenis, F.Stamatelopoulos	SiLo	06/08/2015	Document Structure, Scope & Contributors
V02	A.Lenis, F.Stamatelopoulos	SiLo	04/09/2015	Section 2 updated with main requirement and architecture diagram
V03	A.Lenis	SiLo	19/09/2015	Section 3.1
V04	A.Lenis F.Stamatelopoulos	SiLo	08/10/2015	Section 3.2
V05	A. Lenis	SiLo	10/10/2015	Section 4
V06	L. Bracco, P. Dal Zovo	Reply	12/10/2015	Updates to Section 3.2 mainly.
V07	A. Lenis	SiLo	13/10/2015	Formatting changes in Sections 3,4 Added references in Section 6
V08	A.Lenis F.Stamatelopoulos	SiLo	13/10/2015	Conclusions & Outlook Section
V09	A.Lenis F.Stamatelopoulos	SiLo	15/10/2015	Quality Control and Formatting
V10	A.Lenis, F.Stamatelopoulos	SiLo	23/10/2015	Added the Governace Toolkit
V11	A.Lenis, F.Stamatelopoulos	SiLo	23/10/2015	Added End-Users interaction diagram in section 2.1
V12	L. Bracco, P. Dal Zovo	SiLo	25/10/2015	Last edits before Approval
V13	Aqeel H. Kasmi	NUIG	25/10/2015	Circulated for Approval
Draft	Martin Serrano	NUIG	25/10/2015	EC Submitted

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

TERMS AND ACRONYMS

FCAPS	Fault, Configuration, Accounting, Performance, Security
FIT	Future Internet Technology
GSN	Global Sensor Networks
GUI	Graphical User Interface
ICO	Internet-Connected-Object
IoT	Internet-of-Things
KPI	Key Performance Indicator
MIB	Management Information Base
QoS	Quality of Service
SLA	Service Level Agreements
PPI	Platform Provider Interface
VUAI	Virtual Unified Access Interface

1 INTRODUCTION

1.1 Scope

VITAL WP5 is devoted to the design, development and validation of a range of different applications that operate over the VITAL smart cities platform, which provides the means for accessing Internet Connected Objects (ICO) and their data streams, which may stem from different and independent IoT and smart city deployments. In terms of the VITAL architecture (specified in deliverable D2.3 and updated/revised in deliverable D3.2.2), WP5 implements applications that access ICO data and services through VUAIs (Virtualized Unified Access Interfaces) i.e. ICOs attached to diverse IoT platforms.

The applications of the VITAL platform that are implemented in WP5 are of three different types:

- A management plane application providing FCAPS management functionalities over sensors, ICOs, IoT platforms, IoT data streams and other managed entities that are integrated in the VITAL platform. This management plane application is offered to smart city authorities and administrators of the smart city infrastructure. It is the core subject of deliverable D5.1.
- A development tool providing a single entry point to the VITAL capabilities/services for accessing and utilizing IoT services, along with a visual environment for assembling smart city applications based on the above-listed services. An initial release of this tool has been produced as part of deliverable D5.2.
- A governance toolkit enabling the configuration and deployment of an instance of the VITAL platform to the needs of the specific city. This includes the customization of the VITAL platform to the needs of a specific city, as well as the configuration of Service Level Agreements (SLAs) with providers of IoT platforms and services. The VITAL governance toolkit has a close affiliation with the management plane application, since they both involve the configuration of VITAL components. Hence, the governance toolkit shares the same GUI (Graphical User Interfaces) with the management plane application, thus being a set of dashboards that are embedded with the management GUI presented in deliverable D5.1. The governance toolkit is the third deliverable (D5.3) of WP5 of the project.

Overall, the purpose of this deliverable (D5.3) is to present the specifications of the governance toolkit, along with its initial prototype implementation (release). The second and final version of the deliverable will provide the enhanced and the final implementation of this toolkit.

1.2 Audience

The VITAL governance toolkit aims at enabling deployers, operators and administrators of the VITAL infrastructure (including smart cities solution providers and/or the technical/ICT departments of city authorities) to configure/customize the VITAL platform to their needs. It is therefore primarily addressed to providers of turn-key IoT solutions for smart cities (notably providers leveraging the VITAL platform) and to smart city authorities (notably to their personnel that is in charge of the

platform configuration). The document could be also useful for researchers and engineers wishing to understand the scope and role of the operational parameters of a VITAL deployment.

The deliverable has also a value as an internal document for consortium members, especially those dealing with the deployment and configuration of the VITAL platform (in WP6 of the project).

1.3 Summary and Structure

This initial version of this deliverable starts with a discussion of the motivation and need behind the implementation of the Governance toolkit (and of an associated environment) as part of a smart cities platform. It also illustrates its positioning within the VITAL architecture, including the way it interacts with other modules of the VITAL platform. A significant part of the deliverable is devoted to the specification of the functionalities that are required from the Governance Toolkit. The specifications described in this deliverable will be considered as a design document for the implementation of the governance module. The deliverable is structured as follows:

- Section 2 following this introductory section outlines the drivers behind the specification and implementation of the VITAL Governance Toolkit. It also outlines the positioning of the toolkit within the overall VITAL architecture, along with the specified integration interfaces.
- Section 3 presents an initial set of functionalities to be supported & implemented by the VITAL Governance Toolkit environment. This section provides an overview of the technical architecture and design guidelines, alongside with how Vital modules will interact as to provide higher level federation services.
- Section 4 presents a mockup-based view of the VITAL Governance toolkit's functionalities, aiming to to act as a guideline during development.
- Section 5 is the concluding section of the deliverable. Apart from a summary and concluding remarks, it also provides a plan for future development.

2 Inception and Specification of the VITAL GOVERNANCE Toolkit

2.1 Main Requirements

The following table summarizes the main requirements of the Governance Toolkit. It also provides the rationale of each requirement

Governance Requirement	Rationale
Configure the VITAL platform instance in order to enable or disable specific modules	Enable the customization of the VITAL deployment according to
Configure the VITAL platform instance in order to select/deploy specific IoT systems and services	Customize the VITAL deployment instance to business requirements and SLAs with other platforms
Configure the VITAL platform instance in order to be usable for a specific location	Customize the VITAL deployment for a given city or region; Ignore data and services residing outside the scope of the region
Optimization of Bandwidth and SLAs with IoT Platform Providers.	Economize on the resources (and cost) associated with the VITAL deployment, taking into account SLAs with platform providers
Automated and "Intelligent" Security and Authorization.	Designing and enforcing security policies (i.e. authorization and authentication) from a single point.
Enforcement for Developers – Facilitating Developers.	Ensure that developers engage with data and services falling in their scope of the smart city application that they are targeting
Enforcement for Deployers – Facilitating Deployers.	Ensure that deployers engage with data and services falling in their scope of the deployment at hand.
Integration & Compatibility with VITAL Architecture	Ensure that the Governance toolkit operates over the VITAL components, while respecting their structure
Integration with Management Plane	Ensure that the management plane becomes a single-entry point for accessing the full range of management and configuration functionalities

Table 1: Main Requirements Driving the Implementation of the Governance Toolkit and Their Rationale

Note that the governance module allows parameterization at the platform level and at the application/user level. Platform level parameterization define limitations and configuration that affects the VITAL platform deployment as a whole, e.g. define the VITAL components that will be available, or specific data sources, or a geographical

area limitation. At the application/user level the governance functionality allows the definition of limitations and platform behaviour parameterization for a specific application/user, e.g. consider an mobile application that is allowed access only to a specific set of data sources within a limited geographical region (defined by a polygon) while another application deployed in the VITAL development tools module is allowed the consumption of another set of services and data sources without geographical limitations. Obviously, all applications are limited to the platform level parameterization.

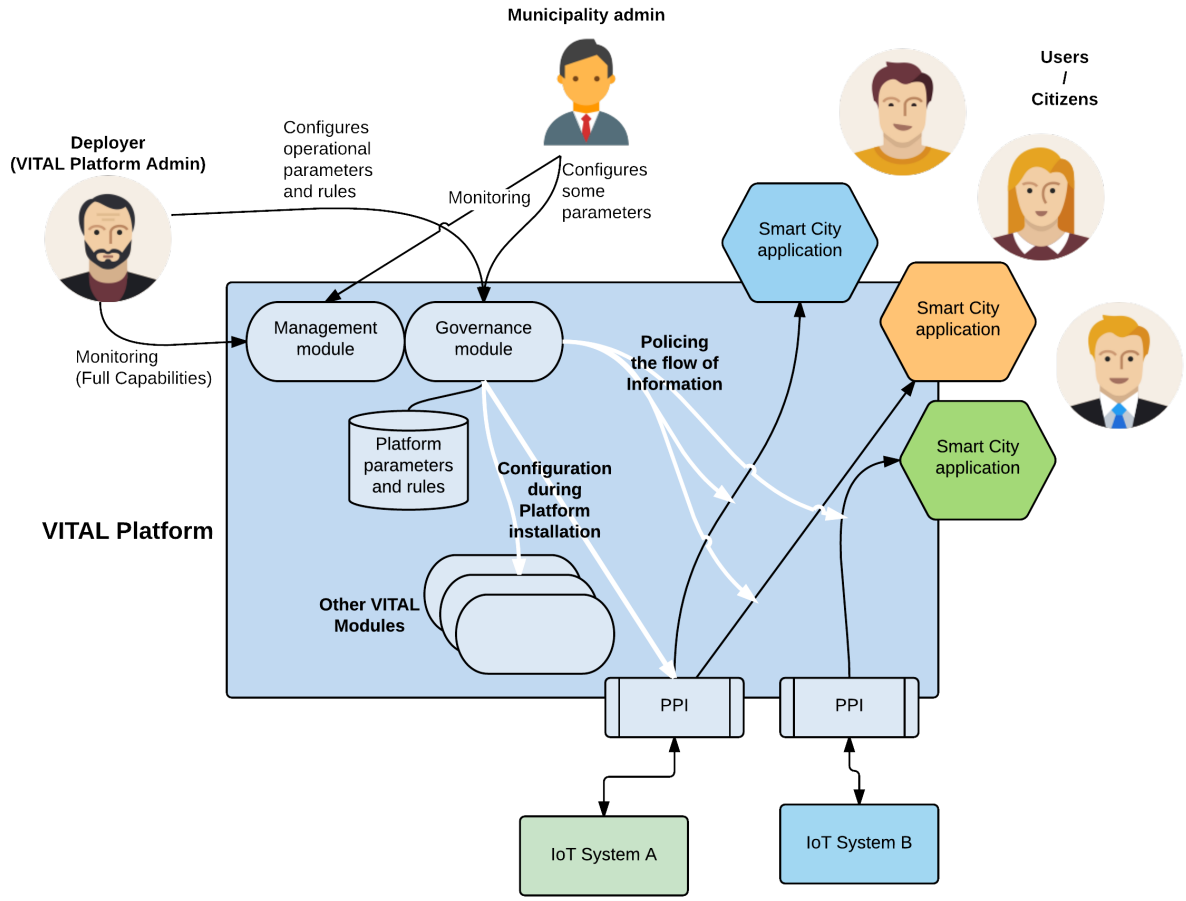

Figure 1 Governance Toolkit interactions with users

In Figure 1, the high level requirements of the Governance toolkit are displayed in relation with end-users types. The overall goal of the Governance toolkit is

(1) to enable Deployers (e.g. Vital Platform Administrators) with the capabilities to manage and configure the Vital installation as a whole - depicted with the "Configuration" arrows -,

(2) to provide the tools necessary for elevated users (e.g. Municipality Administrators) to control the flow of information between Vital Systems and specific applications through high-level policies – arrows labelled with 'Policing' – and finally

(3) Enforce these policies to applications so that these applications, the application developers and in turn the end-users (e.g. Citizens) will be exposed to the requested subsets of data and services with the desired quality of service (QoS) level.

2.2 Architecture Overview

The following diagram (Figure 2) provides an overview of the Governance Tools Module position within the VITAL architecture as well as its dependencies and association with other other VITAL modules.

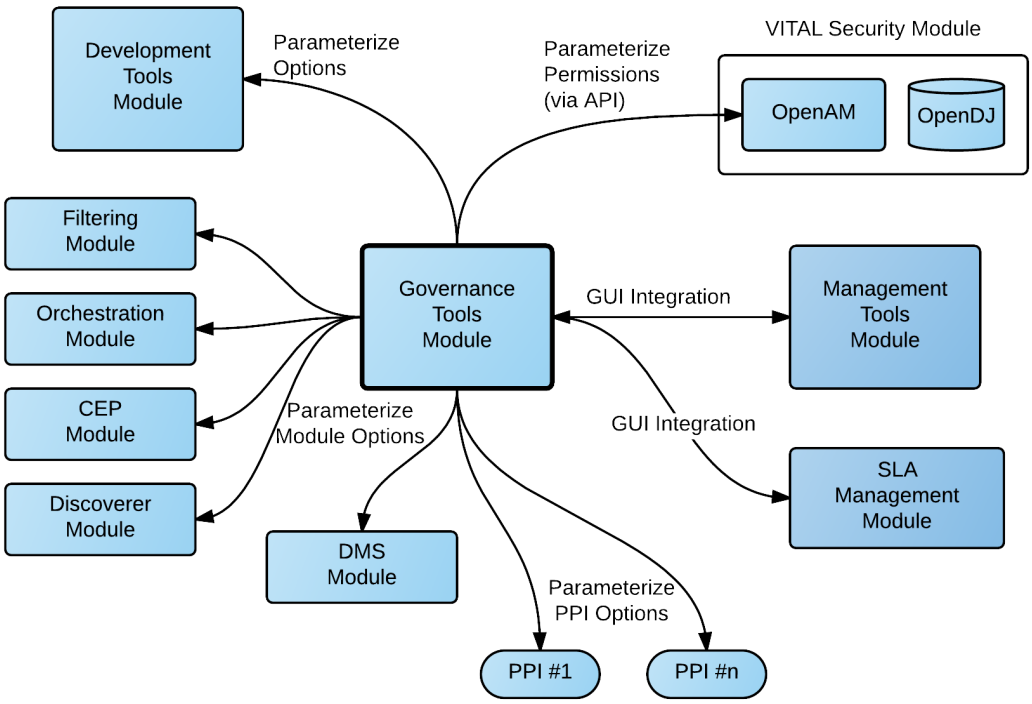

Figure 2: Governance module position & dependencies in the architecture

More specifically the Governance Tools Module is associated with the following modules:

2.2.1 Security Module

Based on an the implementation of OpenAM / OpenDJ communicates via an API for parameterizing the security and access controls options, i.e. configuring permissions to specific VITAL modules for specific applications/users. For example, the platform administrator will allow a specific user to access the CEP module while other application/user credentials will not.

2.2.2 Management Tools Module

This dependency is for GUI integration only, i.e. present the Governance Tools UI via the Management Tools UI. The Governance Tools will not parameterize the Management Tools Module functionality.

2.2.3 SLA Management Module:

- This module is also a GUI integration dependency in order to allow the user to view the SLA Management UI via the Governance Tools UI (e.g. for viewing SLA monitoring tools).

2.2.4 VITAL Services Modules:

Filtering, Orchestration, CEP, Discoverer Modules use API calls offered by the Module for parameterizing its behaviour at the platform or the user level, e.g. setup limitations on functionality supported for the specific installation or for a specific user. In the following section we specify parameterization options for each module.

2.2.5 PPI's, PADA and DMS Modules:

PPIs, PADA and DMS Modules: the Governance Tools Module allows:

The parameterization of DMS, PADA and PPIs. For example, define the communication parameters of a PPI (e.g. network parameters and credentials for connection to the specific underlying IoT system), or setup operational parameters of the DMS Module (e.g. storage limits or Virtuoso parameters).

The definition of limitations to specific IoT systems and/or data sources (type or location limits or throughput limitations) at the platform or at the user level. Definition of limitation at the platform level (i.e. affecting the operation of the current Vital platform installation as a whole) is communicated directly to the related PPIs and the PADA module, while limitations at the user level (i.e. limitations that affect a specific user) is communicated to the DMS Module. For example, if we need to limit the current installation to sensor data from a specific geographical location, or limit communication to only temperature sensor data, then we need to inform the PPIs installed to only retrieve and push to DMS the data we need. A similar example is the case of limiting the overall throughput imposed to a specific underlying IoT System (for cost reasons); in this case the PPIs must handle the policing of the overall traffic caused by the platform. On the other hand, when we need to limit a specific user to a geographical area, or types of sensors then the Governance Module will communicate this information to the DMS Module.

Note that the above logic assumes that all VITAL modules handle internally SLA policing, while the SLA Module provides the functionality for monitoring the enforcement of the SLAs. If this design assumption changes, then the Governance module will have to send the throughput limitations to the SLA Module and not to the individual modules.

3 GOVERNANCE TOOLKIT MODULES & FUNCTIONALITIES

3.1 VITAL Installation and Deployment

3.1.1 Deploying and Configuring Vital modules

The Vital platform is composed at the low-level, with PPIs that connect to data sources and IoT platforms and at the medium level with modules that provide added value functions, like the DMS or the Orchestrator module. The higher level is composed of applications that utilize the provided data and services.

Each Vital installation, i.e. a specific deployment of the Vital platform in a Smart city, needs to adapt to the city's specific requirements. Some modules may prove to be mandatory in one installation, but not in other deployments. For example, a small city might not need the functionality of the Filtering module, whereas for simple applications the meta-service functionalities of the Orchestrator module may prove to be unnecessary.

PPIs and Vital modules are all accessible through their VUAI (Virtual Unified Access Interface) implementation [Vital-D3.2.2-2015]. Part of the VUAI specification is the management related set of services and most notably the ConfigurationService. This service, as defined in [Vital-D5.1.2-2015], allows higher-level components like the Management Platform or the Governance toolkit to identify available configuration parameters and update them to match the requirements.

Through the Configuration Service, the Governance Toolkit will be able to:

1. Enable/Disable specific modules.
2. Parameterize each module.

The Vital platform will need to react to these updates, so that higher level application may reflect what is available and how to use it. For example, once disabling a Vital module, the DMS and Service Discovery modules will need to refresh their list of available services. In turn, the Management platform will remove this service from the Topology / Dashboard view.

3.1.2 Selecting Data Sources

A crucial part of a Vital installation, is to identify and connect to specific data sources and / or IoT platforms. The differentiation between data sources and platforms is mostly conceptual and differ only in the type of data they provide. For example, we consider as data source a Web Service that provides information on Bus routes in a city, whereas an IoT platform is a more advanced component that provides observations from sensors like traffic sensors or weather stations.

As far as Vital is concerned, all available information comes from PPI deployments. Selection of sources of information is then degraded to actually selecting which PPIs the Vital platform will exploit. This is a two step approach:

(1) Identify all available PPIs and the type of information they provide

(2) Configure the PADA module – that is responsible for gathering and pushing data to DMS - to connect only to specific PPIs

Step one, requires access to a list of available PPIs in the system. This can be provided from the PADA module itself or can be a query to DMS.

Step two, requires the PADA module to implement the ConfigurationService of the VUAI interface, with appropriate parameters to add/remove PPIs.

3.1.3 Enforcing City Boundaries

An additional requirement, after selecting PPIs is to filter the data flow to more specific subsets. This is not to be confused with the Filtering Service of Vital, as it is a lower-level configuration option, that actually enforces limitations on the data availability at a platform level. The properties that the governance toolkit will support are:

(1) Location based boundaries: Limit the flow of information only to a specific geographical area.

(2) Observation type boundaries: Limit observations only to specific types.

A PPI may provide data related to Weather and Traffic conditions and for a wide area. The Governance module will be able to handle a requirement it use only traffic related information and for a specific sub-area. This will affect performance and storage requirements, as there will be no need to actually retrieve/store and expose unnecessary information to the Vital platform, only to filter it at an application level.

The Governance toolkit will provide the tools to select an area and the observation type through a list of what is available. When the administrator finalizes her selection the toolkit will connect to the appropriate modules, mainly PPIs and PADA and update their configurations.

Other Vital modules will need to react to this change and update their cache of available information.

3.2 Vital Application / User Federation

While the previous functionalities were associated with the deployment of a new Vital installation on a new environment, the next set of functions affect the runtime performance of the platform as it is perceived by end users and applications. Applications utilizing the Vital platform are software agents that can use credentials/granted tokens like the physical Users, and, similarly to physical users, belong to Groups, used to represent roles and to grant adequate permissions.

3.2.1 Security Policies

In the VITAL application domain, a variety of data from heterogeneous source, demanding different levels of protection shall be made available/accessible to intended users only, directly or through aggregations, compositions and processing functions.

Access governance, i.e. governing who has access to what, where control of access is driven by policies, can be relevant in many scenarios. It shall allow configuring access control as needed to satisfy related requirements of various VITAL stakeholders, spanning from Service Providers, Solution Providers, Citizens and Businesses within the Smart Cities, and City Authorities.

The security module of Vital, with its support for User policies, provides the infrastructure for associating Users with specific Systems and Services.

Policies are a set of rules, allowing or denying access to specific URL endpoints or to URL enpoints matching specified patterns. Users are associated with groups and groups with policies. For example:

Table 2: Examples of Security Policies

User	Group	Rules	Comments
Administrator	Group of Administrators	'*'	Allow access to all
Camden Application	Group of Camden City	http://vital.dms/* http://vital.orchestrator/* http://vital.servicediscovery/* http://vital.ppi.camden.humidity/* http://vital.ppi.camden.timetables/* ….	Access to three Vital services and only Camden related data-sources (PPIs)
PADA_user	Vital_PPIs	http://vital.ppi.*	Has access to all Vital PPIs

These low-level policies associate users with specific services described as endpoint URLs. While this infrastructure is the core of the security system, on a large scale it will be difficult to identify manually all the URLs endpoints for specific scenarios.

Currently, a "Guided setup" view (shown in Figure 9) has been introduced, to simplify the initial configuration of security users and groups based on common needs. In the future releases, the Governance Toolkit will support higher level policies, and will translate them to the low level ones, that the Security module can understand.

For this, it will exploit the capabilities of VITAL services, which can provide relevant information such as those need to apply higher level policies based on:

- Limit User access to specific Vital modules based on
  - Application Specifications
  - Performance metrics
  - SLA Parameters
- Limit User access to specific Systems based on
  - Location preferences
  - Measurement Types
  - Performance metrics
  - SLA Parameters

The Governance toolkit will need to automatically identify the specific list of URL endpoints in order to satisfy the higher level policy descriptions. These capabilities are of course dependent on the metadata descriptions, performance metrics and SLA metrics measures and provided by the systems themselves. The mechanisms for metadata descriptions are described in [Vital-D3.2.2-2015] and for performance and SLAs in [Vital-D5.1.2-2015].

A desired expansion of the above mechanism, is to express policies not only at the URL/service level, but also the data type level.

3.2.2 SLA Monitoring and Management

SLAs, are contracts between applications and service providers, denoting the accepted level of service quality that the application requires. SLAs are composed of a set of metrics and their desired thresholds. For Vital, a definition of System SLAs is documented in [Vital-D5.1.2-2015]. It contains a core set of metrics and an extension to the VUAI interface, on how to retrieve their values.

Examples of metrics for systems are:

- Uptime availability
- Max. Number of Request
- Response time
- Max. Number of Requests for a specific user / application
- Mean time to restore

The Governance Toolkit will retrieve the available information from each system and will generate meta-information on:

- SLA metrics per system / module
- SLA metrics for applications

The first one, is pretty straight-forward based on the VUAI definitions. For each system connected, the toolkit will utilize the already defined SLA operations of the vital:MonitoringService (GetSupportedSLAParameters, GetSLAParameters) and collect the necessary information. The result will be (1) displayed on a monitoring screen (2) on the policy definition module.

The application level SLA metrics is a more complex scenario. An application SLA is composed of a set of requirements from more than one systems in the Vital platform. The Governance toolkit, will need to provide a module that combines low level system SLA metrics to high level application SLA metrics. The end result will indicate if the application receives the expected service quality from the Vital platform.

4 IMPLEMENTED FUNCTIONALITIES AND MOCK-UPS

Similarly, to the implementation of the VITAL Management platform, an incremental top-down approach will be also adopted for the implementation of the VITAL Governance Toolkit.

This approach is based on the segmentation of the toolkit to a number of components that enable configuration of available modules. As each module defines concrete parameterization options, the Governance Toolkit will integrate them and utilize them to achieve its purpose of configurable installation, application policing and SLA management. This incremental approach will be reflected in the successive (and final) release of the present deliverable. Indeed, the present release illustrates high level requirements of the toolkit, while the latter releases of the deliverable will elaborate on implementation and integration scenarios with all configurable entities.

In addition to incremental, the approach taken towards implementing the prototypes accompanying this deliverable is a top-down approach. It starts with the establishment of a general environment for the governance of a Vital installation, which will be refined, detailed and customized to the needs of specific sets/bundles of managed entities, as more modules are integrated in the overall architecture.

4.1 Deployment and Configuration

The following figures, display the design goals for the Deployment and Configuration stage of the Vital Platform. The process is comprised of a series of steps: (1) module selection (2) data-source selection (3) data-type selection.

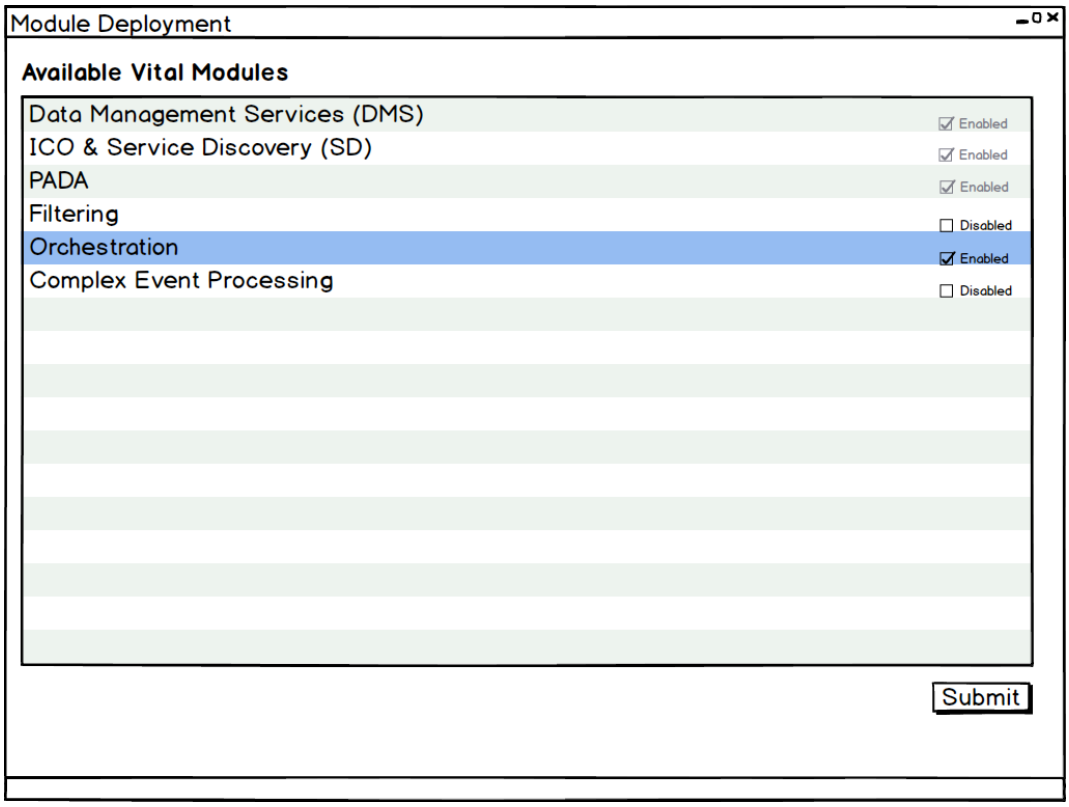

Figure 3: View of available modules, and option to enable/disable the optional ones

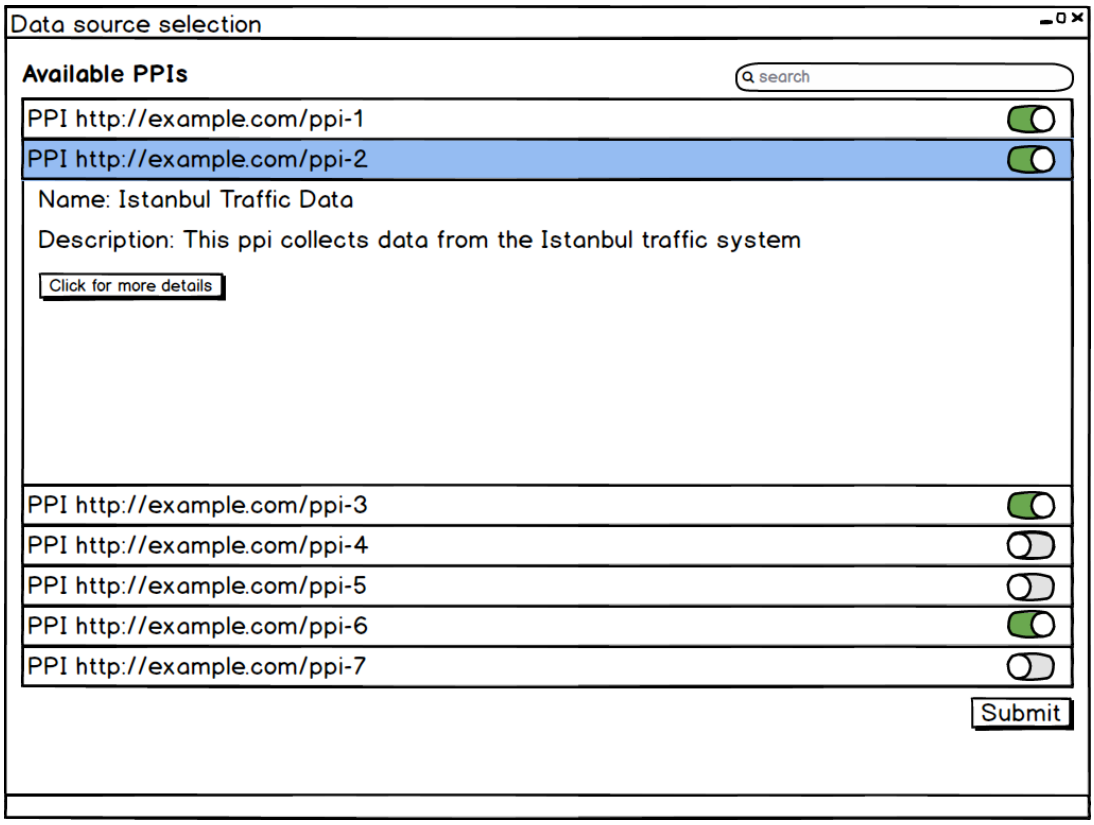

Figure 4: View of available data sources (PPIs) with the option to select the desired ones

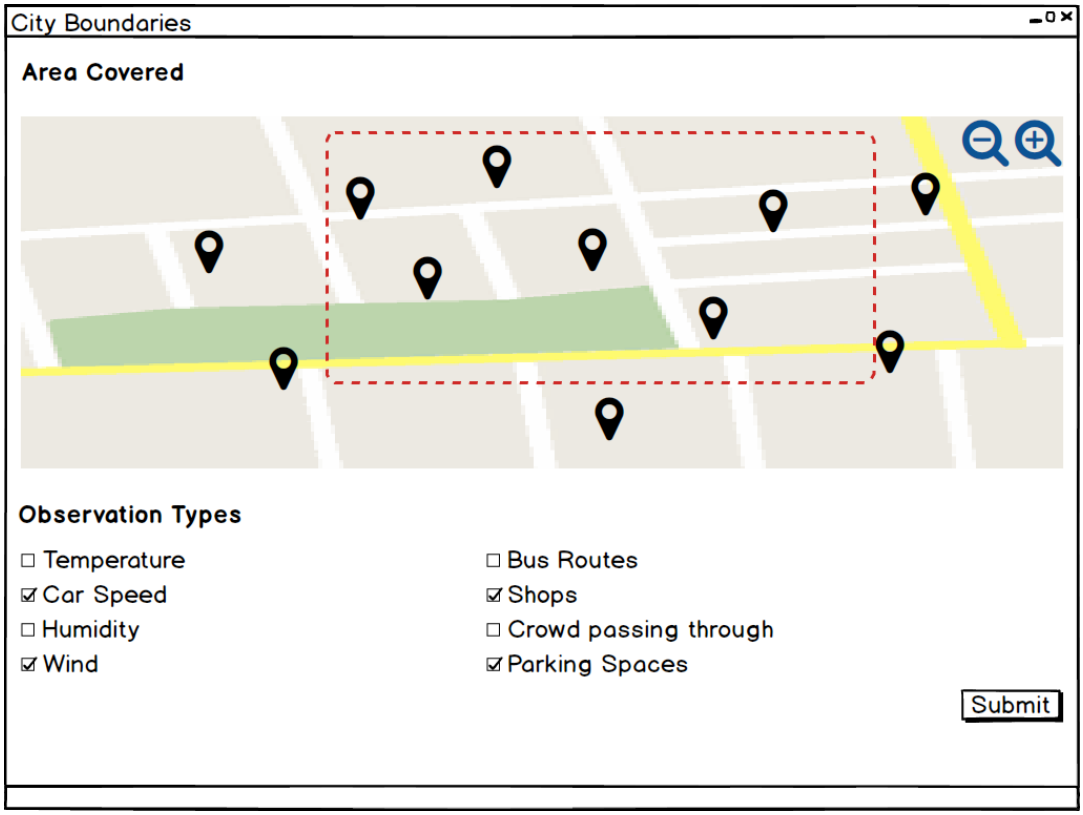

Figure 5: View of data source selection based on location and observation types

4.2 Application / User Federation

The following figures, display the design goals for the Application Federation. The Vital platform allows on runtime to configure Groups of users (Applications) with access to specific resources in the form of high level policies. In parallel, it provides application and system observers with dashboards containing the quality of service for the specific application, as measured by the governance toolkit.

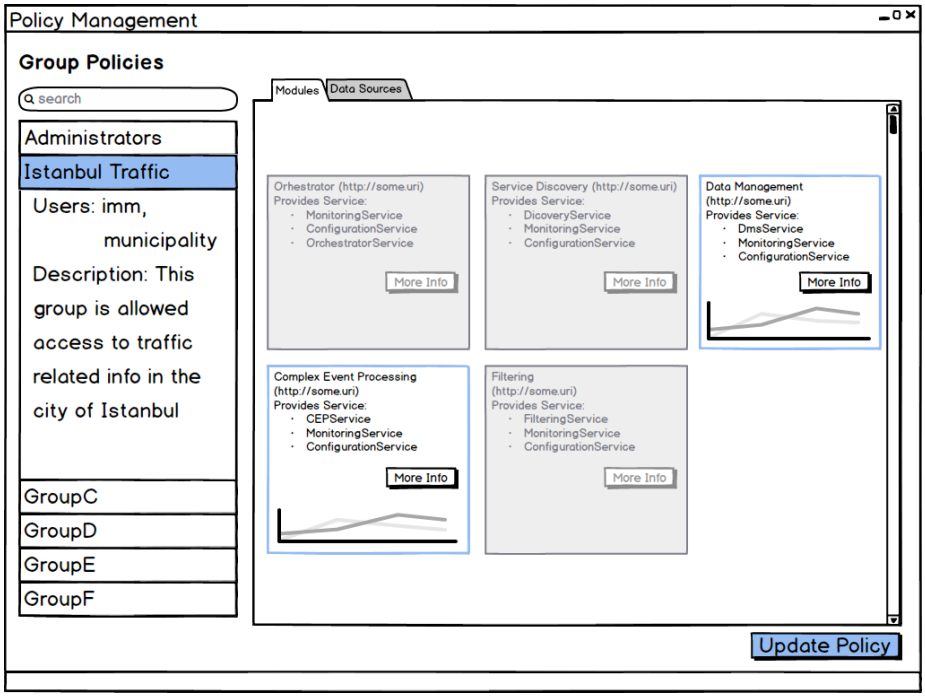

Figure 6: Application Polices - Selection of Vital modules for the specific application

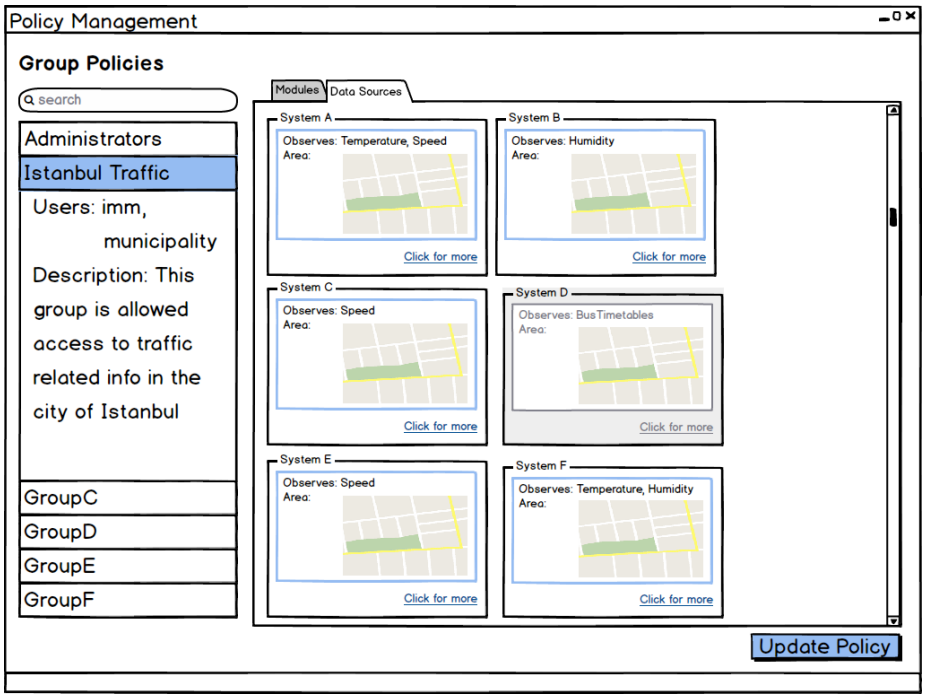

Figure 7: Application Polices - Selection of IoT Systems / Datasources (PPIs) for the specific application

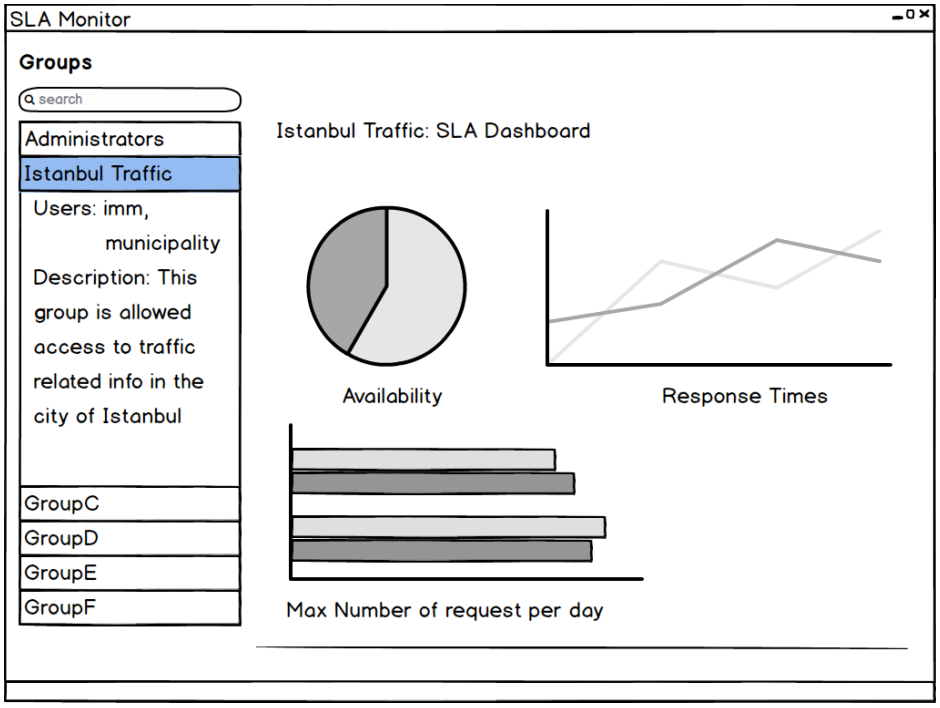

Figure 8: SLA Monitor – Dashboard view of the combined SLA metrics for an Application

A first experimental design was also produced for a possible section of the UI meant to guide the user (deployer/administrator) in the process of a basic configuration of the system exploiting default values and automated actions. You can see how the view currently looks in Figure 8.

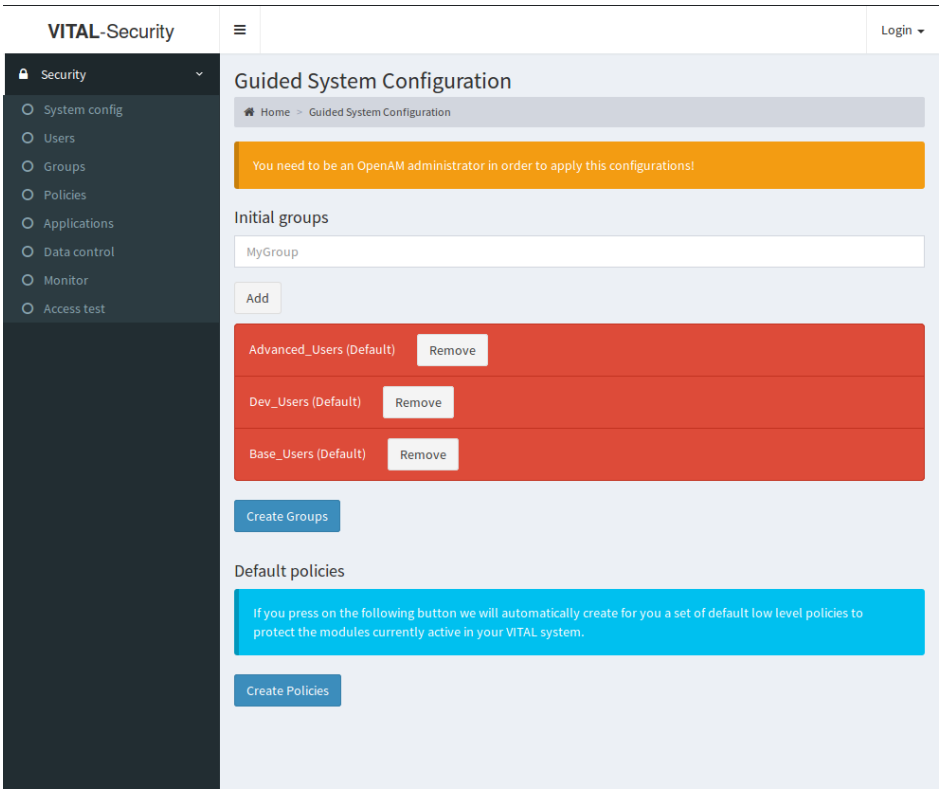

Figure 9: Template prototype for a view to ease system configuration

5 CONCLUSIONS AND FUTURE PLANS

This document has reported on the design and implementation of the first release of the VITAL governance toolkit. The goal of the latter is to provide a range of configuration functionalities, which enable the customization of each VITAL deployment for use within a specific smart city or region. The document starts with a short discussion of the main governance requirements, which have driven the design and implementation of relevant governance/configuration functionalities. The latter are in essence complementing the management functionalities, which have been implemented as part of deliverable D5.1. Furthermore, they are made accessible from the same management interface/tool i.e. the interface implemented and presented as part of deliverable D5.1.

As a general rule the governance functionalities enforce restrictions on the data and services that can be accessed by specific users, taking into account the users' privileges, the characteristics of the smart city where the deployment will take place, as well as SLAs with IoT platform and services providers. The goal of these restrictions is to economize on the required resources and costs, but also to ensure stakeholders' control over the smart city operations that can be performed by developers.

As part of the current (initial) version of the deliverable the main functionalities have been illustrated on the basis of mock-ups. Concrete implementations of these mock-ups will be provided as part of the second and final version of the deliverable. The final version will also contain the specification and implementation of functionalities for the instantiation of the VITAL platform for a specific city, along with functionalities for federating different smart city applications over the same city-specific VITAL instance. Moreover, a mechanism to allow the configuration of policies for fine-grained authorization to access data in the VITAL DMS (Data Management Service) will be defined and implemented. It will use identities managed by OpenAM and access conditions that will be ontologically modelled and stored in the DMS itself. Also, the policy TABs of the UI will be expanded to support the configuration of this kind of policies too, and the support of higher level policies will be introduced. This development will be validated with the VITAL pilot scenarios, yet their scope will not be limited by the scenario functionalities.

The final version of the deliverable will be accompanied by the final implementation of the toolkit, which will be integrated within the management tool implemented in deliverable D5.1. Hence, VITAL will offer a complete management environment, enabling configuration and monitoring operations not only on sensors, devices and IoT platform, but also on users, security policies, SLAs and VITAL deployment instances.

6 REFERENCES

[Vital-D3.2.2-2015] John Soldatos, Katerina Roukounaki et al, “Specification and Implementation of Virtualized Unified Access Interfaces V2”, 2015.

[Vital-D5.1.2-2015] Fotis Stamatelopoulos, Angelos Lenis, Stelios Pantelopoulos, Anne Helmreich et al, “Management services over federated IoT platforms V2”, 2015

