HORIZONS 2020 PROGRAMME

Research and Innovation Action – FIRE Initiative

Call Identifier:	H2020–ICT–2014–1
Project Number:	643943
Project Acronym:	FIESTA-IoT
Project Title:	Federated Interoperable Semantic IoT/cloud Testbeds and Applications

Specification of Experiments, Tools and KPIs

Document Id:	FIESTAIoT-D23-151110-Draft
File Name:	FIESTAIoT-D23-151110-Draft.pdf
Document reference:	Deliverable 2.3
Version:	Draft
Editor:	Bruno Almeida / Tiago Teixeira
Organisation:	UNPARALLEL Innovation, Lda
Date:	10 / 11 / 2015
Document type:	Deliverable
Dissemination level:	PU

DOCUMENT HISTORY

Rev.	Author(s)	Organisation(s)	Date	Comments
V01	Martin Serrano	NUIG-Insight	2015/02/01	Initial Draft Proposal
V02	Marcio Mateus	UNPARALLEL	2015/07/27	Experiment overview
V03	Flavio Cirillo & Ernoe Kovacs	NEC	2015/08/13	Updated NEC's experiment description and overview
V04	Luis Sanchez & David Gómez	UNICAN	2015/08/13	Updated UNICAN's experiment description and overview
V05	Tiago Teixeira	UNPARALLEL	2015/09/15	Updated document structure
V06	Luis Sanchez & David Gómez	UNICAN	2015/09/23	Updated UNICAN experiment specification
V07	Flavio Cirillo & Ernoe Kovacs	NEC	2015/09/23	Updated NEC experiment specification
V08	Rachit Agarwal & Animesh Pathak	INRIA	2015/09/23	Updates Inria's experiment description and Requirement Analysis section
V09	Bruno Almeida	UNPARALLEL	2015/09/23	Minor changes in the experiment's description
V10	Tiago Teixeira	UNPARALLEL	2015/09/28	Comments on Experiments specifications
V11	Luis Sanchez & David Gómez	UNICAN	2015/10/05	Updated UNICAN's experiment description and overview
V12	Flavio Cirillo & Ernoe Kovacs	NEC	2015/10/06	Updated NEC experiment specification
V13	Rachit Agarwal	INRIA	2015/10/08	Updates INRIA experiment description
V14	Flavio Cirillo & Ernoe Kovacs	NEC	2015/10/12	Updated NEC experiment specification
V15	Fábio Oliveira	UNPARALLEL	2015/10/13	Added Consolidation chapter
V16	Tiago Teixeira	UNPARALLEL	2015/10/13	Updated Consolidation Chapter
V17	Luis Sanchez & David Gómez	UNICAN	2015/10/13	Minor changes on UNICAN's experiment
V18	Tiago Teixeira	UNPARALLEL	2015/10/13	Minor Changes
V19	Nikos Kefalakis, Paul Grace, Amelie Gyrard	AIT, ITINNOV, NUIG-Insight	2015/10/22	Technical review & Quality review
V20	Tiago Teixeira	UNPARALLEL	2015/10/23	Changes after Internal Review
V21	Rachit Agarwal, David Gómez, Flavio Cirillo	INRIA, UNICAN, NEC	2015/10/29	Updates base on Technical Review
V22	Bruno Almeida	UNPARALLEL	2015/10/30	Final Version for Submission
V23	Martin Serrano	NUIG	2015/11/05	Circulated for Approval
Draft	Martin Serrano	NUIG	2015/11/10	EC Portal Submitted

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

TERMS AND ACRONYMS

3G	Third Generation of mobile telecommunication technology
API	Application Program Interface
dB	Decibel
DSL	Domain Specific Language
EaaS	Experiment as a Service
GPS	Global Positioning System
GUI	Graphical User Interface
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
OTAP	Over The Air Programing
SEL	Service Experimentation Layer
SSC	Super Stream Collider
urn	Uniform Resource Name
VE	Virtual Entity

1 POSITIONING

1.1 FIESTA-IoT Scope

Recent advances in the Internet of Things (IoT) area have progressively moved in different directions (i.e. designing technology, deploying the systems into the cloud, increasing the number of inter-connected entities, improving the collection of information in real-time, and no less important—the security aspects in IoT). IoT advances have drawn a common grand challenge that focuses on the integration of the heterogeneous IoT generated data. This key challenge is to provide a common sharing model or a set of models organizing the information coming from the connected IoT services, IoT technology and systems, and more important to be able to offer them as experimental services in order to optimise the design of new IoT systems and facilitate the generation of solutions more rapidly.

In FIESTA-IoT we focus on the problem of formulating and managing IoT data from heterogeneous systems and environments and their entity resources (such as smart devices, sensors, actuators, etc.), this vision of integrating IoT platforms, testbeds and their associated silo applications within cloud infrastructures is related to several scientific challenges, such as the need to aggregate and ensure the interoperability of data streams stemming from different IoT platforms or testbeds, as well as the need to provide tools and techniques for building applications that horizontally integrate diverse IoT Solutions. The convergence of IoT with cloud computing is a key enabler for this integration and interoperability, since it allows the aggregation of multiple IoT data streams towards the development and deployment of scalable, elastic and reliable applications that are delivered on-demand according to a pay-as-you-go model.

The activity in FIESTA-IoT is distributed in 7 Work Packages (WP). WP1 is dedicated to the project activities coordination, considering consortium administration, financial management, activity co-ordination, reporting and quality control. In FIESTA-IoT one of the main objectives is to include experimenters and new testbeds to test and provide feedback about the platform and tools, thus open calls for those tenders will be issued (these are also part of the WP1 activity and is called selection of third-parties).

WP2 focuses on stakeholder’s requirements and the analysis of IoT platforms and testbeds in order to define strategies for the definition and inclusion of experiments, tools and Key Performance Indicators (KPIs). The activities in WP2 are focused on studying the IoT platforms and testbeds and the specification of the experiments, the detail of the needed tools for experimentation, and the KPIs for validating the proposed solutions. This WP will conduct the design and development of the Meta-Cloud Architecture (including the relevant directory of IoT resources) and will define the technical specification of the project. WP2 also focuses on analysing the Global Market Confidence programme and establishes the Certification Programme Specifications that will drive the global market confidence and certification actions around the IoT experimentation model.

WP3 focuses on providing technologies, interfaces, methods and solutions to represent the device and network nodes of the testbeds as virtualized resources. The virtualized resources will be represented as services and will be accessible via common service interfaces and Application Program Interfaces - APIs (i.e. the FIESTA-IoT Testbed interfaces/APIs). The virtualized resources and their capabilities and interfaces will be also described using semantic metadata to enable (semi-) automated discovery, selection and access to the testbed devices and resources.

WP4 will implement an infrastructure for accessing data and services from multiple distributed diverse testbeds in a secure and testbed agnostic way. To this end, it will rely on the semantic interoperability of the various testbeds (realized in WP3) and implement a single entry point for accessing the FIESTA-IoT data and resources in a seamless way and according to an on-demand Experimentation-as-a-Service (EaaS) model. The infrastructure to be implemented will be deployed in a cloud environment and will be accessible through a unified portal infrastructure.

WP5 focuses on designing, deploying and delivering a set of experiments, so as to assess the feasibility and applicability of the integration and federation techniques, procedures and functions developed during the project lifetime. It will define a complete set of experiments to test the developments coming from other WPs (mainly WP3 and 4), covering all of the specifications and requirements of WP2. Developments will be tested over available IoT environments and/or smart cities platforms. WP5 will also provide evaluation of the KPIs defined for every experiment/pilot. The final deployed experiments will include a subset of those coming from WP2, 3 and 4, as well as those provided by FIESTA-IoT Open Calls.

WP6 focuses on the establishment and validation of the project's global market confidence on IoT interoperability, which will provide a vehicle for the sustainability and wider use of the project's results. The main activity in this WP focuses on specifying and designing an IoT interoperability programme, including a set of well-defined processes that will facilitate the participation of researchers and enterprises. WP6 works on providing a range of certification and compliance tools, aimed at auditing and ensuring the openness and interoperability of IoT platforms and technologies. WP6 also focuses on interoperability testing and validation and to provide training, consulting and support services to the FIESTA-IoT participants in order to facilitate platforms and tool usability, but also to maximize the value offered to them by using FIESTA-IoT suite and tools.

WP7 focuses on ensuring that the FIESTA-IoT suite, models and tools engage well with the community outside of the project; from promotion and engagement of new customers, to the front line support of current users, and the long-term exploitation of results and sustainability of the facility itself. This will be carried out in a coordinated manner such that a consistent message and professional service is maintained. Dissemination activities and the KPI to measure the impacts will be studied and used in this WP. An ecosystem plan including the specification of processes, responsibilities and targets will be generated and the evaluation and effectiveness of the operating model will be evaluated within this WP. In this WP the successes of stakeholder engagement and reports on their satisfaction with the services offered in FIESTA-IoT will be put in place at the end of the project.

1.2 WP2 Overview

This WP covers the FIESTA-IoT requirements engineering activities and will produce the requirements associated with testbed-agnostic experimentation, as well as with the EaaS model to design and conduct experiments. WP2 is composed of five tasks (depicted in Figure 1), which tackle distinct aspects of the FIESTA-IoT EaaS Experimental Infrastructure:

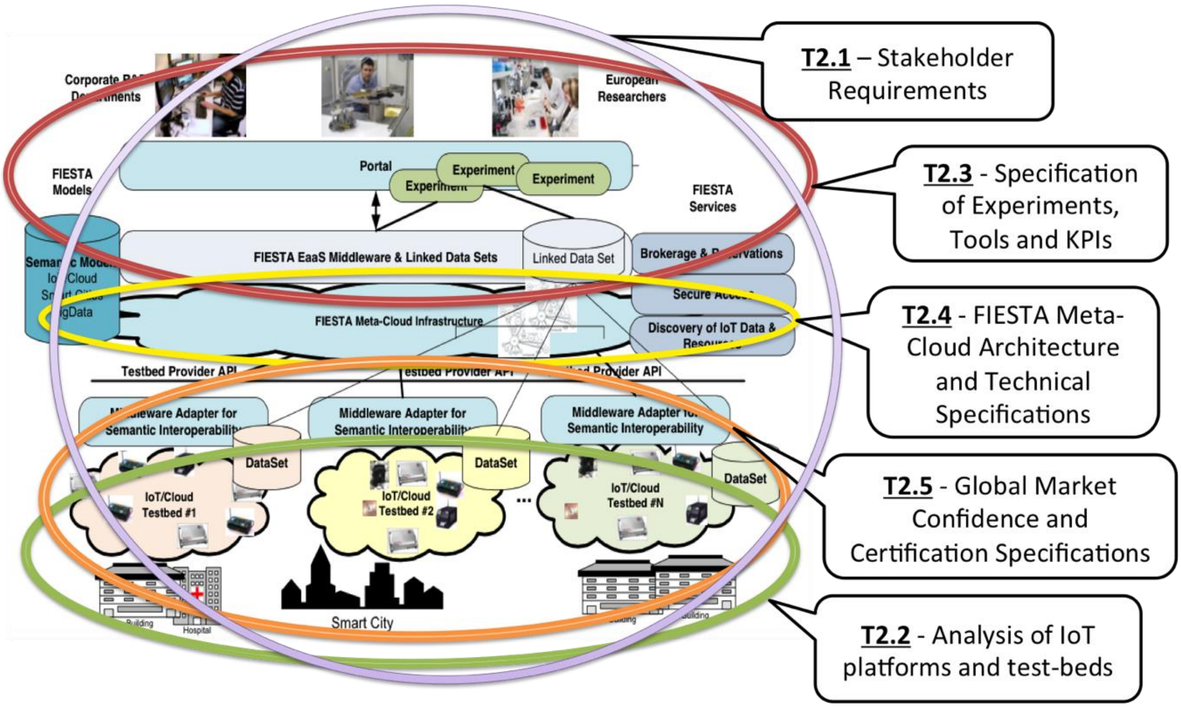

Figure 1 - WP2 Overview

The WP2 Tasks cross all aspects of the FIESTA-IoT Infrastructure. They are:

Task 2.1. Stakeholder Requirements: This task is responsible for gathering and processing all stakeholder requirements (using the Volere requirements specifications). The involved stakeholders include: the IoT testbeds to be integrated, the experiment providers, and also researchers and experimenters. Also external projects (such as Open-IoT and Fed4Fire) will provide requirements so, to prepare FIESTA-IoT to the Open-calls. This task will produce a set of requirements that will be used by all other WP2 tasks.

Task 2.2. Analysis of IoT platforms and testbeds: This task is focused on the testbeds and IoT platforms, analysing and describing what they do and how they do it. It will also use the set of testbed requirements produced in T2.1 to better understand if each testbed can fulfil the stakeholders' requirements. This task will then model the testbeds and IoT platforms in functional blocks using the IoT-A Architecture Reference Model (ARM). It will gather what type of information they provide, and how they provide this information so that Task 2.4 can take this it into account when developing the FIESTA-IoT Architecture. The outcome of this task will provide a basis for WP3.

Task 2.3. Specification of Experiments, Tools and KPIs: This task will specify all planned experiments and extrapolate the needed tools to execute those experiments. It will use the experiment related requirements produced in T2.1 and analyse them in terms of the tools that need to be provided by FIESTA-IoT to the experimenters. It will also specify the KPIs of each experiment so that later validation can occur. The result of this Task will be used as input to WP5.

Task 2.4. FIESTA-IoT Meta-Cloud Architecture and Technical Specifications: This task will define the FIESTA-IoT Meta-Cloud Architecture, leveraging on the IoT-A ARM, and the technical specifications that will drive all of the development work of the project. It will use information from previous tasks to identify the main building blocks, design & technology choices, and specify the functional blocks of the FIESTA-IoT architecture needed for achieving FIESTA-IoT's technical objectives. This architecture will serve as a base for all of the development phase of the project and more specifically for WP4.

Task 2.5. Global Market Confidence and Certification Specifications: This task is intended to study and define the global market confidence and certification specification. This means that this task will define the certification process, and the set of requirements that are required for a testbed to comply in order to be integrated in FIESTA-IoT. The outcome of this task will be used in WP6.

As described in the previous tasks description, the outcomes of each task will be used by other tasks of this WP2, or be used as inputs for the work in other WPs.

These relations between WP2 tasks and other WPs are depicted in Figure 2.

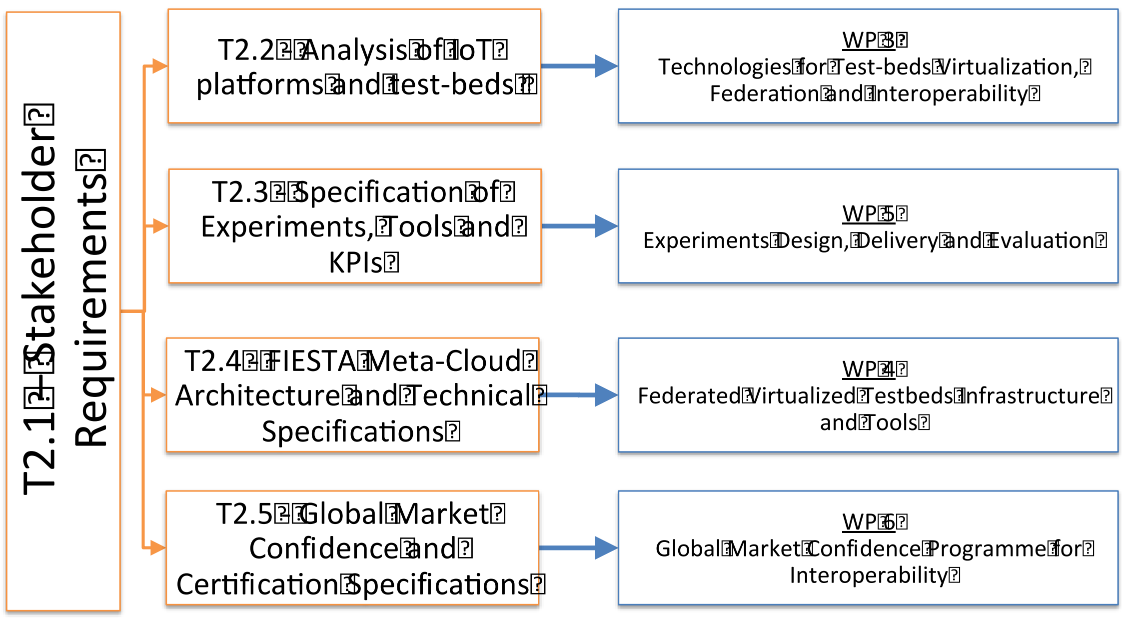

Figure 2 - Relationship between WP2 tasks and with other WPs

In reference to the FIESTA-IoT project general objective(s), WP2 has a set of defined activities (related to these sub-objectives) that are described as follow:

1) Determination of Stakeholder requirements.
2) Description of IoT platforms and testbeds in order to facilitate their integration into the FIESTA-IoT infrastructure.
3) Specification of planned experimentation and its executing tools, and the KPIs that will be used for validation.
4) Definition of the FIESTA-IoT Meta-Cloud architecture and the technical specifications required for the development WPs.
5) Definition of the Global Market Confidence and Certification specifications.

The WP2 will also result in five deliverables, which will be directly linked with the objectives and tasks of the WP. Each deliverable will be an outcome of each task, meaning that deliverable D2.1 will be provided at the end of T2.1 with the results of that specific task. The following table (Table 1) details the set of deliverables to be expected from WP2, with reference to the related tasks, the responsible partner for each deliverable and all others contributors.

Table 1 - WP2 Deliverables

No.	Deliverable	Responsible Partner	Contributors
D2.1	Stakeholders Requirements	UNPARALLEL	NUIG-DERI, NEC, UNICAN, SODERCAN, SDR
D2.2	IoT Platforms and Testbeds Analysis	Com4Innov	KETI, UNICAN, UNPARALLEL, AIT, NUIG-DERI, INRIA, NEC
D2.3	Experiments, Tools and KPIs Specification	UNPARALLEL	UNICAN, INRIA, NEC, NUIG-DERI, AIT, ITINNOV, SODERCAN
D2.4	FIESTA Meta-Cloud Architecture and Technical Specifications	UNIS	AIT, NUIG-DERI, UNICAN, ITINNOV, KETI
D2.5	Global Market Confidence and Certification Programme Specifications	EGM	AIT, SODERCAN

1.3 Audience

This deliverable addresses the following audiences:

- **Researchers and engineers within the FIESTA-IoT consortium**, who will take into account the various requirements in order to research, design and implement the architecture of the FIESTA-IoT Meta-Cloud Architecture.
- **Researchers on Future Internet Research and Experimentation (FIRE) focused on IoT and cloud computing systems experimenters at large**, given that the present deliverable could be useful reading for researchers studying alternative IoT technologies and applications, along with indications and requirements towards building/establishing experimental architectures.
- **Members of other Internet-of-Things (IoT) communities and projects (such as projects of the IERC cluster)**, who can find in this document a readily available requirements analysis for experimentation-like IoT services and tools. For these projects the document could provide insights into requirements and technological building blocks enabling the convergence between utility/cloud computing and the IoT for enabling EaaS.

1.4 Terminology and Definitions

This sub-section is intended to clarify the terminology used during this project. This initial step is intended to clarify all of the important terms used, in order to minimise misunderstandings when referring to specific parts involved in the generation of data and the FIESTA-IoT concepts. The following definitions (listed in Table 2) were set regarding the domain area of FIESTA-IoT, and so are aligned with terminologies used in the FIRE community and in reference to IoT-related projects (such as IoT-A).

Table 2 - Terminology and Definitions table

Term	Definition
Characteristic	An inherent, possibly accidental, trait, quality, or property of resources (for example, arrival rates, formats, value ranges, or relationships between field values).
Device	Technical physical component (hardware) with communication capabilities to other Information technology (IT) systems. A device can be attached to, or embedded inside a physical entity, or monitor a physical entity in its vicinity (IoT-A, 2013). The device could be: • **Sensor**: A sensor is a special device that perceives certain characteristics of the real world and transfers them into a digital representation (IoT-A, 2011). • **Actuator**: An actuator is a mechanical device for moving or controlling a mechanism or system. It takes energy, usually transported by air, electric current, or liquid, and converts that into some kind of motion (IoT-A, 2011).

Discovery	Discovery is a service to find unknown resources/entities/services based on a rough specification of the desired result. It may be utilized by a human or another service. Credentials for authorization are considered when executing the discovery (IoT-A, 2013).
Domain	Refers to an application area where the meaning of data corresponds to the same semantic context. For instance, pressure in Water Management Domain may refer to water pressure on pipes while in Air Quality Domain it refers to atmospheric pressure
Information	Content of communication; data and metadata describing data. The material basis is raw data, which is processed into relevant information, including source information (e.g., analogue and state information) and derived information (e.g., statistical and historical information) (IEEE, 2007).
Measurement	The important data for the experimenter. It represents the minimum piece of information sent by a specific resource, which the experimenter needs in order to fulfil the objective of the experiment
Metadata	The metadata is the additional information associated with the measurement, facilitating its understanding.
Physical Entity	Any physical object that is relevant from a user or application perspective. (IoT-A, 2011). Physical Entities are the objects from the real world that can be sensed and measured and they are virtualized in cyber-space using Virtual Entities.
Requirement	A quantitative statement of business-need that must be met by a particular architecture or work package. (Haren, 2009)
Resource	Computational element that gives access to information about or actuation capabilities on a Physical Entity (IoT-A, 2011).
Stakeholder	An individual, group, or organization, who may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of a project (Project Management Institute, 2013)
Testbed	A testbed is an environment that allows experimentation and testing for research and development products. A testbed provides a rigorous, transparent and replicable environment for experimentation and testing (Gavras, 2010)
Federated testbeds	A testbed federation or federated testbeds is the interconnection of two or more independent testbeds for the creation of a richer environment for experimentation and testing, and for the increased multilateral benefit of the users of the individual independent testbeds (Gavras, 2010)
Interoperability	The ability of two or more systems or components to exchange information and use the information that has been exchanged (IEEE, 1990)
Experimentation facility	An experimentation facility can be understood as an environment with an associated collection of tools and infrastructure that sits on top of one or several testbeds and can be used to conduct experiments to assess and evaluate new paradigms, architectural concepts and applications (MyFIRE, 2011)

Experiment	Experiment is a test under controlled conditions that is made to demonstrate a known truth, examine the validity of a hypothesis, or determine the efficacy of something previously untried (Soukhanov, Ellis, & Severynse, 1992)
Semantic Interoperability	Semantic interoperability is the ability of computer systems to exchange data with unambiguous, shared meaning. Semantic interoperability is a requirement to enable machine computable logic, inference, knowledge discovery, and data federation between information systems
Service	Services (Technology) are services designed to facilitate the use of technology by end users. This services provide specialized technology-oriented solutions by combining the processes/functions of software, hardware, networks, telecommunications and electronics
Virtual Entity	Computational or data element representing a Physical Entity. Virtual Entities can be either Active or Passive Digital Entities (IoT-A, 2013).

1.5 Executive Summary

This deliverable intends to give to the audience an overview of the FIESTA-IoT in-house experiments. These experiments are valid and very useful but are also meant to provide a validation of the FIESTA-IoT mechanisms and functionalities.

There are three in-house experiments defined in FIESTA-IoT, which will be conducted by NEC, UNICAN and INRIA, respectively:

- Data Assembly and Services Portability Experiment;
- Dynamic Discovery of IoT resources for testbed agnostic data access;
- Large Scale Crowd sensing Experiments;

This document will provide an overview of each experiment, demonstrating what kind of actions FIESTA-IoT must be capable of handle while providing access to the in-house testbed's data. The combination of the execution of the three experiments will allow the validation of all of the main FIESTA-IoT functionalities.

Each experiment is duly specified in terms of its execution and the datasets that they will consume from FIESTA-IoT, and in terms of the requirements each of them is related to. For that, each experiment specifies in detail the requirements (presented in D2.1) they relate to, providing a small description of that relationship. The experiments also describe the KPIs that are fulfilled as the best way to further evaluate not only the experiment execution and results, but also the FIESTA-IoT behaviour according to the experimenters. Also described by each of the experiments are the features they require from the FIESTA-IoT experimentation tools.

An aggregation of the information provided in the experiments specifications sections, is then presented in the consolidation chapter, were the information is grouped by Requirements, Datasets, KPI's and Features. This chapter intends to ease the access to information regarding one of these groups, as each of them provide all of the information regarding each of them.

2 EXPERIMENTS OVERVIEW

Three in-house experiments will be executed on FIESTA-IoT, where a different FIESTA-IoT partner will execute each experiment. These experiments have as their purpose not only the execution of the experiment itself, but to test and validate the mechanisms and functionalities provided by FIESTA-IoT. To achieve this objective there is the need to identify which features and internal mechanisms of FIESTA-IoT will each experiment test and validate. The following overview of the in-house experiments has as main purpose the identification of what will be tested and validated by each experiment.

2.1 Data Assembly and Services Portability Experiment

The main goal of this experiment is to verify the FIESTA-IoT Objectives present in the DoA, mainly Objective 2 ("Testbed Agnostic Access to IoT Datasets") and Objective 3 ("Tools and Techniques for IoT Testbeds Interoperability and Portability") with a FIESTA-IoT internal integrated experiment (Objective 5). Successful implementation of this experiment will also verify that the FIESTA-IoT EaaS infrastructure is a suitable infrastructure for large-scale IoT experiments (Objective 1). It will provide valuable insights and experimental facts as for the best practice (Objective 7) and support the Global Market Confidence Program (Objective 4).

This experiment has selected the topic of Smart City Performance Indicators (as for example highlighted by the World Bank study (Hoornweg & Blaha, 2006)) for its experiment. The target is to compute relevant Smart City Performance Indicators from the available set of information. It can be easily seen that computing those indicators requires service portability across the different testbeds as the same indicators need to be computed for the different testbeds. It can also be seen that we need the discovery and semantic interoperability feature of FIESTA-IoT to ensure that we can have reliable indicators despite the data assembly from very heterogeneous data sets in the different testbeds.

The core function of the experiment is an IoT application that builds a Smart City Performance Indicator model based on sensor information. As a novel aspect, we wish to build not only a single indicator visualizing a general indicator like e.g. the "health" of a city. We wish to create a "zoomable" set of indicators where we can analyse the city performance along the following dimensions

- **Detail Level**: from general city indicators to specific indicators of a single aspect of city management, e.g. environmental monitoring.
- **Space**: from indicators considering the complete city to indicators on the level of places, streets, even houses or rooms.
- **Time**: zooming from the most current information to indicators capturing a long time period.

These indicators can be used for visualization, trend analysis and triggering of notifications if a certain situation has occurred. The application will be designed in such a way that different types of sensor information relevant for different application areas can be used. Examples are the monitoring of environmental parameters like pollution, humidity, temperature, light and noise, but could also be: the parking situation in a city/area, water/irrigation levels in a park or agricultural setting, or the activity level in a certain area.

The experiment serves the purpose of showing that semantic interoperability across different IoT infrastructures can be achieved, greatly simplifying the development of applications. Utilizing semantic interoperability for simplifying application development and increasing the portability across networks is highly relevant for NEC's businesses in different IoT domains, in particular the smart city area, where NEC is involved in more than 40 projects around the globe, including commercial deployments.

A representation of information flow for this experiment is represented in Figure 3. In this example there are two experiments where the first asks for location values and the second for temperature values, disregarding the domain that produced these values. Therefore, FIESTA-IoT will collect data from all testbeds capable of providing location and temperature measures. FIESTA-IoT will then represent data from each domain in the corresponding data set structure. Each domain dataset may represent information in a different way, for instance one domain may refer to temperature as "Air Temperature" while other uses the term "Room Temperature". Moreover, location values may be considered in some domains as data and as metadata in other domains. FIESTA-IoT will perform the matching between these domain-specific data representations and the concepts and phenomena behind them, in order to be able to extract the information regarding to the temperature and location pairs collected from each testbed and provide it to the experiment.

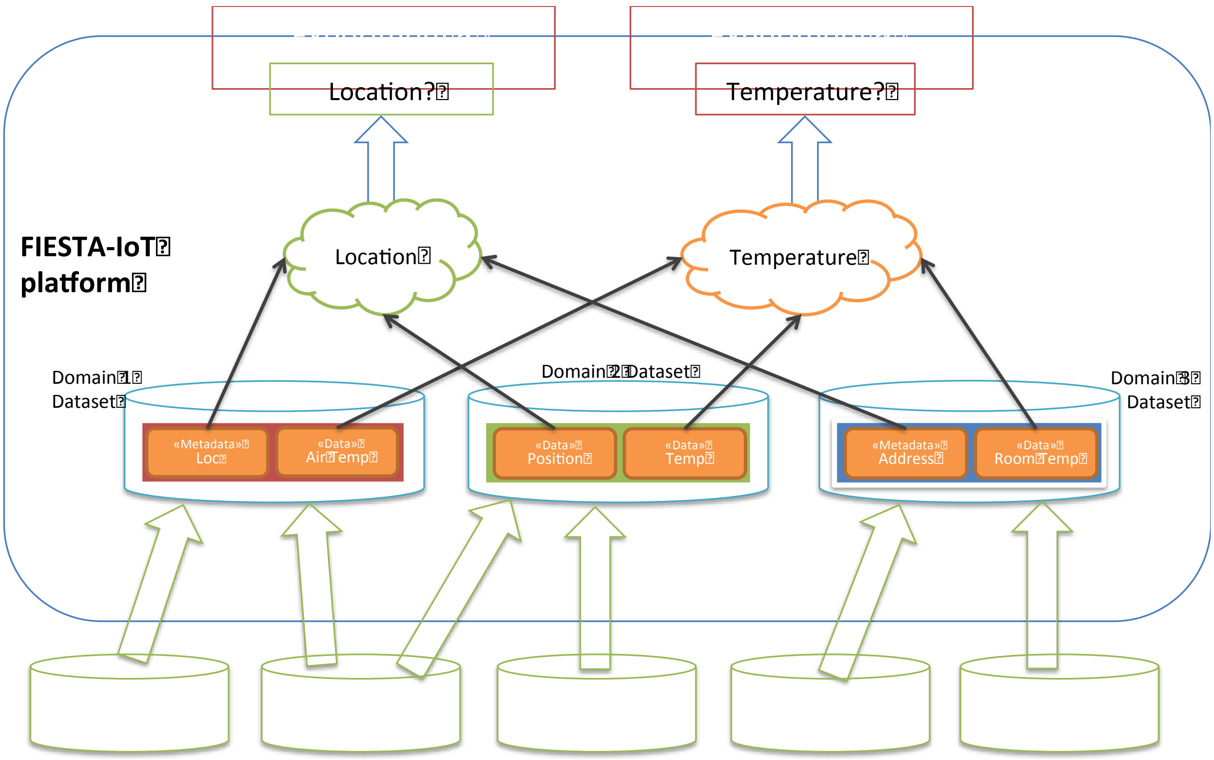

Figure 3 - Information acquisition process for NEC's experiment

2.2 Dynamic Discovery of IoT resources for testbed agnostic data access

The experiment designed by UNICAN has as main goal to test and validate FIESTA-IoT capabilities to discover resources as well as to access and process data from multiple testbeds. For such, this experiment will exploit the FIESTA-IoT meta-

directory where resources (e.g. data streams) will be registered in order to dynamically discovery and use data from one or more testbeds (in a seamless and testbed agnostic way). The meta-directory will also work with common templates to define, identify and classify resources from different environments.

The experiment will focus on the dynamic acquisition and processing of weather information (based on temperature, pressure, wind speed, UV, and humidity data streams), towards consolidating and visualizing data from multiple locations. The experimenter will be able to dynamically specify the locations/areas for which data will be collected. The specification of these areas might lead to data acquisition from one or more testbeds.

This experiment intends to collect data as described in Figure 4. In this experiment, the experimenter requests data from a specific domain, the Weather domain. All testbeds that are capable of gathering weather data within the location/region specified by the experiment will provide this data to FIESTA-IoT platform. The data provided by testbeds will be semantically annotated by FIESTA-IoT platform according to the semantic concepts defined in FIESTA-IoT and assembled in a domain-specific structure. This data can then be delivered to the experiment. This data only has information regarding the semantic concepts and phenomena associated with it, having no information about the source testbeds.

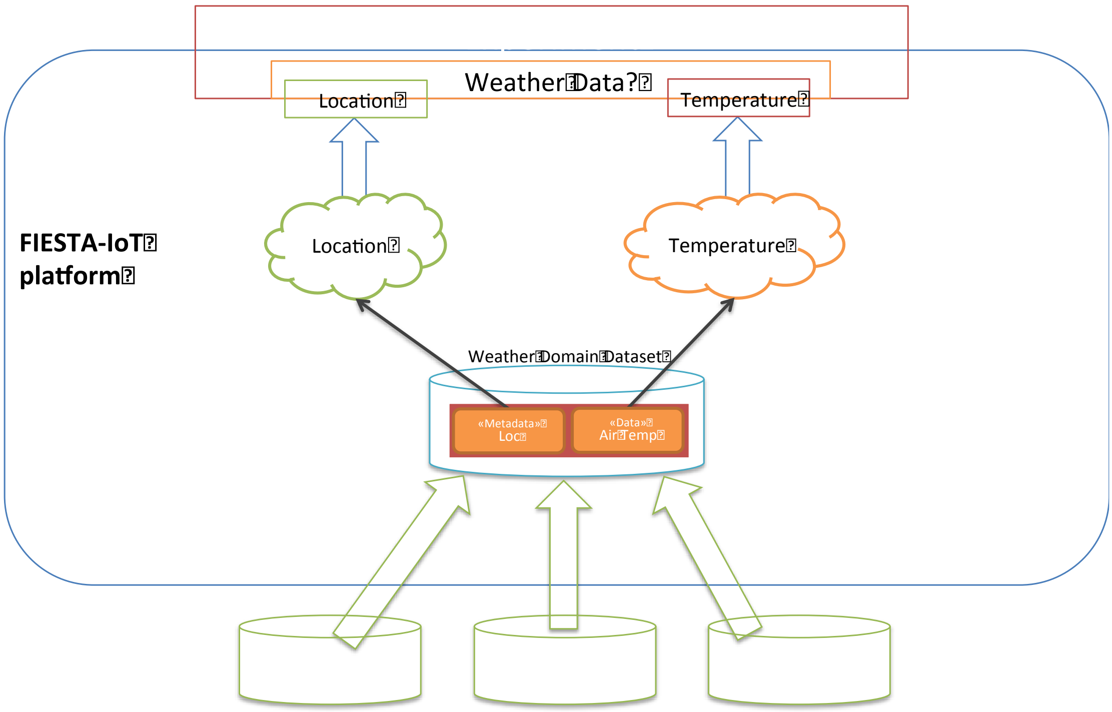

Figure 4 - Information acquisition process for UNICAN's experiment

The added value of the experiment is that the experimenter will be able to dynamically select and process data streams needed for the forecasting process, regardless of the testbed providing these data. The experiment is intended to be easily expandable and readily collect data from any number of additional testbeds that will be integrated/federated as part of the FIESTA-IoT infrastructure.

2.3 Large Scale Crowd sensing Experiments

INRIA will setup and execute a range of crowd sensing experiments using the FIESTA-IoT infrastructure. As part of these experiments, experimenters will be able to specify a set of devices and data streams that could participate in a given crowd sensing experiment (e.g., large scale pollution monitoring or energy metering). The crowd sensing experiments will be able to use data streams from participants interfacing to different IoT platforms/providers in order to provide/stream data acquired via their smart phones. FIESTA-IoT will provide an infrastructure where different data streams could be integrated regardless of the testbeds from where their data are provided.

The crowd sensing experiments will be designed to run on FIESTA-IoT infrastructure using the participatory sensing capabilities that already exist in the testbeds connected to FIESTA-IoT platform. Crowd sensing experiments will not request new sensing capabilities from the testbeds, acting as data collectors as shown in Figure 5.

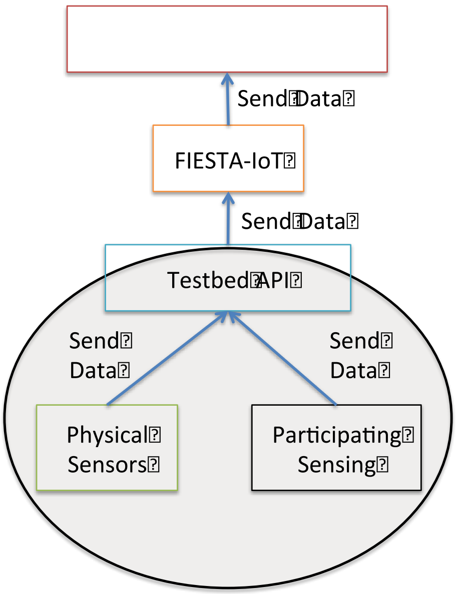

Figure 5 - Data collection approach for INRIA's experiments

Each experiment will focus only in one domain (e.g. road conditions, places visited by an individual, etc.). Therefore, the information acquisition process is similar to the one used described in UNICAN's experiment, represented in Figure 4. Just like in the process described in UNICAN's experiment, the crowd sensing experiment requests data from a specific domain and every testbed able to produce that data will provide it to FIESTA-IoT platform. However, there is a big difference between these two processes since the type of resources used by crowd sensing experiments is very specific – crowd sensing resources.

The crowd sensing resources that will be used for the experiments are participatory sensing resources. Therefore, the availability of these resources does not depend only on the hardware aspects but also in the willingness of users to provide data during the duration of the experiment. Moreover, in order to provide some anonymity to crowd sensing participants, the testbed may generate a new resource every time that a crowd-sensing user provides data. This approach produce situations where one resource is only associated with one measurement at a given instance of time.

2.4 Summary

The functionalities of FIESTA-IoT that each experiment will validate are summarized in the following table. Six major functionalities that the experiments will be able to validate are identified:

- **Testbed-agnostic data acquisition** – experiment needs to collect data from resources in different testbeds in a seamless way;
- **Data acquisition from different domains** –experiment wants to collects data from different domains (e.g. weather, agriculture, energy efficiency, etc.);
- **Data acquisition from one specific domain** – experiment wants to collect data corresponding to a specific domain;
- **Semantic Interoperability between different domains** – experiment requires a semantic mapping between the concepts of different domains;
- **Dynamic discovery and selection of resources** – experiment expects to identify and select the resources that will be able to provide the requested data and will be available during the execution of the experiment;
- **Runtime discover and selection of resources** – experiment wants to collect the requested data from resources that became available during the execution of the experiment.

Table 3 presents a summary of how the FIESTA-IoT in-house experiments will relate and then validate the functionalities that will be provided to FIESTA-IoT users. The approach used in the table uses "double checks" (□□) to refer that the experiment will focus on validating that functionality, and uses "single checks" (□) to refer that the experiment can also be used to validate those functionalities, despite not being its main focus.

Table 3 - Summary of the FIESTA-IoT functionalities that will be validated by in-house experiments

Experiments	Testbed-agnostic data acquisition	Data acquisition from different domains	Data acquisition from one specific domain	Semantic Interoperability between different domains	Dynamic discovery and selection of resources	Runtime discover and selection of resources
Data Assembly and Services Portability Experiment	□	□□		□□	□□	□
Dynamic Discovery of IoT resources for testbed agnostic data access	□□		□□		□	□
Large Scale Crowd sensing Experiments	□		□			□□

3 EXPERIMENTS SPECIFICATION

3.1 Data Assembly and Services Portability Experiment

3.1.1 Experiment description

The proposed experiment focuses on writing an application that builds up a dynamic Smart City Performance Model based on sensor information in the space and time dimension. This can be used for visualization, trend analysis and triggering of notifications if a certain situation has occurred. The application will be designed in such a way that different types of sensor information relevant for different application areas can be used. Examples are the monitoring of environmental parameters like pollution, humidity, temperature, light and noise, but could also be the parking situation in a city/area, water/irrigation levels in a park or agricultural setting or the activity level in a certain area. Each FIESTA-IoT testbed provides at least a subset of the given information, so the goal is to write the application in such a way that it can run on all the different testbeds, ideally also using information from different testbeds, e.g. comparing the same aspect and computing the same indicator. For example activity level analysis applications (based on AutoID, QR Code technologies), notably applications that involve people scanning tags/QR codes, could be supported at both the Santander testbed and the Smart Campus testbeds. It can be expected that large parts of the Smart City Performance Model might be empty as the needed information is not directly available. In this case we can experiment with indirect indicators which compute the indicator from sensor information in related areas, e.g. the traffic situation of a street can be derived from the traffic information on its connected streets.

The experiments we conceive for the Smart City Performance Model can be of two typologies:

- **Resource Oriented**: the input data of the experiments are the information about the resources of the testbed connected to FIESTA-IoT platform. The outcome of such experiments inferred information of the resources and the testbeds, hence inferred information of the IoT infrastructure.
- **Observation Oriented**: the input data are mainly the observations produced by the device deployed in the connected testbed. The output of this kind of experiments is the inferred observation of a situation that happened in the real world.

The experiment serves the purpose of showing that semantic interoperability across different IoT infrastructures can be achieved, greatly simplifying the development of applications. FIESTA-IoT will explore different approaches for annotating information with semantic information, also regarding the granularity, e.g. on the sensor node level or in the cloud, on the level of individual items, for a whole data stream or database table. In addition testbeds themselves have different interaction styles, e.g. request-response or event-based and FIESTA-IoT will support applications in accessing required information in a uniform way, but possibly offering multiple alternatives. The experiment will compare some key alternatives and evaluate them regarding their efficiency, e.g. in terms of the number of interactions required, the size of the result or the overhead on FIESTA-IoT cloud and testbeds. Further, the experiment will also compare the quality of the results and the effort required for doing the adaptations in order to get the application to run on a testbed.

One example set of experiments are anomaly detection algorithms that compute indicators about the health of the sensing infrastructure itself. These techniques can be used to detect sensors malfunctioning in order to ease the maintenance of deployment or as a complement of other IoT analytics algorithms for ignoring faulty data.

Another application is the detection of energy consumption anomalies. This helps identify the malfunctioning of an electrical system thus saving energy and more importantly improves safety by preventing firebreak.

The experiment we will implement is intended to work online and react in real-time. For this reason, it is necessary to have the possibility to subscribe to a specific kind of data (or more than one kind at the same time) specifying also a location scope (the area where the measurement should be taken from). Furthermore, since these kind of algorithms expect time series as datasets, every observation should be tagged with a timestamp. Depending on the application scenario (e.g. shopping mall, pedestrian zone of a city centre, car lane etc.) a different frequency of measurements is needed and it should be specified at subscription time.

A second set of experiments that can be done via the FIESTA-IoT platform is for activities monitoring. For instance it would be useful to apply algorithms for inferring situations in a particular location context in order to trigger notifications. This kind of algorithm is very useful for detecting critical situations and hence for safety reasons. Belonging to the same set of algorithms, in terms of specification requirements, are the ones of pattern mining. Extracting patterns and finding correlation is an important task in data analytics for IoT data streams.

For this second set of experiments it would be needed to have a data subscription with a location scope (like the previous set) in addition to a description of the type of location where the observation was taken (e.g. office, conference room, hall, private room, open area, street cross etc.).

Scope

The overall idea of the intended experiments is to offer the backend services for a city status platform where indicators (Hoornweg, 2006) about the “health” status of the city and its many subsystems are displayed. This can show possible critical situations, trends that need monitoring, or even predictions of events (e.g. crowd forming). The main idea is to show the status of a geographic region regarding different environmental aspects via predefined indicators. The computation of these indicators might need dynamic IoT analytic processes discovering the available IoT resources and using semantic interoperability for ensuring that all available data is taken into account. For environmental it is intended not only the natural environment (e.g. water quality, air pollution, temperature trend etc.) but also urban environment (e.g. crowd forming, traffic situation, structure conditions etc.).

The indicators can be browsed at different granularity ranging through 3 degrees of freedom:

- **Space**: the same indicator can refer to different granularity of a geographic region. For example the air pollution trend can refer to a whole city, a suburb, a block, a building or even a single room of the building.
- **Time**: keeping the example of the air pollution trend indicator, it can show the trend of different sizes of time window or shifting the focus to past situations.

- **Detail level**: indicator can be grouped by top-aspect indicator, for example natural environment could be a group for water quality, air pollution etc. The top indicator can be a simple weighted average of the sub indicators, a threshold of the indicators or even a full analytics computation based on the sub indicators.

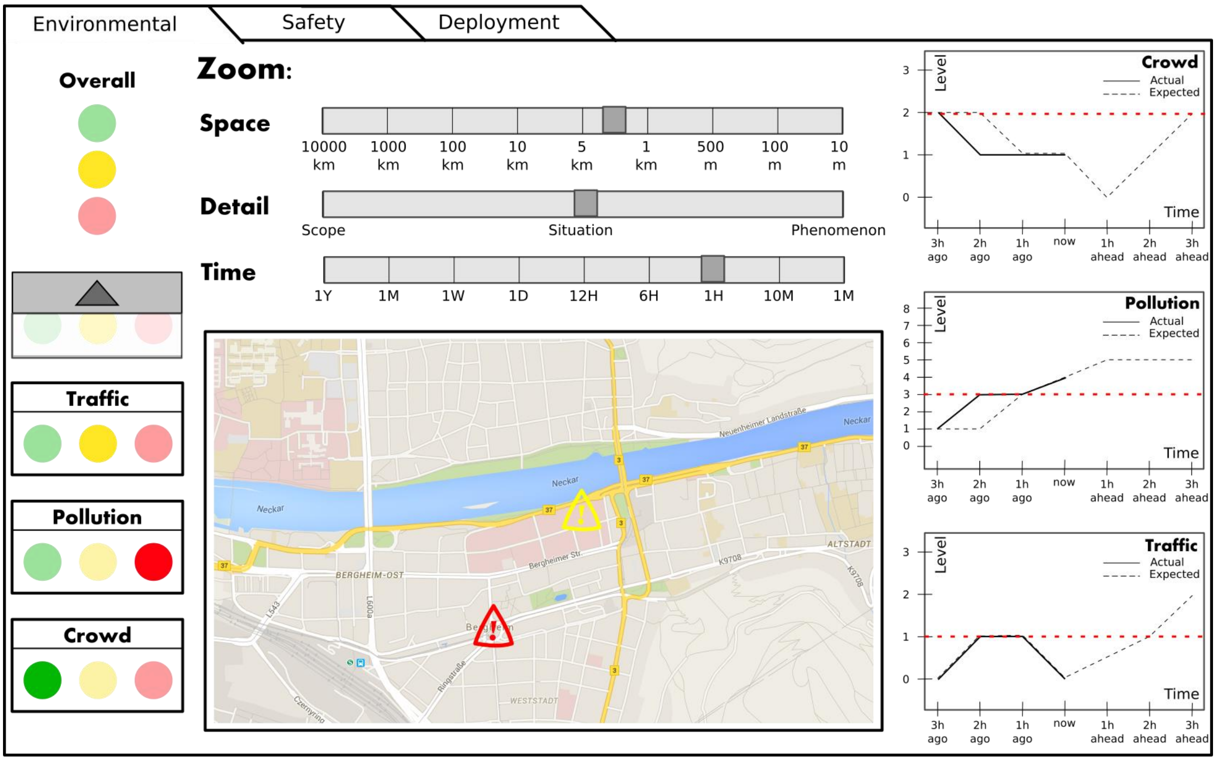

Figure 6: Mock of the Smart City Performance Model

Figure 6 shows a mock dashboard of our intended Smart City Performance Model. In the picture the *Environmental* group of indicators (also called scope) are shown, where for environmental are meant indicators about environment situations like traffic, pollution, crowds etc. Other groups of indicators can be selected by changing the tab on the top. For instance the *Safety* tab would have indicators such flood emergency, fire break emergency etc. Another example is the *Deployment* group of indicators where the indicators shown are, for instance, the status or the quality of the deployments.

The indicators refer to the area of the geographic region shown in the centre of the dashboard. The indicators itself are shown on the left part of the dashboard and summarized as a traffic light indicator: green means that such a situation is safe, yellow when the situation needs to be controlled and red when there is a problem to be solved. The same indicators are shown in graphs on the right side of the dashboard where the real values of the indicators are plotted over the time. Each graph shows two lines: the full line is the actual value computed with the measured observations, the dashed line is the expected value done via analytics. The latter analytics is done in order to predict problems in advance and then change the traffic light from green to yellow, as it happened for the traffic indicators in the mock figure, since although the actual value is under the defined threshold (the plotted red dashed

line) the analytics has predicted that the traffic would increase till the threshold and beyond in the near future.

The overall situation of the environment scope can be seen on the top left corner again with a traffic light summarization (yellow state in the figure).

The three sliding bars on the top of the map are the three degrees of freedom previously explained.

- Sliding the *Space* bar the map in the centre will zoom in and out and the indicators will compute their value using all the observed measurement within such area.
- The second bar is changing the *detail* level of the indicators: sliding to *Scope* the indicators would summarize in only one value and one traffic light indicator intended to show the state of the overall scope (in that case the environmental one); sliding to the *Situation* value the indicators are shown like in the picture, where the dashboard are showing an indicator for each situation; finally sliding to the *Sensors* level (and contextually selecting one situation indicator) the pure sensor values used for computing the selected indicator are shown in the graphs.
- The third sliding bar is changing the time window size. Sliding left to right the indicators are computed on values queried on a smaller and smaller time window and therefore the prediction becomes more and more accurate.

3.1.2 Specification of Datasets

Observation-Oriented experiments:

- Observation (aka Measurement): this is the pure value observed by the resource. The analytics we want to do is based on this information.
- Phenomenon: each observation is recorded because of a phenomenon, without this, the observation is just a number.
- Unit: together with the phenomenon, it completes the scope to give sense to the observation. This information is fundamental for avoiding misinterpretation of different observations, of the same phenomenon, in case of usage of multiples or submultiples of the same unit system or even usage of different unit systems.
- Timestamp: since our experiments will be mainly analytics on time-series, this value is necessary for indicating the time at which the measurement has been observed and so enable the analytics.
- Location: this value indicates the exact geographic position (i.e, latitude and longitude) where the measurement was recorded. If the resource producer is moving (like the buses in the Santander testbed) this information is of utmost importance since there are no other possibilities to infer such information.
- (If applicable) Description of Aggregation: in case of aggregation of data (e.g. Average/Max/Min) or outcome of a script (or even another experiment), this information will help the experimenter to understand what kind of functions have been applied to which original data
- Resource producer: this a unique resource ID, identifying the resource among the others (see Resource ID in the “Resource-Oriented experiments” paragraph of this section). Understanding the resource producer that has

produced the observation will allow the detection of a malfunctioning resource and therefore the invalidation of the observations produced by such resource.

Resource-Oriented experiments:

- Resource ID: this value will indicate univocally to which resource the other information refers. It is not important if such ID is generated by the FIESTA-IoT platform or directly by the testbed owner, what is important is that such ID is unique among the other resources (regardless the testbed) and that it is consistent during the time, i.e. it is not changing. The ID needs to be unique and consistent in order to make possible a creation of a time-series over a resource ID.
- Resource status: this value can be seen as the observation about the resource itself (status can be on, off, sleeping as you suggested or maybe quarantine), it will give information regarding the availability status of the resource, feedback from users about malfunctioning (if available) etc.
- Resource location: This information will locate the resource in the space and therefore it will allow analytics for the scope of the Smart City Performance Model.
- Timestamp: this will identify the time at which the information regarding the resource has been recorded. For the Smart City Performance Model such information will allow analysis on varying of the status of the resource.
- Testbed belonging: this value is expected to be an ID and it will associate the resource to a specific testbed. Such information is necessary in order to make experiments regarding the health of a testbed or quality of testbed deployment,
- Capability list: the list of capabilities of a resource would give information of what phenomena (e.g. temperature, humidity, etc.) the resource is able to observe.

3.1.3 Requirement Analysis

Table 4 - Analysis of the requirements ant their relationship with Experiment 1

Requirement unique ID	Relationship
01_FR_ACC_Resource_independent_info	Most of the analytics experiments are interested mainly in the data produced in a time window and in a geographic region regardless of the particular resource.
02_FR_ACC_Resources_available	IoT analytics experiments need to have a certain amount of data for the whole duration of the experiment.
03_FR_ACC_Past_Present_info	The analytics we would like to apply is mainly based on time series. They could be offline (historical) and online (historical + real-time).
04_FR_ACC_Discover_info_independent_testbed	Analytics experiments are interested in data coming from a certain virtual entity without considering the responsible testbed.

05_FR_ACC_Dynamic_discovery_resources	The experiments are looking for data that complies with some parameters (e.g. virtual entity characteristics); if new data complies with such parameters, this must be available to the experiment.
06_FR_ACC_Querying_data_compositions	Data aggregation, sensor fusion etc. can be useful, if not even necessary, for analytics
07_FR_ACC_Discover_data_phenomenon	Exploiting the capability of filtering out uninteresting data will be essential for the experiments in terms of performance and of accuracy
08_FR_ACC_List_info_related_measurement	The analytics done by the experiments is mainly based on time-series, so information like timestamp are fundamental. Furthermore, the location of a measurement is necessary.
09_FR_ACC_List_info_related_resource	Our resource-oriented experiments rely on this functionality. Furthermore, this feature is necessary at design time of the observation-oriented experiments.
10_FR_ACC_Get_measurements_single_request	The experiment must run always in the same way regardless of the number of testbeds and the diversity between them.
11_FR_ACC_Request_data_different_ways	Sensor fusion experiments need to deal with different typology of sensors which can provide a different amount of data. E.g. a noise sensor is producing many measurements within a second differently to a light sensor. In order to combine these two kinds of sensor data it should be possible to balance the amount of measurement for each type of sensor by giving a period and/or event-based and/or autonomous notification by the sensor.
12_FR_ACC_Choose_metadata_each_measurement	Filtering out metadata about measurements before getting the data will improve the performance of the experiment.
14_FR_ACC_Sample_specific_fractions_data	Minimize the number of data retrieved will improve data analytics experiments performance.
15_FR_ACC_Discover_resources_by_characteristics	Discovery of resources is a necessary functionality leveraged in the experiment design phase in order to have a vision of the resource capabilities available in FIESTA-IoT.
16_FR_ACC_Discover_measurements_by_metadata	Experiments will request measurements by specifying a time window and/or a geographic area
17_FR_ACC_Experimenter_provide_feedback	Retrieving feedback to the platform about the malfunctioning of a resource could improve performance (in terms of accuracy) of the experiments.
18_FR_SEC_Testbed_authentication_mechanisms	Experiments will authenticate before the start of its analyses
20_FR_SEC_Experimenter_single-sign-on	Experiments are intended to be online and real-time, so it is necessary to minimize the time and bandwidth consumed for authentication during execution.

21_FR_SEC_Tool_manage_users	Before the start of the experiment, the experimenters need to have the rights to execute the experiments
22_NFR_ACC_Distinguish_type_of_data	Since our experiment is interested in data it is necessary to be easily able to extract data from the information received
23_NFR_ACC_Page_in_subrequests	Crunching too much data at the same time might create a threat to the stability of the systems (network and computation).
24_NFR_ACC_Tools_planning_auto_tasks	Running scripts (e.g. data aggregation) directly in FIESTA-IoT platform will optimize network usage.
25_NFR_ACC_Set_response_Max_size	Crunching too much data at the same time might create threats to the stability of the systems (network and computation).
26_NFR_ACC_Response_adapt_dynamics_resources	Experiments expect always updated and actual data.
27_NFR_ACC_Register_new_resources	Adding new resources and so new data, will be only beneficial to the analytics.
28_NFR_ACC_Response_delay_controlled	Some IoT analytics experiments need to rely on low latency measurements due to its real-time natureness.
29_NFR_ACC_Data_provided_as_requested	The experiments rely on the assumptions that FIESTA-IoT is retrieving exactly the requested data and in the form that the data was requested (leveraging but within the boundaries of the offered API).
30_NFR_ACC_FIESTA_well_documented	A good documentation is necessary for experiments development.
31_NFR_ACC_Describe_IoT_services_and_applications	Experiments need to deal only with a high-level interface that is exposing access to data and resources.
32_NFR_ACC_Provide_dev_deploy_manag_config_tools	Tools would ease the development of analytics experiments.
33_NFR_ACC_Scalability_data_collection	IoT analytics is meant to work with huge amounts of data.
34_NFR_PLA_Process_non_responded_requests	A failure on a data request should be detected and handled in order to be notified to the experiment.

35_NFR_PLA_Manage_resources_in_query_or_experiment	This requirement is necessary during the preparatory phase but also for analytics experiment like malfunctioning sensors.
36_NFR_PLA_Resources_produce_different_measurements	IoT deployments often use sensors array, i.e. a device carrying different types of sensor.
38_NFR_PLA_Diff_types_requests_processed	see 11_FR_ACC_Request_data_different_ways.
39_NFR_PLA_Info_testbed_agnostic_way	The experiments are based on measurements, from wherever these measurements are coming from.
40_NFR_PLA_Process_feedbacks	The system should be able to quarantine or at least mark the resource if malfunctioning.
41_NFR_PLA_Minimise_processing_delay	Low latency response is necessary for real-time IoT-analytics
42_NFR_PLA_Data_generated_from_processing_info	Data resulting from some processing of the FIESTA-IoT platform might be of interest for further analysis. So FIESTA-IoT would enable the composition of experiments.
43_NFR_PLA_Optimise_computational_assets	Optimizing the usage of computational assets can be only beneficial from the point of view of performances of experiments.
44_NFR_PLA_Prioritization_of_services	Having a prioritization for real-time analysis will improve the quality of results if not even make real-time analysis feasible.
45_NFR_PLA_Orchestration_of_resources	Orchestrating the data flow coming from the resources is needed by our experiments because we are expecting all the data available in FIESTA-IoT platform that are matching with our request. If the data is coming from different resources, different testbeds or even outcome of other experiments, FIESTA-IoT is required to orchestrate all of them.
46_NFR_PLA_Homogeneous_classification	The resource-oriented experiments need to understand clearly the information about the devices and therefore about the related resources. For that reason the classification among them needs to be homogenous.
49_NFR_PLA_Reliable_time_sync	Since time stamping measurements is necessary for our experiments, having them synchronized regardless of the time zone is also necessary.
50_NFR_PLA_FIESTA_scalable_extensible_upgradable	The same experiment should be able to work without considering the number of testbed attached to the FIESTA-IoT platform.

51_NFR_PLA_FIESTA_highly_reliable	Results from analytics that are critical must be confident on the platform where it relies on.
52_NFR_PLA_Elasticity_abundance_computational_assets	Abundance of computation and storage assets can offer room for cache capabilities and thus decreasing latency, further the capability of running script directly in the platform.
53_NFR_PLA_Execution_concurrent_services	The execution of an experiment must not be affected by the presence of a second experiment overlapping with the first in terms of resource/service usage.
54_NFR_INT_Interop_between_fiesta_testbeds	The experiments need to receive data from the testbed and this can be achieved only if interoperability between testbeds and the FIESTA-IoT platform is satisfied.
55_NFR_INT_App_develop_integration_resources	The experimenter is confident of the completeness of the data coming from testbeds that are integrated by FIESTA-IoT platform
58_NFR_MEA_Link_measurements_resources	The capability of associating the measurement with the resource would make feasible, for instance, malfunctioning detection.
59_NFR_MEA_Link_measurements_metadata	This feature can be useful at experiment development.
60_NFR_MEA_Measurements_provide_metadata	Metadata such as timestamp and geotag is necessary for IoT-analytics.
61_NFR_MEA_Measurements_independent_resource	Most of analytics experiments are mainly interested in the data itself and not the resources.
62_NFR_MEA_Measurements_provide_phenomenon	IoT analytics experiments need to know exactly what kind of data is going to be analysed.
63_NFR_RES_Link_resource_tesbed	In order to apply some resource-oriented experimentation it is necessary to have the possibility to link a resource to a testbed.
64_NFR_RES_Resource_provide_characteristics	Some statistical analytics on resources can be done only if characteristics on them are available.
65_NFR_RES_Resource_identified_code	Experiments, like resource malfunctioning detection, can work only if ambiguity are avoided
66_NFR_RES_Calculate_reputation_resource	Some critical analysis would run only on the resources with best reputation.
68_NFR_SEC_Support_certification_authority	Since the experiments are unaware of the testbeds federation, the authentication needs to be done only once.
70_NFR_SEC_Different_profile_types	Have predefined type would help and speed up experiment development

71_NFR_SEC_Privacy_collected_data	Nowadays privacy is a very important topic.
72_NFR_SEM_Semantic_annotations_data	Semantically annotation of data is enabling the portability of the experiments. Without that our experiments need to be rewritten for each testbed.

3.1.4 KPIs

At experimentation time, the data produced by different resources and, more generally, by different testbeds are coming seamlessly through FIESTA-IoT, without any further connection to other interfaces than the ones exposed by FIESTA-IoT. Thus, the following KPI:

- If the incoming measurements, due to a query or a subscription, have been taken from 2 or more testbeds, this KPI is satisfied.

An instance of an experiment indicating the health of a city would be the quality of the resource deployments. In other words, such an experiment would check if there are enough sensors of a certain type deployed and working in a certain geographic area of the city. For such an experiment, two KPIs need to be satisfied:

- 1 or more notifications about changes of resources' availability have been notified to the experiment.
- 1 or more metadata are delivered together with resources information.

Furthermore, in order to quickly proceed with experiment development, it is useful to have already available datasets, without creating a dataset from scratch by getting real-time data. This would need the means to request historical datasets. Thus, the following three KPIs:

- 2 or more measurements, observed from the same device at different time, are returned in the historical query response.
- (optional) If a maximum amount of measurements is specified in the query request, the response will be composed of no more than such amount of measurements.
- (optional) In case of a dataset bigger than the specified limit, the rest of the dataset must be accessible with further requests (e.g. by specifying an offset or a page).

Once the datasets have been analysed and the experiments implemented, the real-time experiment can begin getting live data. Our real-time experiment bases its computation, tautologically, on real-time data, which implies that a measurement needs to be notified to the experiment as soon as the measurement is available.

Our real-time experiments will therefore rely on a subscription-notification system: at experiment initialization, a subscription will be sent to the FIESTA-IoT platform and then compute the incoming real-time data flow. Thus, the following KPI:

- If 1 or more measurements are notified to the experiment after a data subscription, this KPI is satisfied.

If some testbeds, FIESTA-IoT users or FIESTA-IoT maintainer defines some data composition these virtual entities need to be accessible by the experiments. Not only the raw data but also the composite data is used in data analytics. Thus, the following KPI:

- If an experiment receives 1 or more measurements computed by another experiment, this KPI is satisfied.

City health indicators are often interested only in certain phenomena and not in the full data available. For example, an energy consumption anomaly detection indicator would be interested in getting informed when the power consumption of a plug is over a threshold. Furthermore, it will relate such information with temperature and CO2 measurements for firebreak detection. Thus, the following KPI:

- If the experiment receives no observations below a threshold specified by the experimenter, this KPI is satisfied.

Another useful city indicator would be a testbeds ranking by importance (a parameter involved in the importance computation would be for instance the coverage of a testbed), in order to know strategically which testbed needs to be more protected for keeping a minimum threshold of quality of deployment. In order to do this it is necessary to know the responsible testbeds for each resource discovered. Thus, the following KPI:

- (optional) If all the resource descriptions returned to the experiment are including information about testbed belonging of the resource, this KPI is satisfied.

Real-time data analytics need to deal only with the meaningful part of a measurement, in order to optimize the bandwidth usage and to lighten the workload by computing less data. For this reason, it is useful to choose which metadata the experiments receive. Thus, the following KPI:

- If the experiment is receiving no metadata more than the ones specified in the query/subscription, this KPI is satisfied.

In order to run the data analytics experiments it is needed to be authenticated in order to have the proper rights to access data. Once authenticated, the data should flow to the experiments without any other overhead of message exchanging, in order to keep the latency low. Thus, the following KPI:

- If the experiment is receiving 2 or more data message after the authentication, this KPI is satisfied.

3.1.5 Required Tools

- Authentication tools:
    - User management: in order to let us make our experiments we need to be identified as a FIESTA-IoT user and authorized to use the platform. This implies a user management tool for creating the user and access FIESTA-IoT with such credentials.
    - Rights access management: once the user is created there should be a way to define the rights of the user. In other words what the user is allowed to do.
    - Authentication procedure: the FIESTA-IoT user needs to authenticate in order to acquire all his rights on the platform.
- One-time data query tool: retrieving data from FIESTA-IoT via a query needs to be easy. At the same time it is required that the experimenter has a capability to express the simple but also the most complex query, most likely with the same tool. The tool must also support parameters for the query like time window (for historical query) and geographic area. The tool should also provide the means to select which metadata needs to be retrieved in order to minimize the amount of data exchanged and processed. Furthermore, the data could be requested if representing a phenomenon and the tool is required to support the definition of phenomena.
- Continuous data query tool: in order to apply online analytics, it is necessary to be notified of new observations taken. The data flow is started with a single initial subscription to data, after that, the data flows to the experiment as soon it is available. The way how the experimenter defines which data he is interested in needs to be the same as for the one-time query tool with the same capabilities.
- One-time resource discovery tool: the experimenters need to have the possibility to discover resources with a tool similar to the one-time query tool.
- Continuous resource discovery tool: similar to the previous tool but with continuous updating about new resources available, resources not anymore available and resource description updates.
- Experiment definition tool: it is necessary to have a tool for easily defining experiments. This tool need to be as powerful as possible in order to have the freedom to maximise the usage of the functionality available. This tool is intended to describe the lifecycle of the experiment.
- (optional) Experiments GUI: a good opportunity for the experimenter would be the usage of a GUI that enables the experimenter to visualize the status and the results of experiments. Furthermore, it would be useful to have the possibility to define the experiments graphically in order to ease the first usage of the FIESTA-IoT platform by new users.
- Experiments monitoring: In addition to the GUI, an API for accessing and monitoring experiments needs to be in place for enabling the experimenters to reuse FIESTA-IoT platform with other platforms or applications.
- Data composition script tool: aggregating data is fundamental for lightening the computation load of a real-time experiment. Furthermore execute aggregating script in the FIESTA-IoT would minimize the bandwidth consumption. For such rationale a tool offering the possibility to define aggregating script and deploy that in FIESTA-IoT is required.

3.2 Dynamic Discovery of IoT resources for testbed agnostic data access

3.2.1 Experiment description

This experiment will exploit the FIESTA-IoT meta-directory where resources (i.e., data streams) will be registered in order to dynamically discover and use data from one or more testbeds (in a seamless and testbed agnostic way). This meta-directory will be part of the FIESTA-IoT EaaS infrastructure, in order to enable dynamic discovery and access to resources from multiple testbeds. The meta-directory will also work with common templates to define, identify and classify resources from different environments or providers that can become similar information sources or actuators. This functional component and the way it provides access to registered resources and their attributes takes an important step forward to a common semantic definition of testbed's components.

The experiment will focus on the dynamic acquisition and processing of weather information (based on temperature, pressure, wind speed, UV, and humidity data streams), towards consolidating and visualizing data from multiple locations. The experimenter will be able to dynamically specify the locations/areas for which data will be collected, as well as e.g. the physical phenomena he/she wants to retrieve. The specification of these areas might lead to the simultaneous data acquisition from one or multiple testbeds, which offer specified data streams e.g., based on the weather stations that these testbeds deploy. The added value of the experiment is that the experimenter will be able to dynamically select and process data streams needed for the forecasting process, regardless of the testbed providing these data. The experiment will be easily expandable and readily deployable over any number of additional testbeds that will be integrated/federated as part of the infrastructure.

One of the main goals of this project is to provide a seamless access across different testbeds, disregarding their underlying architecture or communication model. For a very first practical experiment, we propose a simple approach, which will consist in the acquisition of a single data type. While simple, it demonstrates the potential of abstracting the data sources so that an experimenter can get the information from any platform provider. Hence, this approach can be expanded to any kind of data in a straightforward manner.

Since in this initial stage of the project we only rely on four legacy deployments (namely, Comm4Innov, KETI, UNIS and UNICAN to extract data), we have gathered in Table 5 the different types of resources that are supported by the different partners' platforms. As can be easily inferred, there is a clear convergence in those sensors that are used to measure weather and environmental phenomena, such as temperature, humidity, luminosity or noise, among others. It is worth highlighting that this can be easily transferred to a forthcoming demonstrator. Besides, it is deemed necessary to show that we have actually adapted a common and seamless interface, so that we can acquire the observations from the different platforms under a unique instance, which is planned to follow a Domain Specific Language (DSL). A good way to represent all this stuff could be through a web page (or smartphone app) split into three areas, showing a map with the information of a particular resource (or a heat map which spans a bounded surface), and two different log sections, showing both *testbed-agnostic* and *specific* requests. In addition, the visualization of the data streams might be interesting, but we can leave it as a secondary feature.

Table 5 - Comparative between the different physical entities (i.e. resources)

Resource/testbed	*C4I*	*KETI*	*UniS*	*UNICAN*
Environmental (e.g. temperature)	✓	✓	✓	✓
Pollution (e.g. CO_2)	✓	✓	✓	✓
Presence control		✓		✓
Traffic monitoring				✓
Parking guidance				✓
Tags			✓	✓
Geo-localization			✓	
Detector (motion/contact)			✓	
Power consumption (e.g. smart sockets)			✓	✓
Crowdsensing/participatory sensing			✓	✓
Mobile sensing (e.g. buses, taxis, etc.)				✓

An illustrative way to outline the experiment is shown in Figure 7, in which a user wants to retrieve the temperature in a concrete area (e.g. extracted from his/hers smartphone's GPS coordinates). Following a top-down approach, the upper side of the figure represents the generation of an arbitrary *testbed-agnostic* request from an external experimenter, which will be delivered to the FIESTA-IoT's core.

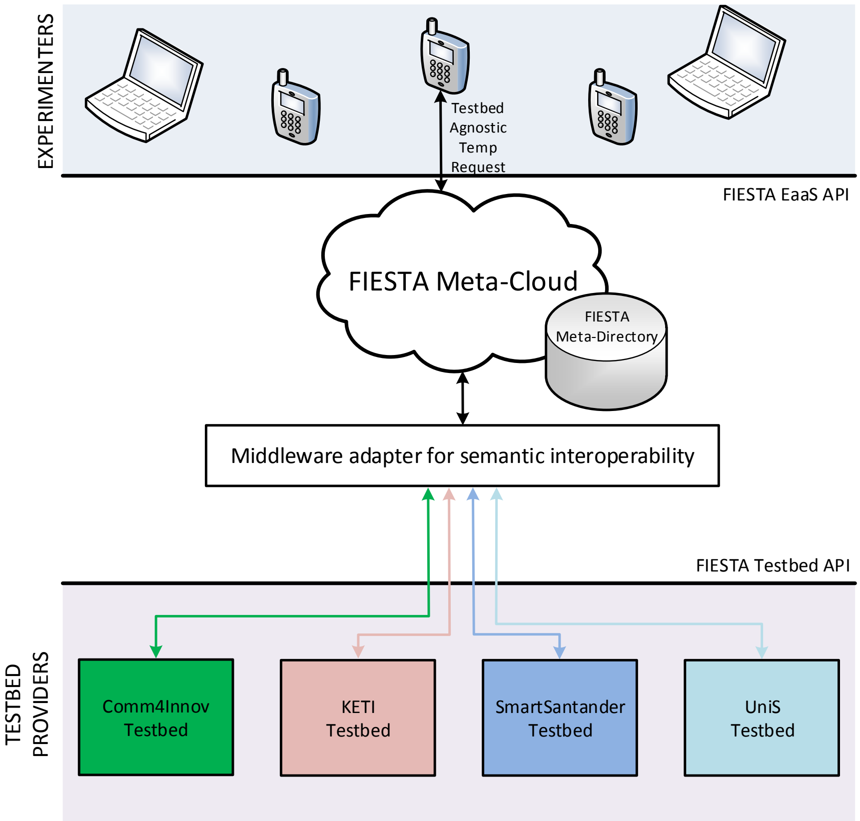

Figure 7 - Seamless access use case schema based on the (preliminary) architecture defined in FIESTA-IoT

Depending on how the FIESTA-IoT architecture and FIESTA-IoT enablers are defined, this testbed-agnostic query will be handled internally at the FIESTA-IoT Meta-Cloud resulting in the requested data-set or data-stream being received at the experimenter endpoint.

In this sense, in Figure 7 we have intentionally left out detailed features of the Meta-Cloud (e.g. federation, semantic annotation, security, etc.) and summed up the FIESTA-IoT *middleware* as an entity that would take the role of "translator" between the *testbed-agnostic* requests and tailor the output messages to the different *testbed-specific* format. However, other alternatives such as pointing the experimenter to *testbed-specific* interfaces would also be possible.

Last, but not least, once the architecture is better defined and the corresponding enablers are ready, the experiment will embrace all of these previously overseen detailed functionalities. Moreover, in order to offer a brief sneak-peek of what we aim to do for this experiment, we show in Figure 8 a mock-up representation of the web page we plan to implement, mainly the home page. As can be extracted from the figure, it will be an interactive map, in which experimenters can both select those VEs they want to interact with; in addition, they will (at the same time) visualize information received from the underlying testbeds. Besides, several ways to represent/process the information will be also available. For further information regarding the tools that will shape this experiment, the reader might refer to Section 3.2.5.

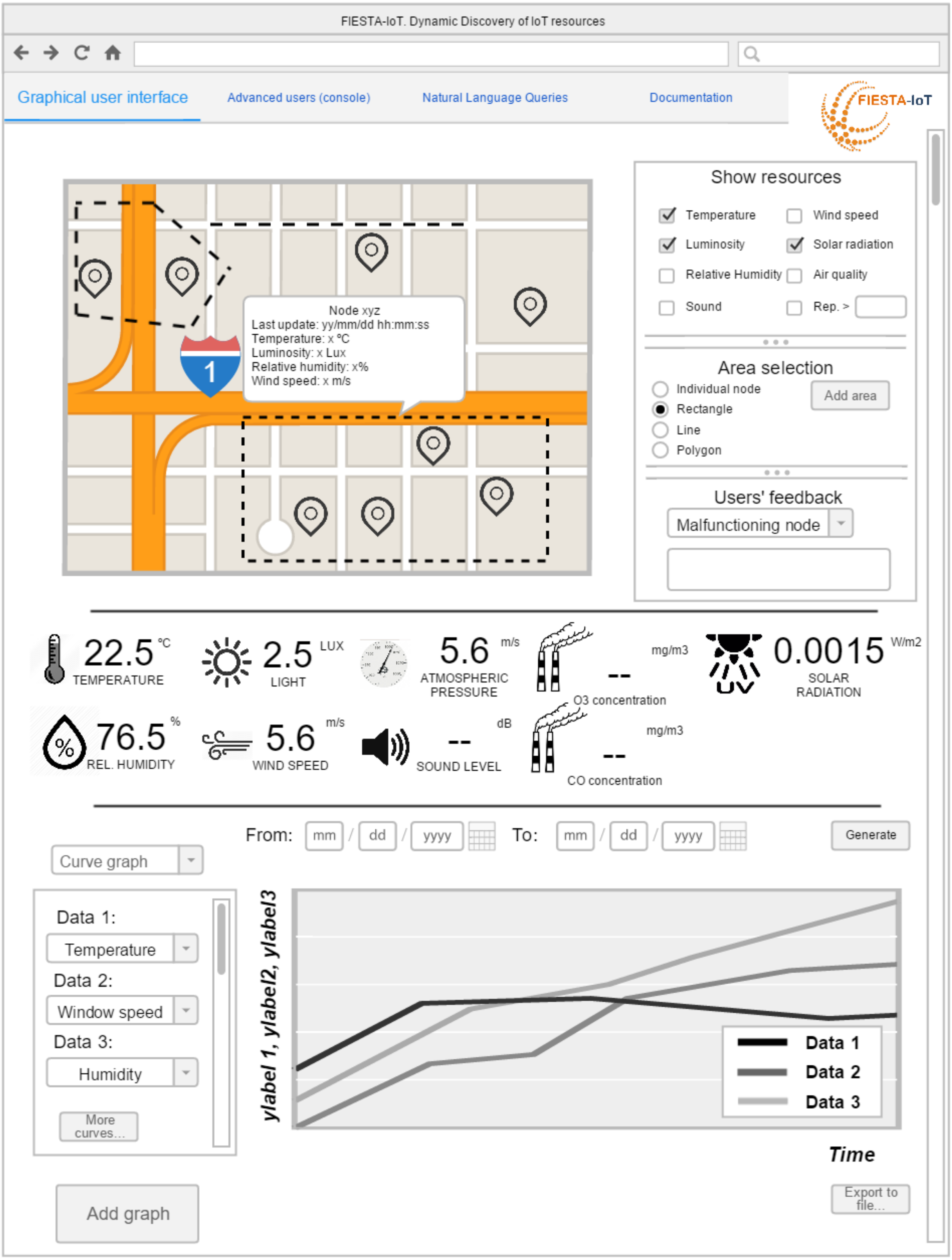

Figure 8 - Initial mockup of the proposed web page

3.2.2 Specification of Datasets

During the execution of this experiment, different data types are to be gathered from the underlying federated testbeds. We list below the information which should be delivered to the experimenter (following a bottom-up direction):

- *Phenomena & unit of measurement*. Without a shred of doubt, these shape the cornerstone of the experiment. Focused on the weather and environmental domain, experimenters will retrieve, through a unique interface, the information gathered from the different sensors that are deployed throughout the various federated testbeds that will shape the FIESTA-IoT meta-cloud. Namely, an initial set of physical phenomena is considered to be part of the experiment:
    - ✓ Ambient Temperature
    - ✓ Relative Humidity
    - ✓ Wind Speed & Direction
    - ✓ Luminosity
    - ✓ Solar radiation
    - ✓ Ground Temperature
    - ✓ Soil Moisture Tension
    - ✓ Atmospheric Pressure
    - ✓ CO_2 concentration

  Nevertheless, this selection is not closed and future sensors, which measure different physical phenomena, will be welcome.

- *FIESTA-IoT Resource unique ID*. In order to provide a testbed-agnostic experiment, each resource of every registered testbed federated at the FIESTA-IoT platform will get a unique identifier (FIESTA urn – uniform resource name). This way, an experiment will be able not (by default) to differentiate or extract the actual underlying testbed that actually owns the physical resource.
- *Location*. As can be easily inferred, every physical resource must have a physical location, normally gathered from a built-in GPS device attached to each device (the position can be “manually” assigned too). In most cases, a “latitude/longitude” pair will be generated (there are more advanced geo-spatial notations<ref to GeoJSON>, though).
- *Timestamp*. Timing information will have an essential role for experimenters. Each time a VE catches/generates a measurement, it will go together (~atomic operation) with the time instant it was taken. Thus, historical data-based queries could be run and experimenters will be able to gather statistical information from the previously stored measurements.
- (Optional) *Local time zone*. Since different testbeds might be placed in different time zones, it might be necessary to explicitly transmit their local time zone so that experimenters can be able to distinguish between the various platforms. As an illustrative example, to compare the temperature measured at sunrise between Santander and Seoul.
- (Optional) *Resource description*. Apart from the raw data extracted from the physical devices/sensors, experimenters might need a minimum level of knowledge about some of their most noteworthy properties, such as its

exposed services, sensors accuracy, sampling frequency, indoor/outdoor location, etc.

- (Optional) *Legacy testbed resource ID*. Some experimenters might want to know the underlying testbed that actually owns a determined resource.
- (Optional) *Management info.* Additional metadata will be appended to share management-based information. Among its main fields, includes (new elements might be added in the future): hardware, firmware version, date of registration, last update, last measurement, etc.

3.2.3 Requirement Analysis

Taking into account the challenges arisen from the design and implementation of this experiment, Table 6 resumes the bindings between the requirements and their relationship, direct or indirect, with this concrete experiment, led by UNICAN.

Table 6 - Analysis of the requirements ant their relationship with Experiment 2

Requirement unique ID	Relationship
01_FR_ACC_Resource_independent_info	Experimenters will graphically select those VEs with which they want to interact.
02_FR_ACC_Resources_available	Once the experiment is initiated, all available (and filtered from user's preferences) resources will be shown in the map.
03_FR_ACC_Past_Present_info	Last/current and historical values will be fully supported by the experiment, allowing the experimenter to select data in multiple time windows.
04_FR_ACC_Discover_info_independent_testbed	The experimenter will not be able to distinguish the original source (i.e. testbed) of the underlying resource. From his/her eyes, FIESTA-IoT will be a single meta-testbed solution.
05_FR_ACC_Dynamic_discovery_resources	The platform will show all available resources to the experimenter. Management tasks (e.g. resource registration, state monitoring, etc.) will have an essential role, keeping the information updated regarding the underlying testbeds' resources.
06_FR_ACC_Querying_data_compositions	Advanced operations (e.g. cross-testbed statistical data, etc.) will be supported, leveraging a new level of flexibility to the experimenter.
07_FR_ACC_Discover_data_phenomenon	The experiment's built-in web page will allow experimenters to filter the output information (e.g. phenomena, thresholds, etc.) according to their preferences.
08_FR_ACC_List_info_related_measurement	Multiple levels of information regarding the measurements can be shown: basic (only raw observations gathered from the resources) or advanced (e.g. miscellaneous metadata).
09_FR_ACC_List_info_related_resource	Multiple levels of information regarding the resources can be shown: basic (type, timestamp and location, phenomena and

	unit of measurement) or advanced (e.g. type of sensor, level of precision, etc.)
10_FR_ACC_Get_measurements_single_request	Data belonging to the FIESTA-IoT federation will be exposed in a unique format, disregarding the underlying FIESTA-IoT-compliant testbed.
11_FR_ACC_Request_data_different_ways	The experiment will handle either the last values gathered from the sensors or the historic ones that have previously been stored into the meta-cloud.
12_FR_ACC_Choose_metadata_each_measurement	The experimenter might tweak his/her preferences in order to customize the metadata he/she wants to retrieve.
15_FR_ACC_Discover_resources_by_characteristics	Experimenters can choose the resources they want to select in a number of ways (e.g. physical VE, space, time, quality, etc.) .
16_FR_ACC_Discover_measurements_by_metadata	Based on experimenter's configuration, a customized resource discovery/harvesting will be supported, based on his/her preferences.
17_FR_ACC_Experimenter_provide_feedback	The experiment might take advantage of experimenters usage and receive active feedback, e.g. detect anomalies, broken resources, etc.
18_FR_SEC_Testbed_authentication_mechanisms	Experimenters have to be authenticated and authorized before having access to the experiment.
20_FR_SEC_Experimenter_single-sign-on	A simple credential system will be used to ease users access to the experiment and reduce the overall latency.
21_FR_SEC_Tool_manage_users	From the user perspective, it will be easy to sign up/in; on the other hand, from the administration side, it will be possible to manage/store/modify the users' credentials/privileges.
22_NFR_ACC_Distinguish_type_of_data	Information generated (i.e. measurements, resource description, metadata, etc.) by FIESTA-IoT towards experimenters will be intuitive and easily to parse.
23_NFR_ACC_Page_in_subrequests	Experiment queries/responses might be split (fragmented).
24_NFR_ACC_Tools_planning_auto_tasks	The usage of user scripts might be useful for generating automatic tasks.
25_NFR_ACC_Set_response_Max_size	There must be a size limit for response messages, since if they are too long, they might collapse the system.
26_NFR_ACC_Response_adapt_dynamics_resource s	The resources displayed on the map will dynamically change, depending on the events that might happen during the experiment (e.g. new nodes, lost connections, etc.).
27_NFR_ACC_Register_new_resources	Third parties that become part of the FIESTA-IoT federation will be able to register new resources. Upon the generation of a new resource/VE/service, the experiment will be accordingly updated, showing the new stuff to the experimenters.

28_NFR_ACC_Response_delay_controlled	One of the main goals of the platform is to generate low latency responses.
29_NFR_ACC_Data_provided_as_requested	Experiment's responses must tailor to experimenters' requests. In other words, the experiment has to be flexible enough to allow complex queries from advanced or skilled experimenters.
30_NFR_ACC_FIESTA_well_documented	Having good documentation is essential for a good understanding and a quick adoption by experimenters.
31_NFR_ACC_Describe_IoT_services_and_applications	Experimenters will not directly contact underlying testbeds; every query/response will go through a high level API between the experimentation plane and the FIESTA-IoT meta-cloud architecture.
32_NFR_ACC_Provide_dev_deploy_manag_config_tools	The experiment will support a degree of flexibility, thus allowing users to execute their own "tweaks".
33_NFR_ACC_Scalability_data_collection	Experimenter requests might grow enormously; the platform must be capable of handling such queries without overloading the system.
34_NFR_PLA_Process_non_responded_requests	The execution of an experiment might fail and the FIESTA-IoT platform must detect and react upon this random and undesired event.
35_NFR_PLA_Manage_resources_in_query_or_experiment	The experiment will be conceived as a visualization tool that makes an easy and friendly use experience for the experimenter.
36_NFR_PLA_Resources_produce_different_measurements	Measurements gathered from different resources can be parsed and filtered, catching the information relevant to the experimenter.
38_NFR_PLA_Diff_types_requests_processed	Different types of service will be provided (e.g. last values, historical data, asynchronous subscriptions, etc.).
39_NFR_PLA_Info_testbed_agnostic_way	The experimenter will not be able to differentiate the testbed provider that owns particular resources.
40_NFR_PLA_Process_feedbacks	The platform can exploit feedback generated by experimenters (see 17_FR_ACC_Experimenter_provide_feedback).
41_NFR_PLA_Minimise_processing_delay	The time taken to process a query has to be reduced as much as possible in order not to annoy the experimenter.
42_NFR_PLA_Data_generated_from_processing_info	Advanced processing of data (e.g. statistical information, graphical outputs, etc.) will be available as part of the experiment.
45_NFR_PLA_Orchestration_of_resources	The experiment will support an additional processing layer capacity to orchestrate and optimize experimenters' requests.
46_NFR_PLA_Homogeneous_classification	The experiment will show the info of the available resources In an homogeneous manner, where the underlying testbeds are seen as "black boxes".
49_NFR_PLA_Reliable_time_sync	The experiment will stick to different time

	zones, depending on the underlying testbeds.
50_NFR_PLA_FIESTA_scalable_extensible_upgradable	New testbeds will become part of the FIESTA-IoT federation; then, their resources will automatically be registered and displayed in the map.
51_NFR_PLA_FIESTA_highly_reliable	The experiment must accomplish a minimum level of reliability.
53_NFR_PLA_Execution_concurrent_services	Multiple experiments (with slight nuances among the different queries) might be executed at the same time, requesting the same IoT services (e.g. subscription based notifications); for that purpose, the platform will aim at optimizing the response generation (e.g.by reusing measurements instead of repeating them).
54_NFR_INT_Interop_between_fiesta_testbeds	Testbeds will communicate with the FIESTA-IoT's core through a common interface, thus enabling, from an experimenter point of view, a testbed agnostic access
55_NFR_INT_App_develop_integration_resources	Services run under this experiment might not work as standalone executions.
58_NFR_MEA_Link_measurements_resources	The experimenter will be able to get all the information of the resource that generated an observation/measurement.
59_NFR_MEA_Link_measurements_metadata	The experiment will be able to connect different measurements from their metadata (e.g. same VE, different time).
60_NFR_MEA_Measurements_provide_metadata	Aside from the raw observations (i.e. phenomena and unit of measurement), additional metadata will be attached to the measurement (e.g. timestamp, location, precision of the sensors, firmware version, etc.).
61_NFR_MEA_Measurements_independant_resource	The experiment will follow a data centric approach, disregarding the resources that generate the information.
62_NFR_MEA_Measurements_provide_phenomenon	Every observation must be linked to a concrete phenomenon (and tied to a unit of measurement).
63_NFR_RES_Link_resource_tesbed	Experimenters might want to retrieve the ID of the testbed that owns a concrete resource. Thus, it will be part of the metadata included in a measurement.
64_NFR_RES_Resource_provide_characteristics	Low-level resource info will be provided to (advanced) experimenters (e.g. static/mobile node, indoor/outdoor location).
65_NFR_RES_Resource_identified_code	Every resource must have a unique ID (primary identifier).
66_NFR_RES_Calculate_reputation_resource	The experimenter might select those resources with a minimum level of reputation/reliability.
67_NFR_RES_Experiment_participants_known	The experiment will gather statistics from participating resources which are part of them (i.e. statistical usage analysis, logging, etc.).
68_NFR_SEC_Support_certification_authority	Experimenters will be able to get

	measurements from VEs of FIESTA –IoT federated testbeds.
69_NFR_SEC_Verify_authorise_user_actions	Experimenter's access to the FIESTA-IoT architecture (i.e. run the experiment) will be validated beforehand.
70_NFR_SEC_Different_profile_types	The experiment will categorize different types of users/experimenters through different profiles.
71_NFR_SEC_Privacy_collected_data	The experiment must ensure the user/experimenter privacy, as well as the data gathered from the federated testbeds.

3.2.4 KPIs

The development of this experiment fits some of the KPIs that were defined to elicit a scope of best practices. Moreover, these indicators do have a tight relationship with the requirements introduced in the previous section, since they will be used to validate a number of them. We list below the main KPIs that are involved in the definition of the experiment and must be accomplished once its development is concluded.

- The platform will foster both first and third party testbeds as new FIESTA-IoT federation members. Their available physical resources/IoT services will be dynamically added to the FIESTA-IoT meta-testbed, which will expose these elements as part of FIESTA-IoT infrastructure. Numerically speaking, minimums of four different datasets (from different testbeds) are going to be accessible through the FIESTA-IoT meta-solution.
- The experiment will focus on a testbed-agnostic access to data, where an experimenter will not be able to make any distinction between underlying testbeds. In a nutshell, every query sent by an experimenter will follow a unique interface which will be parsed and interpreted by the FIESTA-IoT core, which will be in charge of mapping it to the different testbeds' interfaces. In other words, in the eyes of an experimenter, he/she will only see a single large scale distributed platform/meta-testbed. Following the above indicator, this single interface will allow experimenters to access to more than three different testbeds, i.e. the ones which will become part of the FIESTA federation.
- As long as a new resource or service is registered in any of the underlying testbeds (e.g. a new parking sensor has been activated in Santander), the subjacent platforms will update FIESTA-IoT's resources/IoT services repositories. Thus, experimenters will seamlessly perceive a dynamic discovery of resources/IoT services/actuation capabilities. Talking about figures, more than 5,000 new resources/VEs/IoT services will be registered as part of the FIESTA infrastructure.
- There might be cases in which experimenters want to retrieve, in a simple query, information belonging to different platforms. FIESTA-IoT must be capable of generating cross-testbed responses, following a common and unique message format, fulfilling a complete Experiment as a Service (EaaS) approach. Talking about figures, we stick again to the minimum of four different platforms that are to be encapsulated in a single response.

- Besides the default graphical interface used to guide experimenters among the different features embraced by the experiment, which allows users to select from a set of predefined services (e.g. filter from resources' properties, choose from a number of default output options, etc.). Moreover, it will offer the possibility of executing proprietary variations/tweaks of the experiment, carried out by advanced or more skilled experimenters (or even brand new ones), by manually introducing their own scripts/commands, through an Application Program Interface (API) defined for that purpose. Summing up, there will be two different means to run the experiment.
- In addition to the obtaining of the typical last/current values measured by the various resources, additional services are to be provided to experimenters. For instance, the possibility of retrieving historical values previously stored at the FIESTA-IoT meta-cloud (or the underlying testbeds, though), as well as the subscription to determined services, provision of a set of processed statistical values (e.g. average, variance, maximum, minimum, etc.), among many others.
- Only registered (and validated) experimenters will be granted and be able to run the experiment. They will have to sign up through the experiment submission portal that will be defined in the scope of the project. This way, the experiment will be tied to some of the security issues that are being studied throughout FIESTA-IoT's lifetime. Let us set 50 as the minimum number of users registered to run this experiment.
- Different levels of users: basic, advanced, administrator, etc. enable different types of access to the underlying testbeds. Basic access will be tailored to people who only want to play around with the experiment and do not aim to go beyond; hence, they will be only able to use the set of default tools provided by us. Since they will not introduce new information to the system, everyone is welcome to become an experimenter (they must sign up into the experimentation portal, though). In an upper level we find people with a more technical profile and with the capacity to build their own experiments using the one we propose here as a basis and initial reference. In this case, they will be able to submit their brand new experiments through the aforementioned experiment submission portal. Last, but not least, it is deemed essential to have an administrator role, who will be the responsible(s) of managing the activities carried out by users (including the control/agreement on new experiments). Therefore, for this concrete experiment, we will handle up to three levels of users.
- Seize the power of semantics by means of natural language-based queries. A good way to bring closer the experiment to non-technical experimenters is to provide a high level of abstraction; indeed, the lower the use of technical stuff, the better the sensations provoked to the average experimenter. In terms of quantification, we will be able to correctly process a minimum of 50 different sentences/queries, generating accurate and logical responses to each one.

3.2.5 Required Tools

We list the set of tools that will be implemented to deal with the requirements and performance indicators that define the goals and challenges of this experiment.

- *Web page*: As can be easily inferred from the description we made in the previous section, we will mainly rely on the use of a web page (can be also ported to a smartphone app) to address the challenges arisen in this experiment. Figure 8 illustrates an initial wireframe that aims at gathering all the elements and objectives that will shape this pilot, including the dynamic discovery of IoT resources, the abstraction of the underlying testbeds/resources and the different KPIs that were discussed In Section 3.2.4.

  We can distinguish, from the tabs at the top of the page in the figure, up to four different main operations, which are briefly described below:

  - *Graphical user interface*. This first element corresponds to the most basic way to interact with the experiment, in which newbies or not skilled experimenters can easily select and obtain information in a completely visual way. As has been already discussed, the scope of this initial version of the experiment only considers all those resources that pertain to the environment/weather realm. Taking a look at the web's frontend, we can observe that it is split into three main areas, which will be in charge of three different tasks:

    1. *Resource selection and visualization tool*. At a simple glance, we can appreciate the presence of a map, whose functionality is bidirectional and two-fold: on the one hand, it will present to the experimenter all the available resources in a determined zone that were discovered, according to the filters he/she has toggled in/out at the "Show resources" menu, as well as the area that they have selected. It is worth highlighting that experimenters can choose as many VEs (which consist in this concrete case in the tuple *area + phenomena*) as they want. Moreover, experimenters might provide feedback about e.g. malfunctioning nodes, "weird" measurements, etc. Thanks to these notifications, a testbed owner can react and fix those elements that do not work accordingly.
    2. Weather station. In the middle section, we can observe a "weather station-like" output, where we will represent the average, maximum and minimum values of all those VEs that have been previously selected by the experimenter in the above area (one per VE).
    3. Statistical data representation tool. A more advanced output will be also present in this first visualizing tool. By making use of the stored historical values, we will process the information and represent it in a number of manners, e.g. graphically showing the instantaneous values gathered from the requested VEs. For this, an initial, yet scalable, set of pre-compiled options will be available for experimenters. Since users might handle raw data and carry it to another tools (e.g. Matlab, gnuplot, etc.), we also offer the possibility of exporting this data in different formats, like plain text or csv files, among others.

- *Advanced* users tools. The initial set of visualization-based tools that we have described might not offer all of what a determined group of experimenters really need. For this purpose, we not only limit this experiment to the initial set of tools provided by us (as the legacy developers), but we also allow external users to design, tamper or even create their own brand new experiments. Following the API defined to get access to the FIESTA architecture, advanced users will generate new variations of this experiment, tailored from their actual needs, either taking ours as a basis or starting their own from scratch.
- *Natural language query tool.* An interesting added value of the use of semantics can be assessed through the use of a natural language interface that process human-based queries, translating them to semantic queries (e.g. SPARQL). Through an input field, experimenters can launch their questions to the platform and, once they are processed, the experiment will return a response in a visual way (if possible).
- *API documentation tool*. FIESTA-IoT will provide a cutting-edge EaaS vision, allowing third parties to be an important part in the meta-architecture ecosystem, either through creating brand new experiments (above the FIESTA-IoT core) or introducing new testbeds to the federation (bottom side). Hence, it is deemed necessary to have a well-documented section that help developers access the FIESTA-IoT platform and process its data streams. Concerning to this concrete experiment, we will provide information regarding the features and options offered by the graphical user interface, the EaaS API that will act as a communication channel between experimenters and the FIESTA-IoT platform and the DSL, which will define the language to create new experiments.

Whereas experimenters will mainly interact with the graphical user interface, there are a number of subjacent tools that runs under the experiment and are the actual cornerstone of the experiment. We summarize them below and describe their most important functionalities.

- Resource & IoT Service Discovery Tool. The first step upon the execution of the experiment will consist in the discovery of all the resources that are exposed by the different IoT services. Then, from the datasets received from the experimenter (i.e. space, time, phenomena and values filters that will be described in the next point), the query will be processed by the FIESTA-IoT core, which will be in charge of generating a response with the different services that expose those resources that fulfil the experimenters' selection criteria. Amongst these services, the most typical ones will be the request for the last/current values measured by the nodes, the retrieval of some historical data within a bounded time interval or the asynchronous (publish-subscribe) pattern, in which experimenters will directly receive the forthcoming events that are produced in those resources/services of their choice.
- Space/time/phenomena/values/reputation filtering tool. Given that testbeds could have an excessive quantity of resources, experimenters might not be interested in all of them; therefore, they will have to choose the subset of resources/services they want to interact with. This selection might be done in five dimensions: according to their location (space), the moment of their creation (time), the physical phenomena that they are measuring, a

determined value(s) that might work as a dynamic threshold/interval or, depending on the reputation/"quality" of the nodes, admit only those ones which are relied by experimenters.

- Experimenters' feedback tool. This experiment will provide FIESTA with the information generated by external experimenters with regard of the status of the different resources/physical entities. Through this feedback tool, they could notify situations such that they subjectively perceive that something is not working, such as malfunctioning nodes (e.g. those that introduce long latency or even do not respond, etc.), strange measurements, etc. Thanks to this feedback-based management, FIESTA-IoT will detect anomalies in a kind of interactive way, and experimenters will be part of the active management of the meta-testbed.
- User authentication and management tool. Although this one could not be a direct tool in this experiment, security and access control issues have a key role in FIESTA-IoT. Only allowed experimenters will be able to run the experiment; moreover, different levels of access/privileges will distinguish between types of users, e.g. lite, premium, administrators, etc. In addition, FIESTA-IoT will ensure that all the information exchanged between experimenters and the FIESTA-IoT platform will remain hidden to others, hence privacy will be another fundamental pillar of the experiment.
- Natural language processing tool. As hinted above, we need a standalone tool that processes human-based queries, which will be parsed and transformed to semantic queries; after that, they will be further interpreted by the FIESTA-IoT core. To do so, and taking into account this is not a direct requirement of this project, an external and open source solution must be selected to perform this operation.

After the description of the tools, we list the stages that match the general stages that compose an experiment lifecycle.

- *Discovery*. One of the most important objectives of the experiment is to provide a dynamic discovery of the resources/IoT services provided by the underlying testbed that are federated within the FIESTA-IoT meta-testbed.
- *Reservation and configuration*. Once an experimenter has run the experiment (previous submission) and queried for a concrete resource, he/she will be automatically subscribed to the next measurement carried out by the sensors. Besides, feedback from experimenters and reputation-based policies might change the status of the underlying resources.
- *Control*. This concrete set of tools is not focused on the explicit control of the resources, but only the gathering of the information generated by them.
- *Measurement*. Testbed-agnostic and cross-testbed queries are the cornerstone of the experiment. Through them, an experiment is replicable among any of the testbeds involved in the FIESTA-IoT federation.
- *Reporting*. Experimenters can retrieve statistical information of the past event produced in the VEs, either in a graphical way or through raw text files.

3.3 Large Scale Crowd sensing Experiments

3.3.1 Experiment description

Modern smart cities are characterised in part by a prevalence of sensors, either fixed (e.g., municipally-managed environmental sensors) or mobile (e.g., those in mobile devices used by citizens). This opens up the opportunity of crowd sensing the urban environment, for better informing the individual citizens as well as the government, and the benefit of all.

As part of these experiments, experimenters will be able to specify a set of devices and data streams that could participate in a given crowdsourcing experiment (e.g., large scale environmental monitoring). The crowdsourcing experiments will be able to use data streams from participants interfacing to different IoT platforms/providers in order to provide/stream data acquired via their smart phones. FIESTA-IoT will provide an infrastructure where different data streams could be integrated regardless of the testbeds where their data are provided.

As part of the experiment, a living labs approach will be exploited in order to ensure the statistically appropriate representation of the various sensors/data streams in the experiment. The FIESTA-IoT EaaS infrastructure will incorporate novel crowd-sourcing/sensing algorithms into the mobile middleware of the testbeds in order to provide opportunity for dynamically selecting and using IoT resources (e.g., sensors, devices) corresponding to the indicated statistical representation. This will be among the unique capabilities offered by FIESTA-IoT in the scope of the experiment.

A major goal of this experiment is to explore the ability of the FIESTA-IoT platform to manage and execute experiments on the data coming from mobile devices. Further, we would like to explore the ability to ultimately combine data from different co-located testbeds: one present as an app running on citizens' phones, and another present as a web-service that provides access to the data collected by the set of static sensors installed and managed by the city. As an example, as Figure 9, we provide the map of the city of Paris where noise data was collected from mobile devices. The map shows the noise heatmap in the city of Paris.

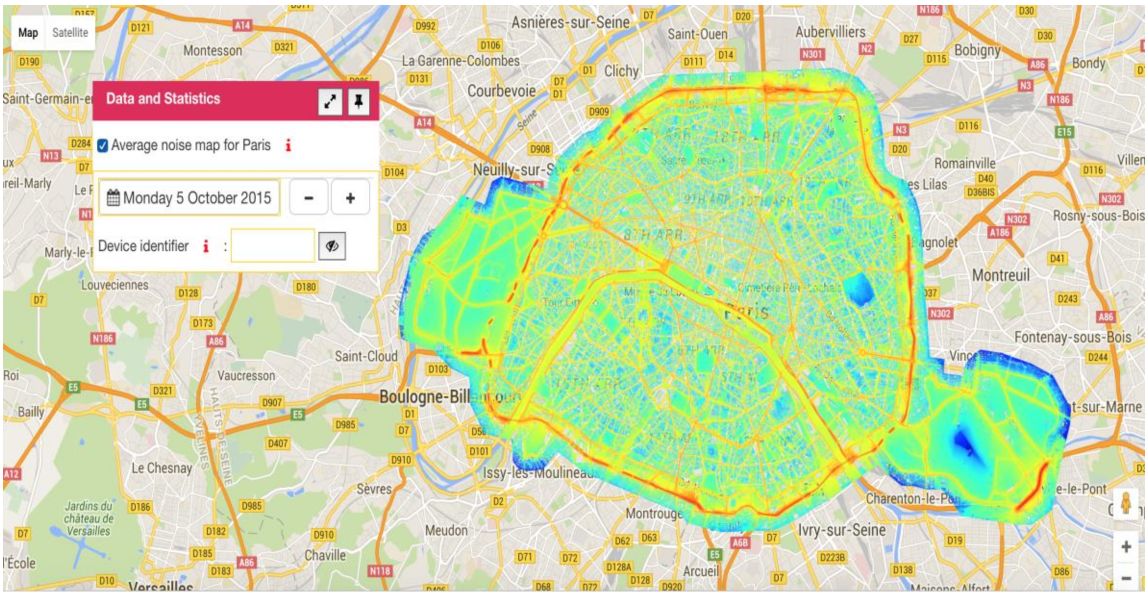

Figure 9 - Initial mockup

3.3.2 Specification of Datasets

The experiment will use the data provided by each device, with the following attributes:

- *Location:* This would be expressed in *(Longitude, Latitude)*, and also accompanied by an *accuracy measure* and the *sensor type* (GPS, 3G, WiFi) used to determine this location
- *(Anonymized) Device ID:* This alphanumeric ID will be used to distinguish data coming from different devices, as well as to correlate data coming from the same mobile device over time. This is especially important since the sensors present on different devices will have different accuracies.
  - *Trade-off between duration of ID persistence and utility of data: We acknowledge that there is a trade-off between the anonymity provided to the users and the utility of the correlation between different data objects*. Further, persisting IDs will be the property that a testbed needs to decide. For Example: a testbed can randomly choose and give ID from a specific set of predefined IDs, while another testbed can randomize (may be) the last "n" digits. Nevertheless, a testbed can also decide to not have any correlation in the IDs. In the above example note that a user specifically is not at all tracked, it is possible that 2 different users have same ID in the dataset for two different data samples. As an experimenter, this would still consider that the data sample was from the same user.
- *Sensed value of environmental data (noise):* This will indicate the noise level in the surroundings of the device, along with the time when the sample was taken.
- *Current Timestamp of the Sensed Value:* This will inform when the value was sensed by the device and will enable temporal aspects related to the value.
- (Optional) *Current state of phone's proximity sensor:* This will inform whether the phone was currently inside a purse/bag (which might affect the quality of sensed data), or uncovered (therefore in a better position to provide noise data)
- (Optional) *Current phone call state:* This will inform whether or not a call is currently active when the noise sample is taken.
- (Optional) *Current physical activity as reported by the phone:* This could be one of resting, walking, running, in vehicle, etc., to further provide context to the data being collected. This context relates to inference of mobility aspects attached to the device. This information is inferred thorough the use of Accelerometer and gyroscope sensor on the phone.
- (Optional) *User-annotation about the source of noise:* The user can optionally provide their input on what caused the noise at any given time.

We put "optional" to certain attributes because, if provided, they provide an added value to the experiment. Although as some of the attributes are not available within current in-house testbeds, these properties might be made available in other testbed that would join fiesta platform in future. Thereby opening an opportunity for the crowdsourcing experiments to evolve. This is also very much aligned with the use of the FIESTA-IoT infrastructure for setup and execution of range of crowd sensing experiments in conjunction with Inria@SiliconValley program and on the basis of the potential integration and use of testbeds from the USA.

3.3.3 Requirement Analysis

Below we analyse the requirements that are and will be satisfied by the experiment.

Table 7 - Analysis of the requirements ant their relationship with Experiment 3

Requirement unique ID	Relationship
01_FR_ACC_Resource_independent_info	The experiment requests data without requiring anything specific about the resource.
03_FR_ACC_Past_Present_info	The experiment can use information generated by the sensors in the past, as well as real-time information, where possible.
04_FR_ACC_Discover_info_independent_testbed	The experiment only cares about data of a certain type from a certain region, without explicitly needing it to be from a certain testbed.
06_FR_ACC_Querying_data_compositions	The experimenter may ask for data that does not directly exist, but may be created using existing samples of environmental data. An example of this would be regional averages of noise values.
07_FR_ACC_Discover_data_phenomenon	The experiment would include queries such as "noise > 70dB".
08_FR_ACC_List_info_related_measurement	As stated in the dataset section above, the metadata is very important.
09_FR_ACC_List_info_related_resource	The annotation of data items depends on the information being available at the resource as well.
10_FR_ACC_Get_measurements_single_request	Multiple environmental testbeds might provide data for a single query.
11_FR_ACC_Request_data_different_ways	The experimenter wants periodic sampling of data, but also to be notified if there is, for example, a noise higher than 85dB.
12_FR_ACC_Choose_metadata_each_measureme nt	The experimenter might want to access certain metadata (e.g., phone sensor type) if the noise value is very high, in order to increase certainty.
14_FR_ACC_Sample_specific_fractions_data	The experimenter can ask for a representative sample of noise values from a city.
15_FR_ACC_Discover_resources_by_characteristic s	The experimenter would like to, for example, see if phones of a certain model provide high noise samples on average then another.
16_FR_ACC_Discover_measurements_by_metadat a	Experimenters may be only interested in data of certain accuracy of the sensor
17_FR_ACC_Experimenter_provide_feedback	Erroneous/miscalibrated sensors should be bannable.
18_FR_SEC_Testbed_authentication_mechanisms	This will be required for privacy concerns. Not directly related since the experimenter

	does not interact directly with the testbed.
19_FR_SEC_Testbed_manage_priviledges	This will be required for privacy concerns. Not directly related since the experimenter does not interact directly with the testbed.
20_FR_SEC_Experimenter_single-sign-on	This relates to the two above; indeed the experimenter should only have credentials for the FIESTA-IoT meta cloud.
21_FR_SEC_Tool_manage_users	The experimenter's account will be created in the system.
22_NFR_ACC_Distinguish_type_of_data	Experimenter would need to distinguish the different types of experimental data coming in.
23_NFR_ACC_Page_in_subrequests	Environmental data is large, and paging will help.
25_NFR_ACC_Set_response_Max_size	This is related to the paging requirement above.
26_NFR_ACC_Response_adapt_dynamics_resourc es	The experiment is agnostic to the availability of specific resources.
27_NFR_ACC_Register_new_resources	The experiment is agnostic to the availability of specific resources.
28_NFR_ACC_Response_delay_controlled	Near real-time environmental data is very useful to have.
29_NFR_ACC_Data_provided_as_requested	Experimenter requests for certain data and in turn receives requested data.
30_NFR_ACC_FIESTA_well_documented	Documentation will be used by experimenter to properly use the system.
31_NFR_ACC_Describe_IoT_services_and_applicat ions	Experimenters might be environmental scientists, and not necessarily Computer Science experts. An easy to use UI will help.
32_NFR_ACC_Provide_dev_deploy_manag_config _tools	Experimenters might be environmental scientists, and not necessarily Computer Science experts. An easy to use UI will help.
33_NFR_ACC_Scalability_data_collection	Environmental sampling leads to large amounts of data.
34_NFR_PLA_Process_non_responded_requests	Needed for detecting when underlying systems are failing.
35_NFR_PLA_Manage_resources_in_query_or_exp eriment	Indirectly required for this experiment to work.
36_NFR_PLA_Resources_produce_different_meas urements	Required to sense multiple phenomena.
38_NFR_PLA_Diff_types_requests_processed	Related to 11_FR_ACC_Request_data_different_way s above
39_NFR_PLA_Info_testbed_agnostic_way	This is an essential requirement of this experiment.
41_NFR_PLA_Minimise_processing_delay	The data should be provided with minimum

	delay.
42_NFR_PLA_Data_generated_from_processing_info	Processed information is also useful for experiment.
43_NFR_PLA_Optimise_computational_assets	Indirectly related, but would help reduce cost of experimentation.
44_NFR_PLA_Prioritization_of_services	Useful for supporting real-time queries.
45_NFR_PLA_Orchestration_of_resources	Indirectly related in order to provide data from multiple resources.
46_NFR_PLA_Homogeneous_classification	Indirectly related, in order to map between the data required by the experiments and the resources providing it.
47_NFR_PLA_Specify_compositon_data_streams	Related to 06_FR_ACC_Querying_data_compositions
49_NFR_PLA_Reliable_time_sync	Aa an experimenter can request data for any city in different timezone. It is important to have such requirement.
50_NFR_PLA_FIESTA_scalable_extensible_upgradable	The experiment should work with new urban testbeds as they are added.
51_NFR_PLA_FIESTA_highly_reliable	Will be evaluated by an experimenter during the experimentation phase.
52_NFR_PLA_Elasticity_abundance_computational_assets	Relates to 50_NFR_PLA_FIESTA_scalable_extensible_upgradable.
53_NFR_PLA_Execution_concurrent_services	Other experiments also need access to data from these sensors.
54_NFR_INT_Interop_between_fiesta_testbeds	Essential in order to support urban sensing.
55_NFR_INT_App_develop_integration_resources	Needed for supporting different types of sensors which provide sound and other environmental data.
56_NFR_INT_Support_diff_tested_data_policies	Indirectly related; will help with performance.
57_NFR_INT_Run_apps_services_interop	Indirectly related to execute urban sensing experiments.
58_NFR_MEA_Link_measurements_resources	Indirectly related due to focus on data
59_NFR_MEA_Link_measurements_metadata	Helpful for correlating different samples of environmental data.
60_NFR_MEA_Measurements_provide_metadata	Essential for performing complex queries
61_NFR_MEA_Measurements_independant_resource	Environmental sensing is about data, agnostic of resources.
62_NFR_MEA_Measurements_provide_phenomenon	Needed in order to differentiate the sampled data.
63_NFR_RES_Link_resource_tesbed	Indirectly related due to focus on data.
64_NFR_RES_Resource_provide_characteristics	Indirectly related due to focus on data.
65_NFR_RES_Resource_identified_code	Indirectly related due to focus on data.
66_NFR_RES_Calculate_reputation_resource	Indirectly related due to focus on data.

67_NFR_RES_Experiment_participants_known	Experiments will gather statistics from participating resources (i.e. mobility pattern).
68_NFR_SEC_Support_certification_authority	Required to support authentication and authorization needed for the experiment.
69_NFR_SEC_Verify_authorise_user_actions	Required to support authentication and authorization needed for the experiment.
70_NFR_SEC_Different_profile_types	Required to support authentication and authorization needed for the experiment.
71_NFR_SEC_Privacy_collected_data	Extremely important in urban sensing with people.
72_NFR_SEM_Semantic_annotations_data	Essential for ensuring that the experiment is specified once and still collects data from multiple testbeds.
73_NFR_SEM_Mapping_semantic_model	Essential for ensuring that the experiment is specified once and still collects data from multiple testbeds.

3.3.4 KPIs

- More than 1 number of testbeds involved in the experiment. The experiment will request samples from different testbeds and generate an integrated view. Thereby focusing on *testbed-agnostic access* to data.
- More than 1 users data was used in the experiment. Since ultimately we would like to involve the users directly (instead of using historical data), it will also be good to know how many individuals and from which region one would need to recruit. Thus the experimenter will also be able to limit data to certain region.
- Number of data samples needed for high-quality results. For example, an experimenter that needs to build a heatmap that consumes 30 days of data from at least 1 sensor that produces high quality samples (location accuracy below 10m) every 5 mins, then the experimenter would require at least 30*12*24 high quality samples.

3.3.5 Required Tools

As seen in previous experiments, this experiment will also need the following tool support from FIESTA-IoT:

- **Graphical user Interface (GUI) tool**: A graphical representation of the data such as noise in the city, should be shown to the experimenter for the clear understanding of the data. The GUI Tool should enable an experimenter to build for example a heatmap of noise data or representation for statistical data, allow to select a time period or other filters, and display travel map of users. This would allow experimenters to understand mobility related aspects.

- **Discovery and Filtering tool:** for matching between the experiment's requirements and testbeds in terms of the type of data provided as well as the time duration when the data was sampled.
    - This will also need to ensure that the dataset coming from each testbed matches the minimum sample-size requirements for the results to be useful for the experimenter.
    - The experimenter might want to limit the data by date or limit data that exhibits certain characteristics. For example, in case of noise, if the noise value is grater than 70dB. Thereby requiring a filtering tool.
    - Additionally, the experimenter might want the data that comes from users exhibiting a certain mobility pattern.

- **Execution and Scheduling tool:** These tools are required to extract the needed data on a one-time or regular basis from the testbeds' data stores. The Scheduling tool also allows the experimenter to schedule an experiment for longer duration of time so that real-time updates to the data are provided to the experiment.

- **Domain Specific Language (DSL) Validation tool**: to validate the DSL syntactically and structurally for the experiment and let experimenters know about the issues in DSL, it is essential to have a validation tool for DSL.

- **Documentation**: As part of the execution tool, it is essential to understand the FIESTA-IoT platform in-order to facilitate experimenters towards creating successful experiment. This should mainly provide tutorials on DSL, using which experiments can be executed on FIESTA-IoT platform.

- **Metering tool:** to understand how much data is provided in a single call to FIESTA-IoT platform and how many calls can be made by an experiment in a day. This would enable experimenter to set limits on how much they collect in a day (and therefore, cap the number of calls to FIESTA-IoT platform) for each experiment.

- **Feedback tool**: to report any anomaly or issues in the samples. The feedback tool would ensure experimenters and users to report any anomaly in the data produced by a resource.

- **Authentication, Authorization and Account Management tool**: An experimenter should be able to authenticate to the FIESTA-IoT Platform as the experiment involves data from users. The experimenter should be verified when first requesting access to the data, i.e., the experiment should clearly state the need for the data and should abide by the privacy policy and terms and conditions set by FIESTA-IoT. The experimenters should be able to request enhancements to their initial set of quota from FIESTA-IoT platform. Further, FIESTA-IoT platform should be able to notify experimenters in case there is any violation of the policy in terms of access that is set using Metering tool.

4 CONSOLIDATION

Over this chapter we present comparative tables, in order to validate the FIESTA-IoT platform by the different experiences. To make it more reading friendly, the experience *Data Assembly and Services Portability Experiment* is referred to as NEC because it will be conducted by NEC, the experience coordinated by UNICAN, *Dynamic Discovery of IoT resources for testbed agnostic data access* is referred as UNICAN and the experience leaded by INRIA *Large Scale Crowd sensing Experiments* is referred to as INRIA.

4.1 Requirements

Each of the experiments have requirements that need to be fulfilled by the FIESTA-IoT platform. Below is a table identifying each experiment and their requirements.

Table 8 - Requirements table

UNIQUE ID	NEC	UNICAN	INRIA
01_FR_ACC_Resource_independent_info	☐	☐	☐
02_FR_ACC_Resources_available	☐	☐	
03_FR_ACC_Past_Present_info	☐	☐	☐
04_FR_ACC_Discover_info_independent_testbed	☐	☐	☐
05_FR_ACC_Dynamic_discovery_resources	☐	☐	
06_FR_ACC_Querying_data_compositions	☐	☐	☐
07_FR_ACC_Discover_data_phenomenon	☐	☐	☐
08_FR_ACC_List_info_related_measurement	☐	☐	☐
09_FR_ACC_List_info_related_resource	☐	☐	☐
10_FR_ACC_Get_measurements_single_request	☐	☐	☐
11_FR_ACC_Request_data_different_ways	☐	☐	☐
12_FR_ACC_Choose_metadata_each_measurement	☐	☐	☐
13_FR_ACC_Expose_testbed_actuation_capabilities			
14_FR_ACC_Sample_specific_fractions_data	☐		☐
15_FR_ACC_Discover_resources_by_characteristics		☐	☐
16_FR_ACC_Discover_measurements_by_metadata	☐	☐	☐
17_FR_ACC_Experimenter_provide_feedback	☐	☐	☐
18_FR_SEC_Testbed_authentication_mechanisms	☐	☐	☐
19_FR_SEC_Testbed_manage_priviledges			☐
20_FR_SEC_Experimenter_single-sign-on	☐	☐	☐
21_FR_SEC_Tool_manage_users	☐	☐	☐
22_NFR_ACC_Distinguish_type_of_data	☐	☐	☐
23_NFR_ACC_Page_in_subrequests	☐	☐	☐
24_NFR_ACC_Tools_planning_auto_tasks	☐	☐	
25_NFR_ACC_Set_response_Max_size	☐	☐	☐
26_NFR_ACC_Response_adapt_dynamics_resources	☐	☐	☐
27_NFR_ACC_Register_new_resources	☐	☐	☐
28_NFR_ACC_Response_delay_controlled	☐	☐	☐
29_NFR_ACC_Data_provided_as_requested	☐	☐	☐
30_NFR_ACC_FIESTA_well_documented	☐	☐	☐
31_NFR_ACC_Describe_IoT_services_and_applications	☐	☐	☐
32_NFR_ACC_Provide_dev_deploy_manag_config_tools	☐	☐	☐
33_NFR_ACC_Scalability_data_collection	☐	☐	☐
34_NFR_PLA_Process_non_responded_requests	☐	☐	☐
35_NFR_PLA_Manage_resources_in_query_or_experiment	☐	☐	☐
36_NFR_PLA_Resources_produce_different_measurements		☐	☐
37_NFR_PLA_Info_lifecycle_management			
38_NFR_PLA_Diff_types_requests_processed	☐	☐	☐

39_NFR_PLA_Info_testbed_agnostic_way	☐	☐	☐
40_NFR_PLA_Process_feedbacks	☐	☐	
41_NFR_PLA_Minimise_processing_delay	☐	☐	☐
42_NFR_PLA_Data_generated_from_processing_info		☐	☐
43_NFR_PLA_Optimise_computational_assets	☐		☐
44_NFR_PLA_Prioritization_of_services	☐		☐
45_NFR_PLA_Orchestration_of_resources	☐	☐	☐
46_NFR_PLA_Homogeneous_classification	☐	☐	☐
47_NFR_PLA_Specify_compositon_data_streams			☐
48_NFR_PLA_Understand_not_used_info			
49_NFR_PLA_Reliable_time_sync	☐	☐	☐
50_NFR_PLA_FIESTA_scalable_extensible_upgradable	☐	☐	☐
51_NFR_PLA_FIESTA_highly_reliable	☐	☐	☐
52_NFR_PLA_Elasticity_abundance_computational_assets	☐		☐
53_NFR_PLA_Execution_concurrent_services	☐	☐	☐
54_NFR_INT_Interop_between_fiesta_testbeds		☐	☐
55_NFR_INT_App_develop_integration_resources	☐	☐	☐
56_NFR_INT_Support_diff_tested_data_policies			☐
57_NFR_INT_Run_apps_services_interop		☐	☐
58_NFR_MEA_Link_measurements_resources	☐	☐	☐
59_NFR_MEA_Link_measurements_metadata	☐	☐	☐
60_NFR_MEA_Measurements_provide_metadata	☐	☐	☐
61_NFR_MEA_Measurements_independant_resource	☐	☐	☐
62_NFR_MEA_Measurements_provide_phenomenon	☐	☐	☐
63_NFR_RES_Link_resource_tesbed		☐	☐
64_NFR_RES_Resource_provide_characteristics	☐	☐	☐
65_NFR_RES_Resource_identified_code	☐	☐	☐
66_NFR_RES_Calculate_reputation_resource	☐	☐	☐
67_NFR_RES_Experiment_participants_known		☐	
68_NFR_SEC_Support_certification_authority	☐	☐	☐
69_NFR_SEC_Verify_authorise_user_actions		☐	☐
70_NFR_SEC_Different_profile_types	☐	☐	☐
71_NFR_SEC_Privacy_collected_data	☐	☐	☐
72_NFR_SEM_Semantic_annotations_data	☐		☐
73_NFR_SEM_Mapping_semantic_model			☐

4.2 Datasets

Experiments need some datasets on their samples to accomplish the proposed objectives. In the next table there is a resume of datasets required (or optional). From the deliverable D2.2, Table 8 we can make a cross-reference from what is available, or is possible to get, from the testbeds into FIESTA-IoT, and what the experiments will need for the datasets. In green, are the features that can be delivered by more than one of the initial testbeds, in orange are marked the features only available from one testbed, and marked blue, are the items that are not available at the moment (explicitly or difficult access).

Table 9 - Resume of datasets

Datasets	NEC	UNICAN	INRIA
Location (GPS / 3G / Wi-Fi)	☐	☐	☐
Time zone (local)		☐	
Timestamp (local)	☐	☐	☐
Environmental Pollution	☐	☐	

Presence control		☐	
Traffic monitoring		☐	
Parking guidance		☐	
Tags		☐	☐
Detector (motion / contact)		☐	
Power consumption		☐	
Crowd sensing		☐	☐
Mobile Sensing (buses, taxis, etc.)		☐	
FIESTA-IoT Unique resource ID	☐	☐	
Local Unique resource ID		☐	☐
Unit of measurement	☐	☐	
Resource description / details	☐	☐	☐
Sensor surrounding noise			☐
Proximity Sensor (phone)			☐
Call state (phone)			☐
Activity reported (phone)			☐
User annotation (phone)			☐

4.3 Experiment KPI's

Experiments KPI's are in their foundation very similar because all experimenters want to take advantage of the exceptional features that FIESTA-IoT intends to offer. In resume, all experiments take advantage of only needing to connect to FIESTA-IoT exposed interfaces. In this context, the federated testbeds, can give a measurement in a single request and be transparent to the user, being *testbed-agnostic* and also scalable in the number of testbeds. In some cases, it is necessary to know in advance the set of resources, in order to reduce the data and also to guarantee that the resources are available through the length of all experiment. On the other hand, some cases need to know when, in run time, the resource is available, and also the cadence and reliability of the data. The data, can be gathered in real time or from a database, with the metadata associated from each resource. FIESTA-IoT should accomplish EaaS (Experiment-as-a-Service) giving response to small or large queries and statistical data over one or across testbeds. Also, very important, is to be able to define and use Virtual Entities (VE). Last, but not least, FIESTA-IoT must provide a graphical user interface (GUI) to enables all kind of users, with or without experience, to add their own experience to be executed in FIESTA-IoT.

Table 10 - Resume of KPI by experience

KPI	NEC	UNICAN	INRIA
Get data from different resources/testbeds on a single request	☐	☐	☐
List of valid resources during the experiment	☐	☐	☐
Continuous Discovery	☐	☐	
Available datasets / historical datasets	☐		

Composite data / Virtual Entities	☐	☐	
Phenomena Discovery	☐		
Quality of data	☐		☐
Choose available metadata	☐		
Experimenters feedback	☐		☐
Authentication of experimenters	☐	☐	
Run personalised scripts		☐	
Invisibility of testbed sensors		☐	
Cross-testbed response		☐	☐
Historical Data		☐	
Natural language queries		☐	
Number of “active user sensor”			☐

4.4 Features

In order to accomplish the experiments, FIESTA-IoT must provide the correct tools. Some of them are similar for all experiments, but some experiments need specific tools that are mentioned in the next table.

Table 11 - Resume of required tools

Tool	NEC	UNICAN	INRIA
Assync query	☐		
Historic query	☐	☐	
Data by parameter	☐	☐	☐
One time / Run time resource discovery	☐	☐	☐
Authentication	☐	☐	☐
Reservation of resources	☐	☐	
Data quality	☐	☐	
Scripting / Advance query tool / API access	☐	☐	☐
Phenomena description	☐	☐	
Natural language queries		☐	
Documentation		☐	☐
GUI	☐	☐	☐
Datasets		☐	
Virtual Entities creation		☐	
Statistic representation		☐	
Feedback		☐	☐
Scheduling			☐
Domain Specific Language Validation			☐
Data metric			☐

5 REFERENCES

Gavras, A. (2010). *Experimentally driven research white paper .* ICT-FIREWORKS .

Haren, V. (2009). *TOGAF Verson 9.0.*

Hoornweg, D., & Blaha, D. (2006). *The Current Status of City Indicators.* The World Bank.

IEEE. (2007). *Guide for Monitoring, Information Exchange, and Control of Distributed Resources Interconnected with Electric Power Systems.* IEEE.

IEEE. (1990). *IEEE standard glossary of software engineering terminology.* IEEE.

IoT-A. (2011, June 16). Retrieved May 29, 2015, from Project Deliverable D1.2 – Initial Architectural Reference Model for IoT: http://www.iot-a.eu/public/public-documents/d1.2/view

IoT-A. (2013, July 15). Retrieved May 29, 2015, from Deliverable D1.5 – Final architectural reference model for the IoT v3.0: http://www.iot-a.eu/public/public-documents/d1.5/view

MyFIRE. (2011, May). Retrieved Jully 6, 2015, from D1.2 Taxonomy on common interpretation of testing, testing approaches and test beds models: http://www.my-fire.eu/documents/11433/38630/D1.2+-+taxonomy+on+common+interpretation+of+testing%2c%20testing+approaches+and+test+bed+models?version=1.0

Project Management Institute. (2013). *A Guide to the Project Management Body of Knowledge* (5th Edition ed.). USA.

Soukhanov, A. H., Ellis, K., & Severynse, M. (1992). *The american heritage dictionary of the english language.* Boston, MA: Houghton Mifflin.

