Universidad de Alcalá
Escuela Politécnica Superior

Ingeniería Informática

Trabajo Fin de Grado

FIWARE-based application for control of Smart Cities

Autor: Alejandro Fernández Maceira

Tutor/es: Ana Castillo Martínez

2020

UNIVERSIDAD DE ALCALÁ
Escuela Politécnica Superior

Ingeniería Informática

Trabajo Fin de Grado
FIWARE-based application for control of Smart Cities

Autor: Alejandro Fernández Maceira

Tutor/es: Ana Castillo Martínez

TRIBUNAL:

Presidente: << Nombre y Apellidos >>

Vocal 1º: << Nombre y Apellidos >>

Vocal 2º: << Nombre y Apellidos >>

FECHA: << Fecha de depósito >>

Agradecimientos

Gracias a mi familia por apoyarme y aconsejarme en todo momento, a mis amigos por estar ahí y a mi tutora por la ayuda y los consejos que me ha dado.

Índex

Índice de imágenes

Índice de tablas

Resumen

Esta tesis es un estudio teórico sobre el framework FIWARE, su ecosistema y sus aplicaciones prácticas. Primero se hace una descripción de FIWARE como ecosistema, los principios en los que está basado, así como su misión y un histórico de su implementación. Después se detallan los programas que forman el ecosistema y su comunidad. En la parte técnica, se describe, con el uso de ejemplos, la tecnología que utilizan los distintos componentes que forman FIWARE y el mercado en el que adquirir las soluciones. Por último se muestran algunos casos de éxito de la implementación de FIWARE.

Palabras Clave

FIWARE, Smart Cities, Internet of Things, Framework, Open Source.

Abstract

This thesis is a theoretical study about the FIWARE framework, its ecosystem and its practical applications. First, a description of FIWARE as an ecosystem and the principles it is based on, as well as its mission and a timeline of its implementation is done. Then, the main programs and the community that form the ecosystem are detailed. On the technical section, it is described, with the use of examples, the technology employed in each FIWARE component and the market where the solutions can be acquired. Finally, some success stories are shown where FIWARE was implemented.

Key words

FIWARE, Smart Cities, Internet of Things, Framework, Open Source.

1. Introduction

Nowadays, technology is used in almost every aspect of our lives. Being able to control the different technological devices and use them to improve our lives is now the top priority. Combining the growth cities are experiencing with technology, and more specifically, Internet of Things, life on cities can be managed more easily while saving resources in the process. That is the reason why there has been a sudden rise in platforms and development of frameworks, to deal with this newfound field of study.

Internet of Things devices and sensors must be set up in a network where the information is shared among them, and then transmitted to a central server to operate with that information to obtain results that allow to solve several problems the cities have, for example lighting or waste management. The control of this system is best done with a centralized framework and a solution that joins all the different systems in one place. But cities have a limited budget, and normally they cannot invest public money in the development of smart solutions, even though it may help in the long run. However, there is the problem of installing multiple solutions from different sources, because they most probably will not work well together, so the management gets complicated and it produces a sense of rejection for the whole Smart City topic, rejecting the installation of a system in the city. If the city wants to install a framework for the management of smart solutions, it turns to private investors and stakeholders for advice, but they will have many options to choose from.

Among the various options of frameworks available in the market, there is one that shines above the rest and that is FIWARE. Not only is it a framework for the development of Smart applications, but also a full ecosystem with different programs and features that differentiate it from its competitors. One of the characteristics that make this framework interesting for the development of applications for the control of Smart Cities is the open nature of the project, as everything related to the platform is and will be free and open for everyone, from the standards and technology used to program the solutions to the datasets obtained from the sensors installed in cities. In addition, as it is open, it can work in cooperation with other solutions that are already in place or will be installed in the future, and because it is all centralized, there will be no need to worry about the management of different applications, it can all be done from one place.

2. Objective

The main objective of this thesis consists in establishing FIWARE as a clear tool to develop Smart systems that can solve most of the problems that todays' cities suffer. A deep analysis of the technologies and standards used by FIWARE will be performed in order to have a solid foundation on what to expect from a framework dedicated to the Internet of Things. Moreover, the whole FIWARE ecosystem will be detailed, from the different programs that it is composed of to the origin and mission that it wants to achieve. Finally, an objective is to show that FIWARE has already been employed in real-life scenarios and it has been implemented in a successful way by giving examples of such implementations.

3. State of the Art

The current technologies available allow to develop a robust Smart City architecture for any city and their specific circumstances. Next, the main components of a Smart City will be presented and detailed, focusing on the current technologies and denoting the best choices for a Smart City project.

3.1. Internet of Things

Internet of Things (IoT) is a term that doesn't have a clear definition, it depends on the point of view. The concept of IoT that better applies to the topics presented in this thesis can be described as follows: The connection of different devices that capture information from the real world, share that information between them, and process it in order to provide services to everyone, directly or indirectly [1].

Nowadays, technology allows developers and researchers to improve IoT-based systems and implement new functionalities that were not available in the beginning of this technology.

The most recent IoT projects in relation to Smart Cities are focused on helping in the management of climate change related problems that many countries suffer, developed and underdeveloped alike. Some of these projects revolve around creating efficient water-delivery systems for agriculture [2] or achieving the Sustainable Development Goals of United Nations [3].

In order to create a consistent network of IoT devices that can exchange the information they receive from the environment and send the data to the servers, a connection between all the different devices must be made. Depending on the distance of the connection among devices, there can be two categories of protocols: short range and long range [4].

Long range protocols, also known as Low Power Wide Area Network (LPWAN), allow connections up to 50 kilometers radius. The best example is cellular connection, such as GSM (2G), HSPA (3G), LTE (4G) and the emergent 5G. These protocols can be found in mobile phones to get internet connection but are rarely used in small IoT networks due to the high-power consumption of the communication, even though they can transfer a huge amount of data in a short time. In terms of security, this protocol uses RC4 cipher, which has been proved unsecure, but is very fast.

Short range protocols are known for their low-power consumption and limited bandwidth. Some protocols that have recently seen a spike in usage are NFC and Z-Wave, but BLE and 6LoWPAN also belong in this classification. The latter examples are older but are still used to this day due to their easy implementation and low energy consumption. Bluetooth guarantee fast and easy connections and was very used in mobile phones to transfer files but has been outdated because long range mobile protocols and instant messaging applications are now more accessible. In the case of 6LoWPAN, IPV6 over low power networks, the use of IP packets takes full advantage of

the IP standard and the years of experience with the protocol, and it is a widely acclaimed IoT protocol for communication in a Smart City environment.

NFC and Zigbee are more recent, and they continue to be installed in new systems and projects. Near Field Communication requires a very small distance between objects to communicate, making it especially useful in contactless payment and smartphone communication with other mobile phones or data in the form of small chips. Zigbee allows for more distance than NFC but the key aspect of this protocol is that it supports different types of network topologies like mesh, star and tree. Moreover, it has an AES-128 encryption support, which contrary to some other long-range protocols, has been proven secure. Its effective range is up to 300 meters within line of sight in the outside, and 100 meters inside buildings or homes [5]. All these characteristics make Zigbee an outstanding protocol for IoT in Smart Cities, with support for systems whether they are installed inside buildings or outside, ensuring the security of the connection and support for multiple network topologies.

Lastly, there are some other IoT protocols that are also of importance as they will be mentioned later in the thesis. The first technology is SigFox, a low power technology for wireless communication of a diverse range of low energy objects such as sensors and M2M applications. It allows the transportation of small amounts of data ranging up to 50 kilometers. SigFox uses Ultra Narrow Band (UNB) technology, designed to handle low data transfer speeds from 10 to 1000 bits per second, and can run on a small battery. SigFox support start network topology. Other protocol is LoRaWAN, a media access control (MAC) protocol for wide area networks. It is designed to allow low-powered devices to communicate with Internet-connected applications over long-range wireless connections. LoRaWAN can be mapped to the second and third layer of the OSI model [6]. The next protocol is MQTT (Message Query Telemetry Transport), a publish/subscribe, simple and lightweight messaging protocol, designed for constrained devices and low-bandwidth, high-latency or unreliable networks. The design principles are to minimize network bandwidth and device resource requirements while also attempting to ensure reliability and assurance of delivery [7].

3.2. Smart Cities

As with IoT, the term Smart City will vary depending on the point of study. The definition that most suits this thesis is as follows: A Smart City is a city that uses all available technology and resources in an intelligent and coordinated manner to develop urban centers that are at once integrated, habitable and sustainable [8]. This definition is related to the previous definition of Internet of Things, as the means of communication between systems and the objectives of such systems are made possible thanks to the IoT.

Some of the assets that IoT manages are transport, lighting, power supply, waste disposal, water supply, shopping centers, public safety, and many more that are directly (transport) or indirectly (power supply) related to the citizens living in the city. According to the European Innovation

Partnership on Smart Cities and Communities (EIP-SCC)[1], Smart Cities have six main fields of action or action clusters where resources should be focused on, which are citizen focus; business model and finance; integrated infrastructure and processes, integrated planning, policy and regulations; sustainable districts and sustainable urban mobility [9].

Nowadays, there are some cities that can be considered "Smart", and each one has a certain level of development in a certain area, such as transportation, energy management, recycling efficiency and many more. These areas are not mutually exclusive, as one city can have more than one area implemented with IoT, and that does not mean that other areas are less developed, even though they can be. The city of Santander, in Spain, is an example of a Smart City, as it has a network composed of more than 15,400 devices collecting information and solving issues like traffic management, waste treatment or street lighting, as well as an application available to the citizens to report problems and make suggestions to the government directly [10]. All the information can be accessed freely on the FIWARE Open Data webpage, and anyone can read the data collected by the IoT devices.

3.2.1. Maturity Model

In order to have a way to measure the level of development a city has regarding technology and IoT, some organizations such as the International Data Corporation (IDC) have created a self-assessment tool that helps in the identification of systems that are well developed and those that need improvement. In addition to the tool, there is also a model, called maturity model, that classifies cities in different levels according to certain metrics. The IDC was the first entity that published a full document detailing a scale in which a city is located [11]. These different levels in the scale are five, and go from one to five in increasing order, level one being the lowest development, and level five being the biggest. These levels are as follows:

- Level 1 or Ad Hoc. This stage represents the general way city government works when there are no concrete smart projects. Cities that belong in this level react to problems as they arise with little to no IoT involvement.

- Level 2 or Opportunistic. Key concept is system collaboration. In this stage, opportunistic projects begin its deployment thanks to the incoming data. Stakeholders start to develop a common strategy, language and cross boundaries with other partners.

- Level 3 or Repeatable. Key concept is system integration. In this stage, projects, events and processes that are recurring are identified for further integration. The city becomes outcome driven and led by strategy, with system-wide technology. Strategies are properly defined and documented. Finding new sustainable funding models and governance issues become a focus.

[1] The European Innovation Partnership on Smart Cities and Communities (EIP-SCC) is a major market-changing undertaking supported by the European Commission bringing together cities, industries, SMEs, investors, researchers and other smart city actors [12].

- Level 4 or Managed. Key concept is managed system. Response systems and technology assets are deployed, while standards begin to emerge. Improved prediction and real-time response deliver better outcomes. Focus shifts into budget, IT investment and governance structure, affecting all parts of the city.

- Level 5 or Optimized. Key concept is sustainability. Now, a city-wide platform is fully functional, new projects and management of current ones are developed using agile methodologies. Focus is on constantly improving systems within the bigger system and compete with other cities.

Most cities nowadays are currently in the first level of the maturity model, as there are no Smart devices and systems installed. There are a handful of cities in the opportunistic and repeatable level, but most of them are focused on researching new projects and evaluating their capabilities for smart projects.

This model, according to IDC, is best used as a guide to establish goals and a common vision, language and roadmap that can clear the way for a future development of a proper smart city.

3.2.2. Architecture

As previously mentioned, IoT devices are essential for any implementation of Smart City projects. The data they capture and the following evaluation into statistics and reports are invaluable for a correct development of the smart projects in cities so they can make their way into the higher levels of the maturity model.

The first thing to do is plan ahead how the implementation of the devices will be. To do so, it is best to create a draft of an architecture to follow in order to not lose the scope of the project and quickly identify the systems that take part in any given scenario. There are many different IoT architectures, as everyone should be adapted to the circumstances it is developed, because each city is different in various factors, such as culture, density of population, social foundations, and so on. Nonetheless, some authors have developed a working, generic model to follow that contains the necessary guidelines to create a cohesive and strong architecture to start from, and then work upon it, improving or expanding its components, until it encompasses all the systems that are, or will be implemented. The following picture represents a generic IoT architecture [13].

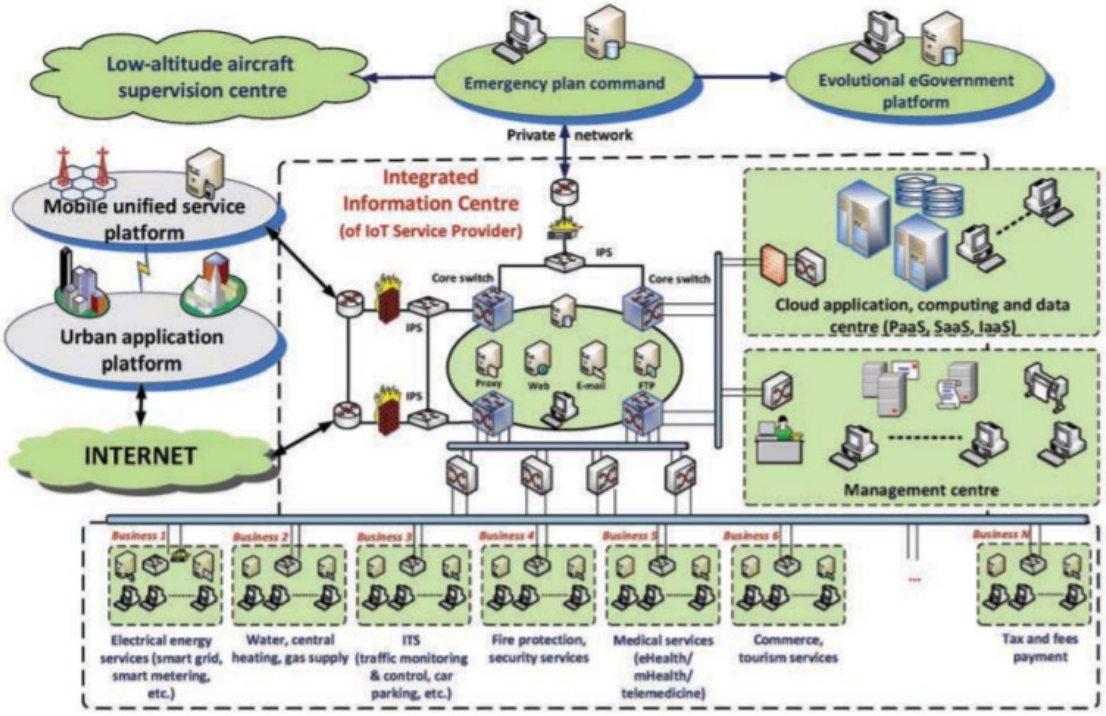

Figure 1. Generic IoT architecture for Smart Cities

3.2.3. Programming context

Following the creation of the architecture, in order to develop new IoT systems that can be used in cities to gather data and create projects that help citizens in their everyday life, and government to make decisions according to such data, the IoT devices and their behavior must be programmed in a certain context. The way sensors pick up the data from the real world, then transmit the digital values to a server to interpret them and generate reports that can be understood by humans involve many different programs and software, so it can get a little complicated to manage the architecture implemented in a big city. For every part of the architecture, there are multiple options that can be implemented, but choosing one of them will depend on many different factors, for example, the budget, the software functionalities, or code management.

Following the previous diagram (Figure 1), there needs to be a few different programming languages and software applied in distinct areas of the architecture. For example, in order to manage the database with all the data from the IoT sensors, SQL and relational databases are the top choices for a relational database, but other options such as MongoDB, which is non-relational, are a good choice too [14]. From the sensors side, some programming languages like C, Java and Python are the most chosen options, mainly because they are very versatile in the case of Python and Java, and very reliable and low-level oriented in the case of C.

However, managing all different areas of the architecture can be a hard task. That is the reason why new frameworks are appearing lately, as they encompass all the areas into one centralized environment.

4. FIWARE

FIWARE is an open source framework of curated APIs especially designed for smart solutions and accelerate its development [15]. It is based on openness, which distinguish FIWARE from other similar platforms such as SAP [16] or TCS' Intelligent Urban Exchange (IUX) [17] on the field of Smart Cities. The basics of the open platform are the following.

- Open source. Every component in the framework is open source, which means that anyone can access its code and features without monetary cost, have modification permissions for such code, and publish the results freely, contributing to the community in the process.

- Open community. The FIWARE community is the cornerstone of the whole project and its goal is to accomplish the FIWARE mission. This community has free admission to organizations and individuals while providing full access to the collected data from different cities and corporations to develop new solutions.

- Open data. The datasets provided by the members of the FIWARE community are open to the general public, everyone can access the data and check the information without requiring a FIWARE account.

Given all these characteristics, FIWARE can also be interpreted as an ecosystem where innovation, community and software tools come together to provide smart solutions that can apply to many different fields of knowledge.

4.1. FIWARE origin and mission

FIWARE was born as a response for private software solutions aimed at Smart Cities and their lack of replication and generalization. It was the result of an agreement between the European commission and four enterprises: Orange, Telefónica, Atos and Engineering; making FIWARE a public-private collaboration [18]. It is especially useful in the field of smart cities because it ensures interoperability and the standardization of data models.

Initially, the project was called FI-PPP, Future Internet Public Private Partnership, but it would then become the different programs of FIWARE [19]. It was a European program aimed at accelerating the adoption of Future Internet technologies, advancing the European market for smart infrastructures following an industry-driven approach, but also user-oriented.

4.1.1. FIWARE deployment phases

The FIWARE project was built in phases, each one with an action plan regarding the objectives to reach and the means to achieve them.

The first phase began in May 2011, with an initial investment of ninety million euros, and lasted until March 2013. It was aimed at defining the technological foundation and develop the first

eight use cases in industry sectors like transport, smart cities and agriculture. The foundation was called FIWARE, and its goal was to facilitate access to services, cloud hosting, IoT devices, data and context management. In this phase there were tested many infrastructures, making an inventory of the ones that were public and available in the INFINITY repository, and the next architectures were planned.

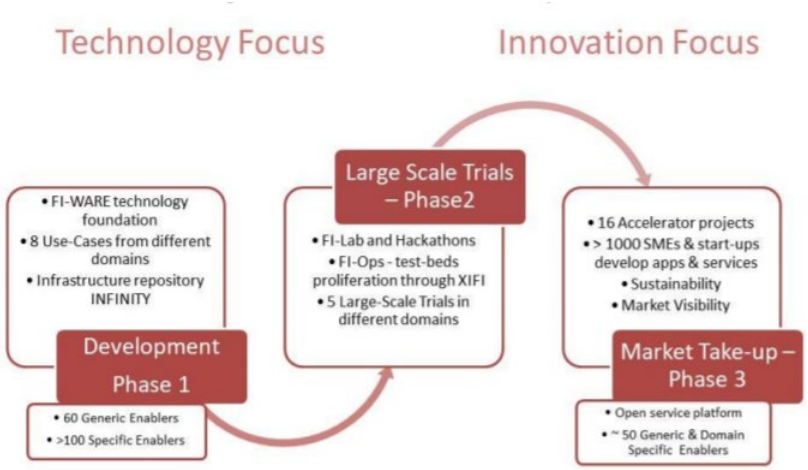

Figure 2. Phases of the FI-PPP project implementation

The second phase began in April 2013 and ended in September 2015[2] with an investment of eighty million euros. In this phase, the main objective was to develop the core platform, FIWARE, while implementing its nodes. Also, the availability of the infrastructure was ensured to perform some Large-Scale Trials, that would lately become the different programs of FIWARE. Additionally, those trials began working, some of them based on the first phase use cases such as the related to smart cities, while others were new such as the health ones [20]. Notably, in this phase, FIWARE Lab, originally FI-Lab, was born as a sandbox environment for developers to test their solutions.

[2] The second phase should have ended in March 2015 but was extended six months more to give better support for the accelerators in the next phase.

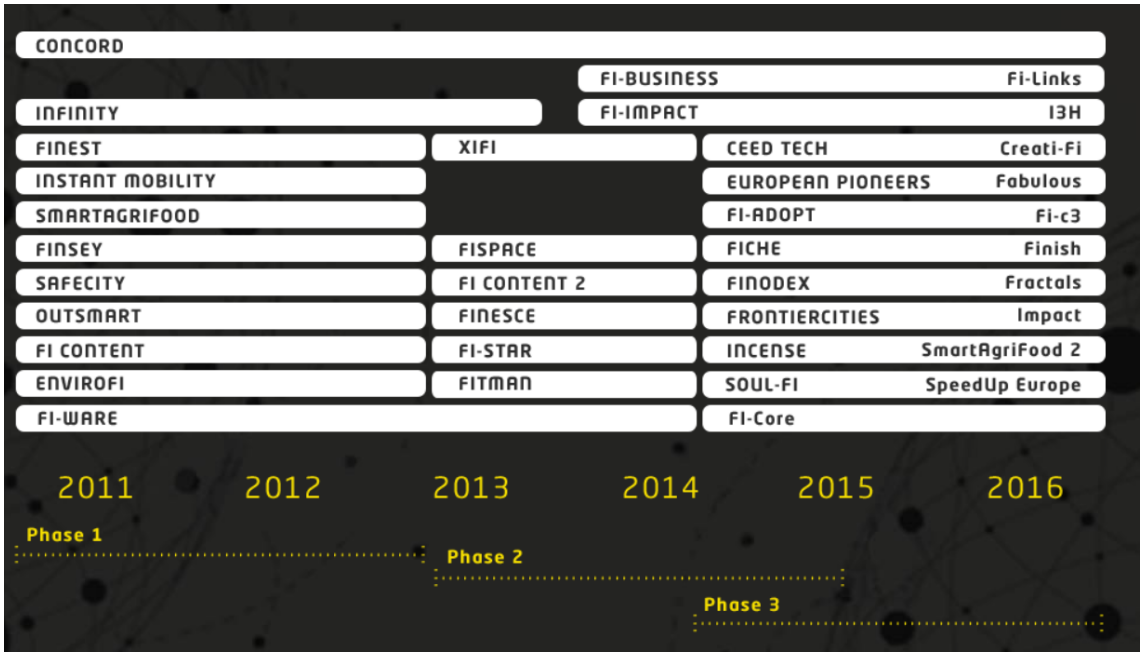

Figure 3. FIWARE projects during the different phases with timeline.

The third phase began in April 2014 and ended in December 2016, with an investment of one hundred million euros. In this phase, the efforts were put into attracting entrepreneurs, startups and SMEs into the platform, while at the same time improving upon the infrastructure for the Large-Scale Trials. The most important milestone of this phase was the selection of the first sixteen accelerators through the FIWARE Accelerator program, as thanks to their work the FI-PPP program was a success, promoting the usage of the FIWARE technology. Additionally, four new program support actions were launched: Fi-Business, sharing best practices and building a community; Fi-Links, elaborating a roadmap and help in international relations, would later become FIWARE Mundus; Fi-Impact, measuring the impact the accelerators were having; and I3H, creating a network of digital innovation hubs, would later become FIWARE iHubs. Finally, the Generic Enablers were further developed, and efforts were put for them to be supported by the market.

4.1.2. Projects involved in the deployment of FIWARE

While the first and second phase were developing, the INFINITY (INfrastructure support and capacity building for the Future INternet communITY) project was crucial to the success of those phases. This project was aimed at communicating with organizations across Europe to build a repository with the available information about infrastructures and interoperability requirements, documenting operational constraints and identifying opportunities that could facilitate large-scale testing [21]. Given the goal, it is clear that the first two phases were greatly aided by this project, as the Large-Scale Trials were possible thanks to the information contained in the repository.

Other parallel actor very important during the whole 3 phases was CONCORD, the Facilitation and Support action for the EU-funded Future Internet Public-Private Partnerships (FI-PPP) program [22]. The support this project gave to the stakeholders of FI-PPP was all about the adoption of fair, neutral, open and inclusive procedures, creating a transparent decision-making

collaborative environment among all projects, where the common problems and issues could be resolved, giving a clear and centralized platform to coordinate the whole FI-PPP project.

Last but not least, a small but important project was XIFI, Experimental Infrastructures for the Future Internet, responsible for the capacity building part of the FI-PPP program. It created a sustainable network of Future Internet infrastructure, establishing a European market for the Large-Scale Trials. The actions taken to ensure the achievement of the mission were the integration of the existing components, identified by the INFINITY project, with the open Generic Enablers, checking if the interoperability requirements were satisfied. About the parties involved, the XIFI project was carried out by different entities, from universities and research centers to major telecom operators and SMEs, all with significant experience in the ICT field [23].

4.1.3. FIWARE mission

All of the different programs that FIWARE is composed of follow the philosophy dictated by the mission, that is: *"to build an open sustainable ecosystem around public, royalty-free and implementation-driven software platform standards that will ease the development of new Smart Applications in multiple sectors"*. Furthermore, FIWARE follows a globalization strategy, represented clearly in FIWARE Mundus, where the objective is to expand the project beyond Europe. Nowadays, other countries such as Mexico, Brazil or India also have FIWARE-developed solutions, but the majority of nodes are still in Europe.

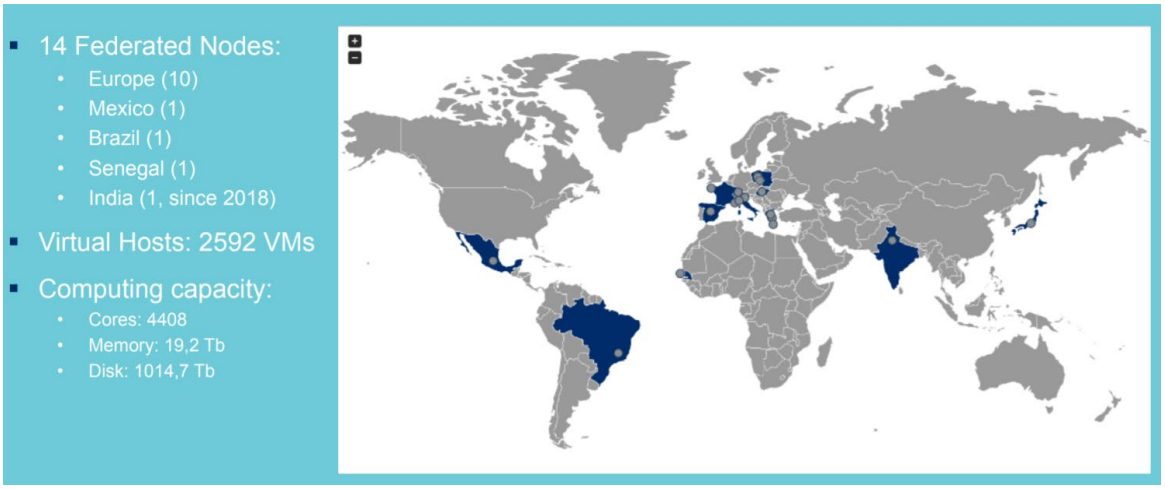

Figure 4. FIWARE expansion across the globe. [24]

4.2. FIWARE Foundation

The first organ in the FIWARE ecosystem is the Foundation. The FIWARE Foundation is, in their own words, "the legal independent body providing shared resources to help achieve the FIWARE mission by promoting, augmenting, protecting, and validating the FIWARE technologies as well as the activities of the FIWARE community, empowering its members including end users, developers and rest of stakeholders in the entire ecosystem" [25]. This means that the Foundation takes care of the relationships between every member in the platform, as well as providing the

guidelines to follow and the means to accomplish them. It is the main organ in the FIWARE ecosystem and is responsible for all the decisions taken that involve the course of action related to the expansion of the FIWARE technology, such as the iHubs or the Mundus programs. In addition, the Foundation is a non-profit organization, as its only goal is to spread the knowledge of FIWARE solutions and their open source standards and data interoperability.

4.2.1. Goals and origin

The Foundation is based on five clear goals:

- Empower. The developers with clear tools that create clean code and ease the validation process. The users with great software to access to and providing easy-to-use feedback tools. The FIWARE members by organizing events and activities.

- Promote. The new technologies coming with FIWARE, the developers who produce such technologies, the users who create the offering for those technologies and the whole FIWARE ecosystem as creator of economic opportunities.

- Augment. Increasing interfaces for connectivity, promoting the usage of new standards and developing use cases for new applications.

- Protect. The FIWARE trademark, the compliance with the code of conduct, the technologies that compose the platform and the openness and transparency following the FIWARE mission.

- Validate. By having a quality assurance for the FIWARE technologies and organizations that can verify and label the "Powered by FIWARE" solutions.

The Foundation was created when the FIWARE project started, back in 2016 with the first members Orange, Telefónica, Atos and Engineering. Since then, the members have grown until reaching more than 345 across the globe, from big enterprises to individuals. As expected from FIWARE, the Foundation is open to anyone: an individual can join, as well as a city council that wants to contribute to the project. Even though it is open, it does not mean it is free. An individual member can sign up for free, but when a startup or some bigger organization wants to join, a membership fee is applied. Following this philosophy, the influence and power in decision-making will be limited to the basic members but improved to the ones who contribute the most.

4.2.2. Internal structure

Moving on to the internal structure of the Foundation, it is composed of the following bodies: Board of Directors (BoD), Board of Officers (BoO), Technical Steering Committee (TSC) and the General Assembly.

The first body is the Board of Directors, and its main function is controlling all the business and actions of the FIWARE Foundation. Some other functions are the determination of the General

Assembly and BoD meetings, the enforcement of the code of conduct and bylaws and the management of financial resources. The members in this body are one representative for each Platinum member, representatives for Gold members but not more than Platinum, the CEO and the chair of TSC, both of those without voting capabilities.

Next is the Board of Officers, which takes care of the daily tasks of the FIWARE Foundation. Its functions are the elaboration of the annual report and keep the financial balance up to date. The only members of this body are the CEO, a treasurer and a secretary.

The following body is the Technical Steering Committee, and it is in charge of all the technological decisions affecting the FIWARE platform. The decisions taken by this body regarding additions, deletions or modifications of FIWARE technology will be considered by the BoD, even though they will have the last word. The TSC is composed of one member from each Platinum member, two per FIWARE Chapter (Architecture and Mission) and some selected Individual members whose actions and support of the technology is remarkable.

The final body is the General Assembly, which is an annual assembly of all the members of the Foundation, those are, the represented in each member tier. Among the responsibilities of the GA are the approval of the budget for the following year, supervising the work of the BoD and BoO, and any decision to make proposed by the Board of Directors.

4.2.3. Membership tiers

Next, depending on the degree of support and involvement within the FIWARE ecosystem, the members will be placed into different tiers.

- The highest tier is the Platinum member tier and is reserved to the most active and supportive members of the community. The fees and requirements are higher as well, in relation to the position these members hold. This tier also grants access to have a representative seat in the Board of Directors, a vote in the General Assembly of the Foundation and have a representative in the Technical Steering Committee. However, the Platinum membership can be lost if the member changes ownership or the field of activity it was previously working on [26]. The membership fee for a Platinum member is a hundred thousand euros annually, having a two-year commitment. Some members present in this tier are the four original founders and others such as NEC or Red Hat.

- The next tier is the gold tier and is composed of enterprises and organizations committed to FIWARE and its mission, taking an active role in its diffusion and expansion. This tier grants access to elect a Gold member representative in the Board of Directors and a vote in the General Assembly of the Foundation. The membership fee for a Gold member depends if it is an organization or a startup. In the former, the fee varies from two thousand five hundred euros to fifty thousand euros annually based on the capital, while in the latter the fee is a thousand euros annually. Some members that belong in the Gold tier are Everis, Zabala or Addix.

- The following tier is the Associate tier and is comprised of associations and legal entities that are non-profit and have an interest in expanding the FIWARE mission. This tier grants a vote in the General Assembly of the Foundation, but there cannot be members from this tier in the Board of Directors. The membership fee for an Associate is just a thousand euros annually. Some members in this tier are universities, such as Universidad Politécnica de Madrid or Università di Genova, but there are some organizations too, such as International Data Spaces or Irish Manufacturing Research.

- The last tier is the Individual tier and consists of any natural person who promotes the FIWARE mission. These members can vote in the General Assembly of the Foundation and have a seat in the Technical Steering Committee. Only the individuals working for a Platinum or Gold member can be part of the Board of Directors. There is no membership fee for an Individual member.

MEMBERSHIP LEVEL	ANNUAL FEE	FTES	BOD SEAT	TSC SEAT	NOTES
PLATINUM	FLAT FEE 100K€	2	YES	YES	2Y INITIAL COMMITMENT, ANNUAL PAYMENT
PLATINUM – STRATEGIC END-USER	FLAT FEE 100 K€	1	YES	YES	2Y INITIAL COMMITMENT, ANNUAL PAYMENT
GOLD	2.5-50K€ BASED ON ORGANISATION	0	BY ELECTION	NO	0.025% OF YEARLY REVENUE
GOLD – STRATEGIC END-USER	1.25-25K€ BASED ON ORGANISATION	0	BY ELECTION	NO	0.0125% OF YEARLY REVENUE [1]
GOLD – STARTUP	1K€	0	NO	NO	[2]
ASSOCIATE	1 K€	0	NO	NO	
INDIVIDUAL	0	0	NO	BY ELECTION	

Table 1 Table with the tiers, fees and representation rights

Simultaneously, there are some special members known as Strategic End Users, which can be part of the Platinum and Gold tier. These members do not belong in the Information Communications Technology (ICT) field but are users of it and are mostly city councils implementing some FIWARE-related solution while helping in the spread of the FIWARE mission, such as the Santander, Genova or Montevideo city councils. Even though these members can be in the Platinum tier, actually there are none, all of them belong in the Gold tier. As active

users of ICT, these members have a reduction in the membership fee, going from a thousand and twenty-five hundred euros to twenty-five thousand euros annually.

4.2.4. Access requirements

Accessing the Foundation requires following a procedure whatever the tier the member is applying to. The basic requirements are: provide the required documentation to the Secretary and then sign an adherence agreement, the code of conduct and the fee code of the Foundation. Should the application be approved by the Board of Directors and the documentation verified, the CEO or some member of the BoD in their behalf will admit the applicant as member in the selected tier. Only the BoD can decide whether to incorporate or not a new member into the Platinum, Gold or Associate tier.

Figure 5. Some of the FIWARE members and their respective tiers

4.3. FIWARE Programs

FIWARE is formed by different programs, each one covering a specific field of knowledge. These programs are FIWARE Lab, which is a non-commercial environment where testing based on FIWARE technologies occur; FIWARE Mundus, which is an initiative to promote the use of FIWARE all around the world; FIWARE iHubs, which are centers of FIWARE located in cities where developers come together to create smart solutions; and FIWARE Accelerate, which aims at promoting the usage of FIWARE-related solutions among start-ups and small and medium enterprises.

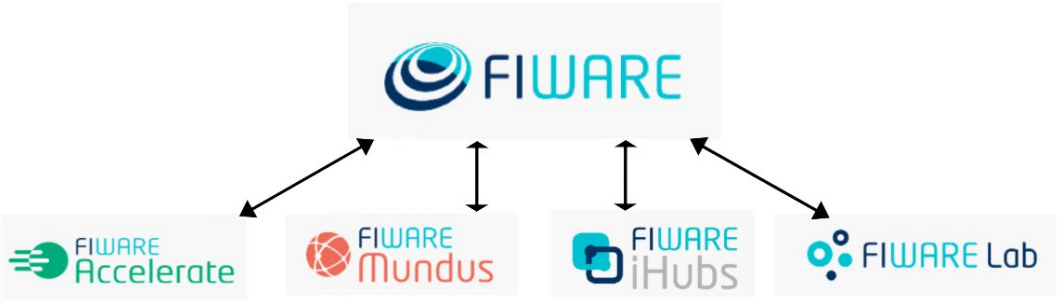

Figure 6. FIWARE Environment

4.3.1. FIWARE Community

The FIWARE community is an independent open community whose members are focused on accomplishing the FIWARE mission [27]. The community is formed by every participant in the FIWARE ecosystem, whether they are active in the Foundation, contributing creating new smart solutions for a city or simply developing relevant resources in FIWARE Lab, all of them are part of the community, as well as the members from the different FIWARE programs such as Mundus, iHubs or Accelerate located in countries around the world.

Figure 7. FIWARE Community logo

The community is based on independence in decision making, openness, transparency and meritocracy. The work supervision in the community is spread across various bodies responsible for the technical questions that may arise during the contribution process. The Technical Committee and the FIWARE Chapter deal with those questions, while the FIWARE Ecosystem Support takes care of the non-technical inquiries from the different FIWARE programs.

As stated before, the access to the community is open to anyone, individuals or organizations. In order to access some features and contribute to the programs, a join up is required, and it works the same way as joining the Foundation. Nonetheless, there is no need to sign up to be able to contribute, the FIWARE Lab and the developer tools are free of access and creating resources already counts as a contribution as long as it follows the FIWARE mission and the code of conduct.

Continuing with the various options the community offers, the evangelist program was created to spread the FIWARE mission and platform to every part of the world. The evangelists have a few tasks, namely help the reach of the FIWARE ecosystem, write about event experiences and work closely with the iHubs. However, being an evangelist has its advantages, such as attending the

FIWARE events and speak at the global summit, represent the platform in public and online and get an evangelist distinction [28].

Another program from the community is FIWARE Experts. To be a part of the team of experts, the candidate must comply with the following requirements: made a contribution recognized by the FIWARE community, have an understanding of all the parts that form the FIWARE ecosystem, know how to architect FIWARE-based solutions in a scale and be up-to-date in the NGSI technology. The candidate must then submit its application for review, which has to contain a body of work that defines the knowledge acquired. The application will be reviewed by the examiners and will call the candidate for a face-to-face interview where both will discuss the previously mentioned requirements to check the level of the candidate. Finally, if the candidate passes all the examinations, they will be suitable for entering the Experts team and show their expertise of the FIWARE ecosystem in its entirety[29].

4.3.2. FIWARE Accelerator

The first program is FIWARE Accelerator. The purpose of this program is to support startups and small and medium enterprises (SME) in the development of projects based on the FIWARE technology to promote its usage and creation of smart solutions. The goal is for the startups to publish the innovative solutions developed into the market while gaining professional recognition among organizations and business opportunities, acquiring the label "Powered by FIWARE" [30]. Being part of this program increases the professional contacts as there is a network among the Global FIWARE Community and the participants of the Accelerator program.

Figure 8. FIWARE Accelerate logo

4.3.2.1. Origin and the first Accelerators

FIWARE Accelerator started in September 2014 when the European Union started a campaign mobilizing up to one hundred million euros in funding to support entrepreneurs in Europe and the rest of the world, to develop innovative applications based on the FIWARE technology. The program has since continued to offer opportunities to startups by giving special technical consultancy and networking with relevant clients and investors. The campaign initially funded sixteen Accelerators with eighty million euros to promote the FIWARE technology in Europe, as the organizations were chosen by the European Commission. Those organizations are spread across eight industrial sectors, each one specializing in one different sector, namely Smart Cities, health, agriculture, energy, media & content, logistics, transport and learning [31].

Figure 9. First 6 FIWARE Accelerators with the available funding and their field of application

The accelerator program is backed up by some bigger organizations that provide resources and helping tools to any startup that requires them. Furthermore, they support the startups with funding for the development of smart solutions in the field of knowledge the provider works in, counting with their supervision in the process of creation and at the time of publication. For example, two Accelerator partners for startups are Zabala Innovation Consulting in Spain and Gateway 49 Accelerator in Lübeck.

Figure 10. Next 10 FIWARE Accelerators with the available funding and their field of application.

4.3.2.2. Organization profile

Next, depending on the profile of the organization that wants to participate in the FIWARE Accelerator program, there can be two possible options: a startup or a business incubator. In the first case, the organization is looking to be adopted by a bigger enterprise that provides funding and the necessary resources to develop their smart solutions based on FIWARE. In the second case, the organization is much bigger and seeks to adopt a small enterprise, generally a startup, and provide the FIWARE tools required.

- Startup. Being part of the FIWARE Accelerator program as a startup comes with technical and business training to create new Smart solutions in any field, opportunities coming from events, challenges and hackathons with prize money, but also visibility for the startup in the FIWARE website in the form of blogs, news and Impact Stories, as well as in the FIWARE social media and Press Kits. Other advantages from being part of the program are networking, which allows the startup to communicate and meet with international professionals within the FIWARE ecosystem, including Experts and Evangelists; and the access to high-level resources in the FIWARE platform, like the Lab and test environment, while also taking advantage of the regional iHubs and training centers [32].

- Business incubator. In the case of a business incubator, the program is aimed at the Technology Parks, Capital Companies and Digital Innovation Hubs with training and coaching services. The program offers the organization the whole FIWARE platform, with the benefits of being royalty-free, the startups can begin working with ready-to-use products thanks to the open FIWARE NGSI standard. The training needed to manage the new technology and create new smart solutions is also provided to the main incubator and the related startups and SMEs. Additionally, there is a possibility of signing the Collaboration Agreement and partner with FIWARE, allowing the incubator to become an iHub location. Assistance to events is also provided, not only as an attendant but as a speaker in the related panels, as FIWARE gifts free tickets and access to reserved spaces in the Summit. Finally, being an incubator opens up networking possibilities with every other organization in the program, counting with experts in any field.

Country	Accelerator Coordinators	Consortium Partners	Country	Accelerator Coordinators	Consortium Partners
Belgium	3	17	Luxembourg	-	1
Bulgaria	-	1	Netherlands	1	12
Denmark	-	8	Poland	-	4
Estonia	1	2	Portugal	1	1
Finland	1	4	Serbia	1	3
France	1	3	Slovakia	-	2
Germany	3	10	Slovenia	-	1
Greece	-	3	Spain	2	15
Hungary	-	2	Sweden	-	1
Ireland	-	2	Turkey	-	2
Italy	2	12	United Kingdom	-	4
Lithuania	-	1			

Figure 11. Spread of the initial FIWARE Accelerators across Europe.

4.3.2.3. Accelerators examples

In order to show the efficacy of this program, two examples of startups that successfully benefitted from FIWARE Accelerator will be analyzed.

4.3.2.3.1. DigitAnimal

The first case is the Spanish startup company DigitAnimal. The proposed problem is that the benefit margins from livestock activities is minimal, as well as the mandatory electronic tagging of animals with costly GPS solutions, and that many software programs rely on human action to work properly. The proposed solution consisted in IoT wearables for animals that showed their health conditions and their location, sending the information to a cloud server where different Big Data algorithms generated the insights needed for farmers. With this system the human interaction and paperwork is reduced, whilst augmenting the margin of benefits and the compliance with the European laws by having the cattle located in every moment. This system benefits not only the farmers, but also the consumers, as they have truthful information about the state of the product; and the authorities, as they can check for meat infection and security compliance [33].

4.3.2.3.2. Everimpact

Other example that is more related to the field of Smart Cities is the case of the French startup Everimpact. The challenge to solve was to reduce the contamination in cities, measuring the values and locating their source. The climate change is undeniable, and most of the carbon dioxide emissions come from cities and their factories, so it is there where to put the most effort. The first step is to identify the sources of the emissions, and the second step is to assign funding to those locations, but it is a costly step and the decisions should not be taken lightly. The solution proposed by Everimpact is to measure the CO2 emissions by using the data coming from satellites and ground sensors. This way, cities can have a map at street and building level to pinpoint the exact location the emission is coming from, helping cities in focusing their climate actions and

check the results. Being able to monetize the CO2 reduction with taxes such as the Carbon Tax can prove beneficial to shift from a traditional and polluted city to a Smart and clean one. However, FIWARE is one important aspect of the implementation, as the software solution does not have to be built from scratch, using the Generic Enablers allow for a quick set-up of the program and is key in having advantage over the competition [34].

In conclusion, the FIWARE Accelerator program helps the smaller enterprises and startups to gain experience and build a network of contacts while developing useful Smart solutions that help cities and bring technology to sectors where it is most needed.

4.3.3. FIWARE Mundus

The second program is FIWARE Mundus. Its purpose is to promote the usage of FIWARE-based solutions and set-up the FIWARE ecosystems at a global level [35]. To achieve this goal, it looks for international membership, exploring common business strategies and promote the FIWARE activities and events in those countries where the platform is not yet in place. Apart from the main objective of promoting the adoption in Europe and the rest of the world of the FIWARE technology, the second objective is to ensure that the ecosystem is sustainable financially at medium and long term, appealing to stakeholders from the different countries where the project is being spread.

Figure 12. FIWARE Mundus logo.

In order to expand the FIWARE mission and the advantages of the technology, it is needed to deploy new FIWARE nodes in locations where the platform is not present, but also in countries with presence where cooperation opportunities may arise like sharing knowledge about FIWARE programs and best uses of the technology. Some activities currently being performed by FIWARE Mundus are helping public and private organizations in charge of innovation to support the usage of FIWARE and setting up the ecosystem at regional level with focus on making it sustainable and gather information about similar technologies from countries with other platforms and ecosystems in place and adapt those ideas into long-term investments for the FIWARE business model and technologies.

4.3.3.1. Global expansion

Next up is the global expansion of FIWARE-based solutions. As the main goal of Mundus program is to expand the ecosystem, it is natural to find new FIWARE ecosystems installed in locations across the globe in every continent. Some examples of such solutions can be found in countries like Mexico, where the collaboration between the local government and the private FIWARE program has created interesting and helpful projects focused on agriculture and cities.

The Healthcare initiative with FIWARE was based on a software application for mobile phones that collected physical information from clinical test to determine the risk of fall in adults. Another smart application in Mexico with help from FIWARE was the Green Route, a mobile application where users could check the route to reach a destination taking into account the health conditions of the users as well as the pollution, pollen or favorite transportation method [36].

Figure 13. Map of the expansion of the FIWARE technology. Blue markers are countries and orange ones are cities with operating FIWARE solutions.

4.3.3.2. FIWARE Mundus in the USA

Other examples of the work FIWARE Mundus does in the expansion of FIWARE technologies in developed countries can be found in the United States. Following the events of the 2016 Global City Team Challenge (GCTC) Expo with participation of the FIWARE Mundus community in USA, different projects were presented, such as “Leisurity” a collaboration between the cities of Valencia in Spain and Lindale in Texas that focuses on smart leisure like a smart parking system and monitorization of events; a smart city app in Hudson developed in the city of Logroño (Spain) where all the smart services are unified in a single place; the showcase of an app where all the citizens can send tickets about urban waste management in Guadalajara (Spain); the partnership between the city of Eindhoven (Netherlands) and the organization Atos leading a project to organize traffic information and data in a smartphone app; and finally the Italian cities of Turin, Genoa and Milan in conjunction with Engineering are developing a way to track waste management, establish smart parking system and prevent environmental disasters [37].

4.3.3.3. FIWARE Mundus in Canada

Continuing with the expansion in North America, in Canada, the final events of the ERA-CAN+ project, a collaboration between some important Canadian organizations from Edmonton

Research Park[3] and FIWARE Mundus reached an agreement to promote the technologies in the city of Edmonton, facilitating the adoption of FIWARE solutions in the local ecosystem. Thanks to this event and the FIWARE Mundus delegation in Canada, many regional stakeholders were willing to explore new alternatives based on FIWARE, for instance, the Defense Research and Development Canada (DRDC) was interested in the technology, and the Ministère de l'Économie, de la Science et de l'Innovation from the province of Québec looks forward to explore the partnership with FIWARE organizations [38]. Given the success of the partnership, FIWARE Mundus will continue its operations with European teams to create new technological opportunities in the country.

4.3.3.4. FIWARE Mundus in Asia

Moving on to another continent, in Asia there has been a couple of events where FIWARE projects promoted by the regional Mundus ecosystem were showcased. One of them was a workshop in June 2019 where more than thirty high-level participants from Hong Kong Office of the Government Chief Information Officer (OGCIO) along with WeGov.org and the EU Office to HK were reunited to address "The future of Open Governance & Innovation in Asian Cities" [39]. Another event took place in January 2019 where the European Chamber of Commerce in Hong Kong and the Hong Kong Science and Technology Parks Corporation (HKSTP) presented the conference "Building an open ecosystem for Smart City developments in Hong Kong and Europe". The panel, presented by two FIWARE Evangelists, explored and discussed the FIWARE framework and its possible implications and impacts in the new data models for the future [40]. The "Connected Cities Conference" in January 2019 in Hong Kong also counted with the participation of the local FIWARE Mundus and some Evangelists, in an event where projects examined the way technology can improve the lives in urban cities in Hong Kong but also in the rest of the Greater Bay Area[4] [41].

4.3.3.5. FIWARE Mundus in Africa

Finally, in the African continent there also have been a few events in the more developed countries with local FIWARE Mundus offices. The Maroc Numeric Cluster[5], participant of the Smart City Expo World Congress 2017 in Casablanca, invited the African FIWARE Mundus committee to this event. Both the MNC and the FIWARE Foundation signed an agreement that establishes, among other things, the set-up of a FIWARE node in Morocco, as well as propel an action plan to set a FIWARE environment in the city and develop new solutions using this technology.

---

[3] The Edmonton Research Park (ERP) is the name used to recognize more than 1,500 members at 55+ companies working in diverse fields, from biotechnology to energy.

[4] The Greater Bay Area is a megalopolis, consisting of nine cities and two special administrative regions in south China, including Hong Kong and Macau.

[5] The Maroc Numeric Cluster is a public/private governance structure formed by many organizations, from SME to bigger enterprises. It works in conjunction with the Moroccan Government with a plan to research new IT solutions for Security, Mobile services and Business software among others [43].

In conclusion, the FIWARE Mundus program is an interesting way for the world to know the importance of, not only the FIWARE ecosystem and technology, but the need of Smart solutions that help with the most actual topics like climate change or the smart inversion of funds.

4.3.4. FIWARE iHubs

The third program is FIWARE iHubs. Its purpose is to create a community around FIWARE technologies and working together to come up with smart solutions. A FIWARE iHub is the physical center for the adoption of FIWARE technologies among businesses in a given region. iHubs increase the competitiveness through modernization and, at a regional level, facilitate the diversification of the local economy and expansion of the local market. Additionally, they have to be operated by an organization member of the FIWARE Foundation.

Figure 14. FIWARE iHubs logo.

The FIWARE iHubs are the origin of innovation, creating new markets and staying relevant in the region, turning ideas into ready-to-use solutions, supporting the creation of new services and making an impact in the market or society of the region. The physical centers act as a way for private companies, public administrators, research institutions and universities to come together, benefit from multi-regional partner cooperation in the development of innovative solutions and build a network of contacts and interpersonal relationships that can influence organizations even outside the region [42].

4.3.4.1. Services provided

The services provided in the iHubs for local companies are technological consulting, training and research of the FIWARE technology, as well as support when joining the FIWARE Marketplace. Some other non-technological services the iHubs provide are market information, certification of developed solutions, individual coaching for SMEs, product analysis and organization of events.

Depending on the number of resources and services the iHub provides, the center belongs to a certain level in a classification. There are four categories of FIWARE iHubs, from Basic to three stars as the available services grow. Giving each iHub a place in the classification and approving the creation of new centers is made by the FIWARE iHubs Committee with the support of the Foundation, reviewing the candidatures presented and checking whether they comply with the performance levels required by the established classification.

The number of activities every Hub must perform is as follows:

- Training events and webinars on business models, technology use cases, access to finance and FIWARE Accelerator programs.

- FIWARE technology events for universities, research institutions and business.
- Support for local SMEs and startups to join the FIWARE Marketplace, as well as facilitating training and coaching in technical problems.
- Promoting the FIWARE ecosystem, local SMEs and startups at regional fairs and congresses.
- Communication activity in the regional social media.
- Partnership with the local government.

To determine the level in the classification for an iHub, the number, quality and frequency of all these activities is taken into account.

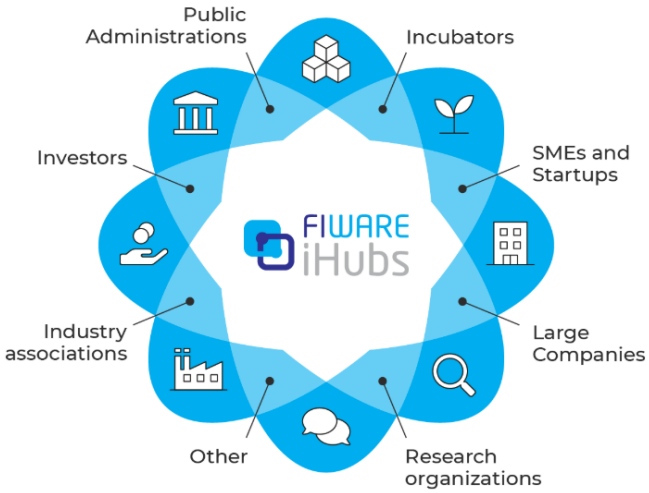

Figure 15. FIWARE iHubs map of participants.

4.3.4.2. Benefits for organizations

Besides, establishing and running an iHub grants some benefits to the organization.

- Involvement into the official FIWARE iHubs program. FIWARE iHubs will be recognized as official nodes of the iHubs network and as global FIWARE stakeholders, proving useful in business positioning and in the FIWARE ecosystem. They will also have a participating role in the FIWARE iHubs Support Committee.
- Right to include FIWARE brands and marketing material in the FIWARE iHub publications. Being able to use the FIWARE brand and its products for marketing purposes further improves the
- Inclusion on the FIWARE Foundation website. iHubs will appear in the official list of the Foundation and will be part of the FIWARE iHubs Committee.

- Privileged connection to the network of FIWARE members, experts and evangelists. As part of the iHubs network, they can establish a connection with every other iHub. Additionally, they will be granted privileged contact with FIWARE Foundation members, other FIWARE program members, Evangelists and Experts. Having access to those resources allow for more collaboration opportunities with organizations, meaning more funding and better services and positioning to influence other regional startups.

- Preferential access to FIWARE Lab. All iHubs are granted preferential access to the FIWARE Lab cloud sandbox environment, setting the virtual infrastructure to develop, test and run FIWARE-based solutions.

- Support to activities by the FIWARE Foundation. The Foundation continuously organizes events, hackathons and conferences. Not only will iHubs have ticket booking preference, but they will also be active participants in those events, gaining international recognition and support for their work. In addition, the Foundation commits to give financial and press support to the activities organized by the iHub whenever possible.

- Direct connection with FIWARE Acceleration Programs. As the FIWARE Accelerate aims at promoting the use of FIWARE technologies among startups and SMEs, iHubs will benefit from having a direct connection between both programs, because many business and partnership opportunities can arise as well as extra funding and training for the incubated organizations.

- Certification to give official FIWARE training and coaching. If the requirements are met, the iHubs will be able to provide official training on FIWARE technologies, with the option of becoming a certified entity.

- Promotion of local companies at international events. If the FIWARE Foundation attends or organizes an event or conference, the iHubs will have preferential position to promote the regional companies they want. This will help in building a network of business contacts and gain visibility for new customers and potential investors.

Depending on the time of experience with FIWARE technology there are two types of applicants, incubated iHubs or established iHub.

On the one hand, Incubated FIWARE iHub application is for organizations that do not have enough experience in the FIWARE ecosystem and want to start activities as an iHub. The list of activities to perform in the next year must be provided along with a list of resources, similar to the Basic iHub level.

On the other hand, the Established FIWARE iHub application is for organizations that have been running activities about the FIWARE ecosystem, its technology and associated with the iHubs, but are not yet recognized as such. These applicants will be placed in a level from one to four stars and must also provide evidence of the activities they perform and the resources they have.

4.3.4.3. Classification levels

Every FIWARE iHub must ensure that any business and organization in the region has access to FIWARE technology *"at a working distance"*. The classification of levels the iHubs can belong to is based on the number and quality of activities grouped in five categories shown in the image below.

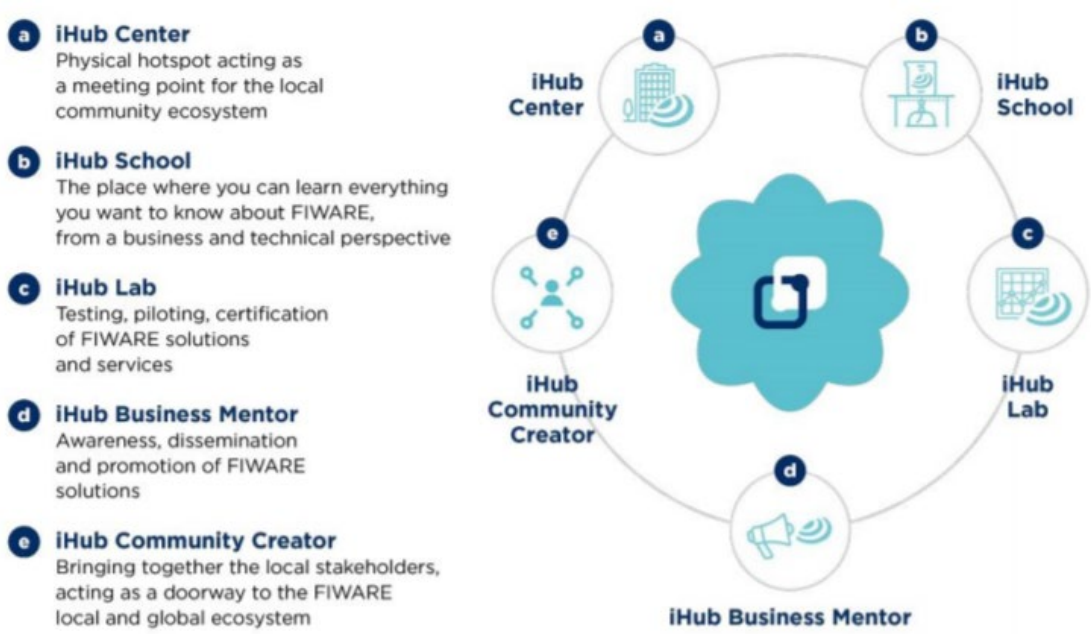

Figure 16. FIWARE iHubs activity groups.

Besides the activities and its groups, the iHubs are placed in a position that serves as a quality measure for the amount and quality of the activities performed in that iHub, differentiating it from other iHubs around the world. The classification also identifies the technical and professional requirements iHub should meet[44].

4.3.4.3.1. Basic level

In the basic level, iHubs offer teaching and training through events and are active at fairs and congresses promoting FIWARE technology. Cooperating with the local industry, iHubs organize workshops with businesses and associations. Regarding the physical space, iHubs ought to be in a building capable of hosting presentations and holding technical equipment, where there should also be a dedicated FIWARE space with promotion material like brochures, flyers and t-shirts available to visitors.

An example of a FIWARE iHub in the Basic level is located in Geneva, Switzerland, called UDG FIWARE iHub. It is run by the organization UDG Alliance, and it is dedicated to the field of Smart Cities and Smart Energy [45].

4.3.4.3.2. Standard Services (1 Star)

iHubs in the standard service level offer all the basic services from the previous level but at an advanced level. They run the iHub Lab which offers workshops and testing of "Powered by FIWARE" solutions and "FIWARE Ready" technologies, as well as organizing FIWARE dissemination events, collaborating with local universities and other iHubs and running workshops for trainers, research groups and public staff. iHubs promote FIWARE Accelerator programs and manages community user accounts for the FIWARE Lab.

In this level, the iHubs have a school that gives the iHub staff access to dedicated FIWARE training and certification activities. In addition, the iHubs should connect to the local industry through agreements with technology parks and city councils.

An example of a FIWARE iHub in the Standard Services level is located in Perugia, Italy, called FIWARE Innova Hub. It is dedicated to Smart Cities and Smart Agri-food [46].

4.3.4.3.3. Advanced Services (2 Stars)

In this level, iHubs have the characteristics of Standard Services (1 Star) iHubs but offering training workshops for teachers, students, research groups and municipal staff at advanced or expert level, as well as online or presential coaching specialized in business. These iHubs can operate environments for testing and have a dedicated showroom for "Powered by FIWARE" solutions and "FIWARE Ready" technologies in demo phase.

The iHub school now has physical space for running workshops, hackathons, training courses and a Smart Lab open to the public.

Regarding local communication, iHubs should create FIWARE working groups with local industry, universities, associations and enterprises. Furthermore, the iHub should connect to the rest of the FIWARE Community, actively participating in the FIWARE Summit and other FIWARE community events.

Additionally, iHubs should also promote solutions along with other iHubs, getting involved in FIWARE Accelerator programs and actively collaborating with FIWARE Mundus. Finally, to show the status of a 2 Star level, contributing to the FIWARE website and posting in the FIWARE blog is almost mandatory to reflect a strong compromise with the ecosystem.

An example of a FIWARE iHub in the Advanced Services level is located in Logroño, Spain, called FIWARE Rioja. It is formed by JIG Internet Consulting and Bosonit, two IT companies with experience in FIWARE technology, and it is dedicated to Smart Cities, Smart Energy and Smart Agri-food [47].

4.3.4.3.4. Premium Services (3 Stars)

Lastly, the highest level a FIWARE iHub can achieve is the Premium Services status. In this category, iHubs should offer training and workshop events, SMEs or startups incubation and bring

support to development of FIWARE prototypes. The iHub should meet the requirements to certify developers and trainers of "Powered by FIWARE" solutions, "FIWARE Ready" IoT devices and Generic Enablers. They should support the deployment of FIWARE pilots in the market, create promotion events for FIWARE-based solutions and technologies, help SMEs and startups to reach clients or customers and communicate activities in regional media.

In addition, the iHub should feature a dedicated showroom where products in pilot and production phases are showcased. The iHub should have a dedicated social network manager who posts regularly in popular blogs, is connected to fairs and press media and collaborates with FIWARE Foundation Community Managers in Social Media activities. Finally, a 3 Star iHub should connect with the rest of the FIWARE Community, actively participate in the FIWARE iHub Committee and provide certification services to other FIWARE iHubs.

The only FIWARE iHub in the Premium Services level is located in Seville and Málaga, Spain, called FIWARE Zone. It is a partnership between Telefónica and the regional government, and it is dedicated to Smart Cities, Smart Agri-food, industry, robotics and ports [48].

	iHub	Markets	Location	Service Level	
1	FIWARE Zone	Smart Cities, Smart AgriFood, Industry, Robotics, Ports	Seville/Málaga, Spain	★★★	Premium Services
2	FIWARE Rioja	Smart Cities, Smart Energy, Smart AgriFood	Logroño, Spain	★★	Advanced Services
3	FIWARE Space	Smart Cities, Smart AgriFood	Badajoz, Spain	★★	
4	Future City iHub	Smart Cities	Amersfoort, the Netherlands	★	Standard Services
5	FIWARE Innova iHub	Smart Cities, Smart AgriFood	Perugia, Italy	★	
6	IoT Booster	Smart Cities, Smart AgriFood, Smart Industry	Saint Quentin, France	★	
7	UberHub FIWARE iHub	Smart Cities	Uberlândia, Brazil	★	
8	Lanif	Smart Cities	México City, Mexico	★	
9	FiiHUB	Multi Domains	Canary Island, Spain Azores, Madeira, Portugal	★	
10	Arnaldo Castro Servicios y Tecnologia S.A.	Smart Cities	Montevideo, Uruguay	☆	Basic Services
11	Ciemsa Montevideo	Smart Cities	Montevideo, Uruguay	☆	
12	UDG FIWARE iHub	Smart Energy, Smart Cities	Geneva, Switzerland	☆	
13	Detecon FIWARE iHub	Smart Cities, Smart Energy	Cologne, Germany	●	Incubated
14	Instituto Ciudades del Futuro	Smart Cities	La Plata, Argentina	●	
15	iHub FIWARE Bridge	Smart Cities	Tunis, Tunisia	●	

Figure 17. List of running FIWARE iHubs, with working field, location and service level.

4.3.4.4. FIWARE iHubs expansion

FIWARE iHubs program is set to spread the knowledge of the FIWARE technology and ecosystem in regions with help from local organizations and business. As with FIWARE Mundus, the iHubs are located in every continent but Asia, where there is planned a future iHub, but not yet established (Figure 17). Originally, only Europe had iHubs, as there is where the whole FIWARE project started, however, more and more organizations have recently been interested in the technology and have been creating new iHubs in different countries to develop Smart solutions that solve specifical problems they have.

The first iHub open was the FIWARE Zone in the city of Seville, Spain, in January 2017, with a later establishment in the city of Málaga. In words of the director of the FIWARE Zone: *"It is a space where entrepreneurs, large companies, universities, the public administration and end users will meet together and develop innovative smart solutions based on FIWARE. Where training, certification and support of FIWARE technologies is provided and experimentation or showcase of FIWARE solutions and FIWARE-ready technologies will be offered."* [49]. Regarding the activities performed in those iHubs were the preparation of events about the FIWARE technology and the testing and development of FIWARE-based solutions addressing Smart Cities and Agri-food, with a dedicated space for co-working, as well as providing access to different IoT devices and sensors to test them in a real-life environment.

At the time of opening the FIWARE Zone, the FIWARE iHubs program was not yet properly identified, but it laid the foundation of what a FIWARE iHub should be. Later in 2018, the iHubs program was defined, and the first eleven iHubs were open. Those iHubs were [50]:

- **FIWARE Zone** from Seville/Malaga, Spain
- **FIWARE Rioja** from Logroño, Spain
- **IoT Booster** from Saint-Quentin, France
- **UberHub FIWARE iHub** from Uberlândia, Brazil
- **CIEMSA Montevideo** from Montevideo, Uruguay
- **FIWAREMAC (FIMAC) iHub** from Santa Cruz de Tenerife, Spain
- **Future Mobility FiHUB Thessaloniki** from Thessaloniki, Greece
- **Laboratorio Nacional de Internet del Futuro – LaNIF** (National Future Internet Laboratory) from Mexico City, Mexico
- **UDG-FIWARE-iHub-Western-Switzerland** from Geneva, Switzerland
- **Aplis Hub** from Prague, Czech Republic
- **iHUB Umbria** from Perugia, Italy
- **Future City iHub** from Amersfoort, the Netherlands
- **SprintPoint** from Cluj-Napoca, Romania.

Figure 18. Map of actual and future FIWARE iHubs location and member count per continent.

The latest iHubs open were in Austin - Texas, United States of America, Lagoa – Azores and Funchal – Madeira (Portugal) in 2020 [51]. In the Texas iHub, the conglomerate of companies that came together to form it are Volpi Ventures, Top Energy USA, Zorrito Capital, and ITD America, and the projects performed in the iHub are related to Smart Cities, Smart Health, Smart Energy and Agrifood, helping the local economy with the realization of workshops and events. In the Madeira iHub the founder was the Madeira Interactive Technologies Institute, which had been developing activities previous to the creation of the iHub. It is focused on Smart Energy, Smart Cities, Smart Industry and Agrifood, but the main objective is to spread information and training in the FIWARE Framework, IoT and Big Data technologies for their adoption in the regional market and open the economy to these new open standards. Finally, the Azores iHub is located in the Nonagon Science and Technology Park in Lagoa, and is focused on Tourism 4.0, Smart Cities and Smart Energy, looking for new ways of creating business opportunities to the local SMEs by promoting the use of technological solutions based on FIWARE and open standards.

In conclusion, the FIWARE iHubs program works to spread the FIWARE technology to local areas and create partnership between the bigger organizations that run the iHub and the regional SMEs, startups and government to develop Smart solutions that solve specific problems in such region. However, the participants of the program must carry out mandatory activities and comply with some requisites depending on the level of development of the iHub. Finally, the iHubs program continues to reach new countries, organizations and stakeholders attracted by the open nature of the FIWARE ecosystem.

4.3.5. FIWARE Lab

The fourth program is FIWARE Lab. FIWARE Lab is the non-commercial sandbox environment of the FIWARE Community, offering the capability to innovate and experiment with the FIWARE Technologies free of charge. Organizations and individuals can test FIWARE technologies as well as their applications within the FIWARE Lab, with the possibility to exploit

Open Data published by cities and other organizations. FIWARE Lab is deployed over a geographically distributed network of federated FIWARE Lab nodes, each one mapped to one or more datacenters where an OpenStack instance has been deployed, operated by a specific organization. FIWARE Lab nodes are also part of local ecosystems with FIWARE iHubs supporting and promoting access to the technology [52].

Figure 19. FIWARE Lab logo.

The access to FIWARE Lab is free, and everyone can join and experiment with the sandbox environment testing the FIWARE Generic Enablers to create smart solutions after the testing is done. There are two ways of joining the FIWARE Lab program: as a normal user to test the software and the environment, and as a provider of virtual resources as a node in the FIWARE Lab network.

4.3.5.1. Join FIWARE Lab as a user

In order to join the FIWARE Lab environment and test smart solutions, there are a few steps to be made previously.

First of all, an account creation request must be sent to the FIWARE Lab team. The request is done with the fulfillment of a form where some data needs to be entered depending on the type of account that wish to be created. It can be a "Trial" account or a "Community account" [53].

4.3.5.1.1. Trial account

A trial account grants access to the FIWARE Lab resources with limited actions for a period of fourteen days. Testing of the FIWARE technologies can be made in this timeframe, but when it expires, the account gets disabled and the associated resources freed. To continue using FIWARE Lab, the user must send a Community account upgrade request before the time expires.

Regarding the information to be sent in the request, it is needed the name of the user, the email address and the FIWARE Lab region, besides accepting the Terms and Conditions of FIWARE. In the FIWARE Lab region selection, there are, at the time of writing, four options: Lannion5 (France), Noida (India), Poznan(Czech Republic) and Vicenza (Italy) [54], each one referencing the city where the physical node and the servers are located.

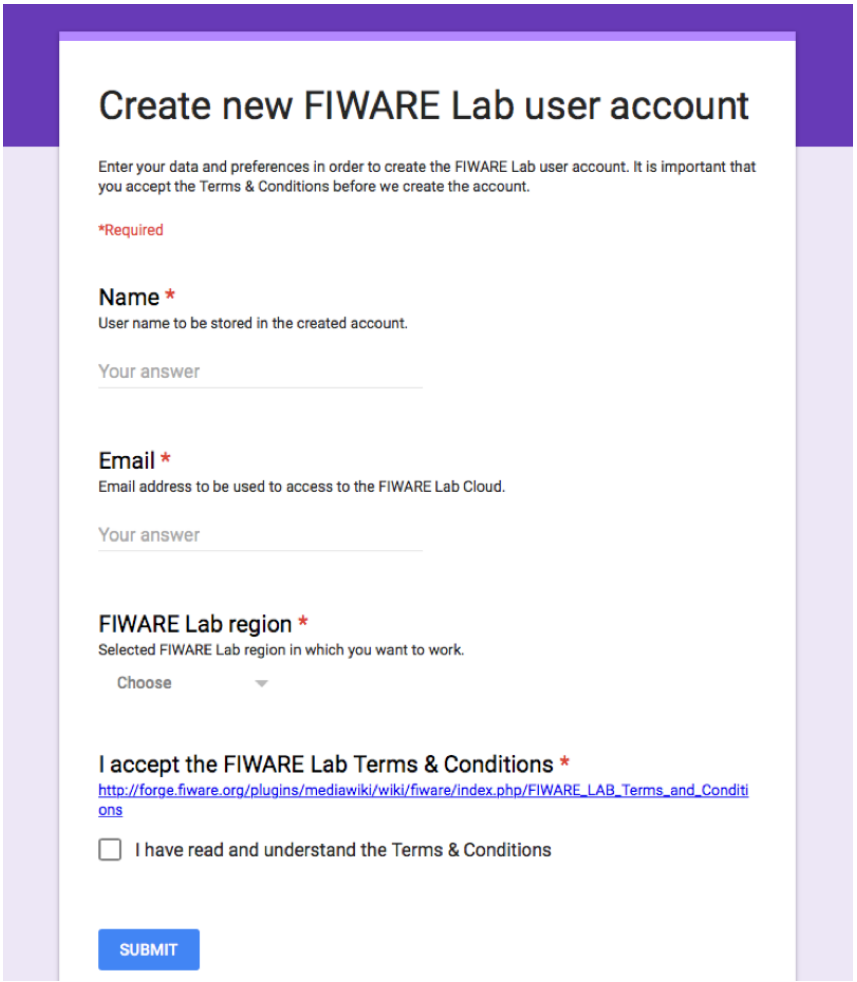

Figure 20. FIWARE Lab Trial account form.

After that, the user will receive an email coming from the admin of the FIWARE Lab node regional node with the login information needed to sign in the FIWARE Lab Cloud Portal. Then, the user will be able to successfully login and access the resources in FIWARE Lab for fourteen days.

4.3.5.1.2. Community account

A Community account grants the user access to the FIWARE Lab resources for a long period of time to develop smart applications based on FIWARE technologies. A formal request must be filled where the main objective of the application to develop must be clarified, as well as the planned resources to use from the FIWARE Lab. The request has to pass a thorough evaluation, and if the result is positive, the user will have access to the resources for a period of nine months, with the possibility of extending the duration if the application has not been finished yet.

Regarding the information to fulfill, the Community account requires more data from the user. In addition to the name, email and regional node, the user must specify the name of the startup/SME, the number of developers and the department they belong, if the company is part of an accelerator

program; and some technical information such as the number of virtual machines, RAM, virtual CPUs, storage, and public IPs needed for the project.

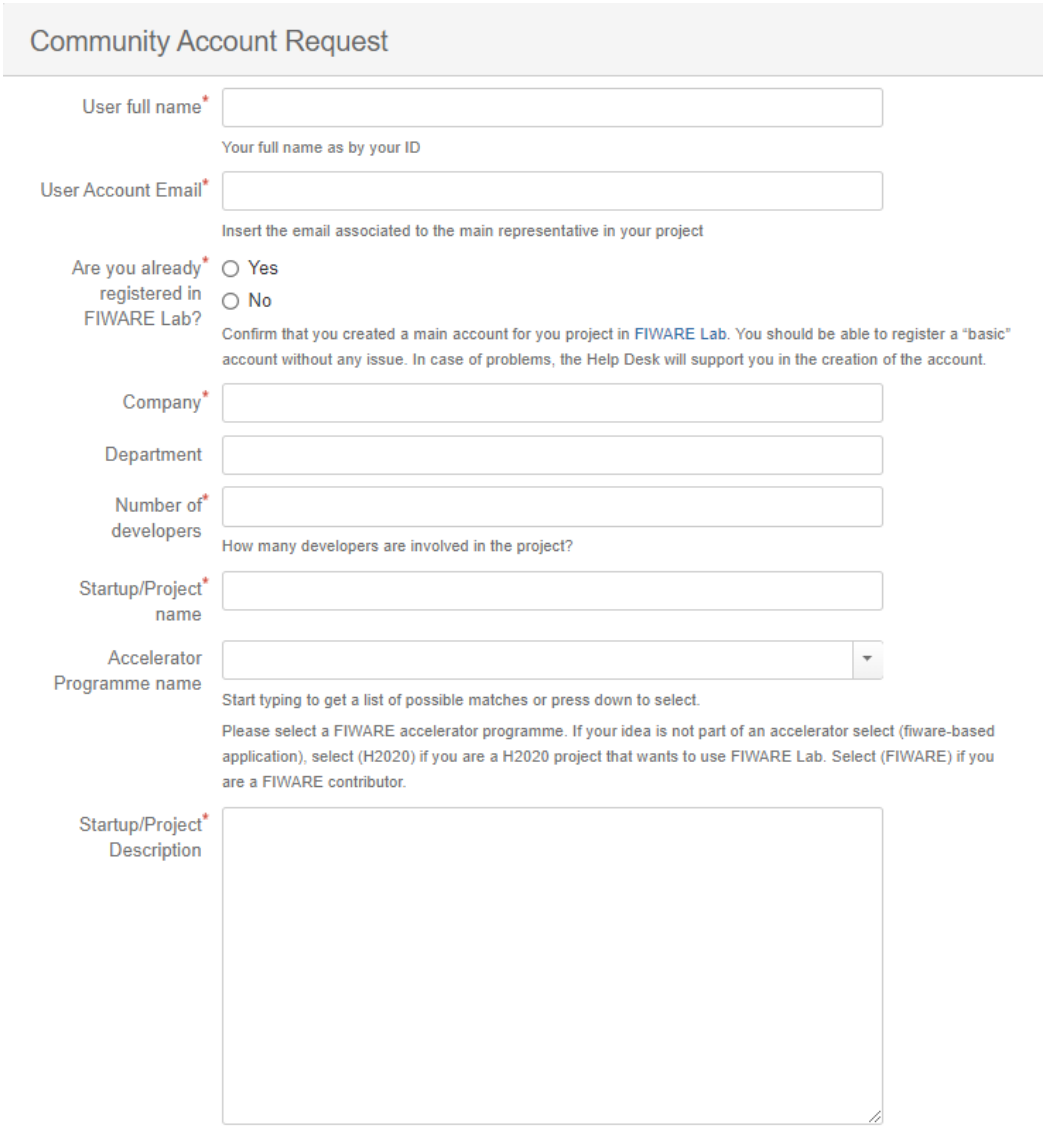

Figure 21. Part of a FIWARE Lab Community account form.

If the application gets approved, the user will be able to join FIWARE Lab and make use of the resources with the limits established in the account creation request form.

4.3.5.2. Join FIWARE Lab as a node

In order to become an active node in the FIWARE Lab network, a few requirements must be met. They are divided in two groups: hardware and network requirements.

About the network requirements, they depend on the number of end-users the node will support. The machines need up to 10 virtual cores, 20 Gb RAM, 150 Gb of hard drive and at least 22 Gb for images management.

Regarding the network requirements, the connectivity capacity will be used to connect to the backbone of the FIWARE Lab to support node management operations and provide connectivity to deployed services for end-users. Applicants must also provide a pool of public IPs that can be used during the OpenStack deployment (in order to expose the public API) and at a later time by the FIWARE Lab end-users. The technical requirements are as follows [55]:

- 1 Gbps connectivity for the backbone.
- 100 Mbps Internet connectivity for end-users.
- Firewall to ensure security.
- At least 1 public IPv4 available for each end-user (20 public IPs for 20 community users) plus public IPs for the different OpenStack services.
- IPv6 support desirable, although not required.

Once the hardware requirements are met, a few more steps must be performed to gain access to the FIWARE Lab program as an active node. In the following table the steps are presented, as well as the person responsible for the accomplishment of the step [56].

TASK	TASK DESCRIPTION	TASK OWNER
00	Sign the Letter to become a new FIWARE Lab node.	Node Admin
01	Provide contacts information.	Node Admin
02	Insert the new node within the agenda of the weekly meetings.	FIWARE Lab Admins
03	Join the weekly meetings.	Node Admin
04	Join the fiware-lab-federation-nodes@lists.fiware.org mail list.	Node Admin + FIWARE Lab Admins
05	Insert the new node in Jira for helpdesk and FLUA management.	FIWARE Lab Admins
06	Sign in for Jira account: https://jira.fiware.org	Node Admin
07	Request the creation of the FIWARE Lab administrator account: https://cloud.lab.fiware.org	Node Admin
08	Webinar to explain how to use the main tools and methodology such as Jira, Help tickets, FLUAs.	Node Admin + FIWARE Lab Admins
09	Install the local OpenStack node.	Node Admin
10	Federate the node.	Node Admin
11	Update the federation data within the corresponding JIRA work item created for the federation process.	Node Admin
12	Install and configure the monitoring system.	Node Admin + FIWARE Lab Admins
13	Configure the Sanity Check.	Node Admins + FIWARE Lab Admins
14	Insert the new node within Infographic and Health Status.	Node Admins + FIWARE Lab Admins
15	Synchronize GE image list.	FIWARE Lab Admins

Table 2 Table with the steps to become a FIWARE Lab node.

Should the node be discontinued from active service, the next steps must be performed to notify the necessary users and administrators of its closure.

TASK	TASK DESCRIPTION	TASK OWNER
01	Inform the FIWARE Lab management about your decision at least 1 month before the shutdown.	Node Admin
02	Inform all FIWARE Lab users about the shutdown via the FIWARE Lab notification tool.	FIWARE Lab Admins
03	Take care of the migration of all active users toward a persistent node.	Node Admin + FIWARE Lab Admins
04	Disconnect the node from centralized Keystone.	Node Admin + FIWARE Lab Admins
05	Delete the node from the list of available nodes in JIRA, FLUA, Sanity Check, Infographic, Health Status.	FIWARE Lab Admins
06	Delete from the fiware-lab-federation-nodes@lists.fiware.org mail list.	FIWARE Lab Admins
07	Delete the node from the weekly meeting agenda/minute.	FIWARE Lab Admins
08	Ask the node to send a final report/lessons learnt.	FIWARE Lab Admins + Node Admin

Table 3 Table with the steps to exit the FIWARE Lab program.

4.3.5.2.1. OpenStack

FIWARE Lab nodes are based on OpenStack distributions. OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources in a datacenter, all managed through APIs with common authentication mechanisms, while also having a dashboard available, to give administrators control and empowering their users to provision resources through a web interface [57].

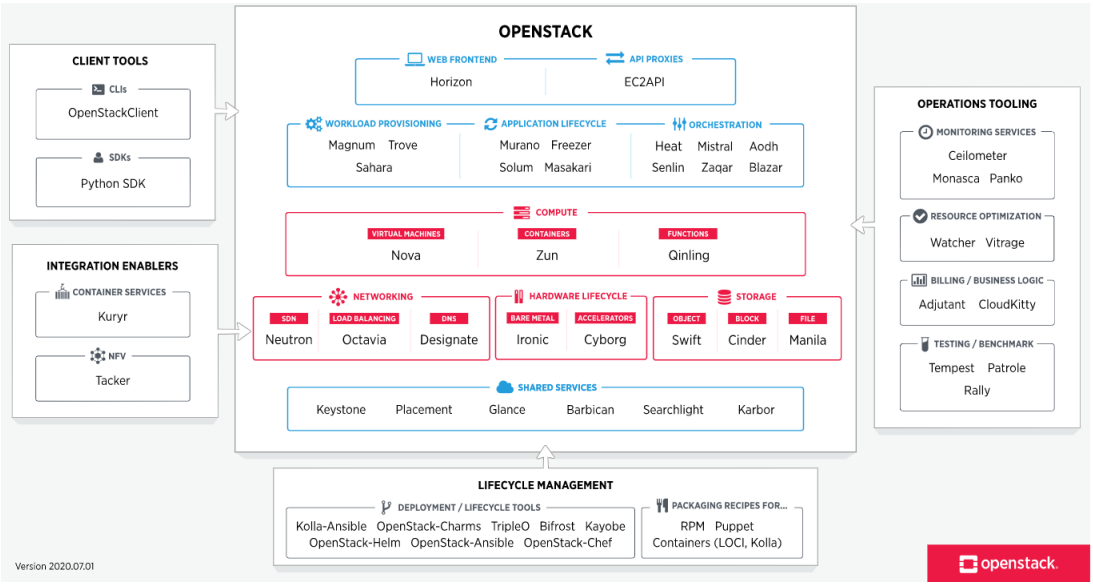

Figure 22. OpenStack and the services forming it

Regarding the update lifecycle proposed by FIWARE, the policy is to be two updates behind the official OpenStack release, to ensure that any possible security and performance issue is fixed and to prevent end-of-life unsupported versions [58]. If the version is not properly updated, the FIWARE services will not work until it is upgraded to the latest supported version.

4.3.5.2.1.1. OpenStack services requirements

From the image above (Figure 22. OpenStack and the services forming it), FIWARE Lab has some mandatory services that must be installed to mantain the compatibility among all the components in the node. The requirements are as follows:

- OpenStack Nova using Linux Kernel-based Virtual Machine (KVM) as hypervisor.
- OpenStack Glance with Swift as default backend type.
- OpenStack Cinder with Logical Volume Manager (LVM) for device mapping.
- OpenStack Neutron with Open vSwitch or VxLAN tunnels, to ensure that floating IPs are available to users.
- OpenStack Ceilometer with MongoDB.
- For the initial installation, OpenStack Keystone and OpenStack Horizon must be installed and then removed to work with the FIWARE-based component instead (FIWARE Lab keystone and FIWARE Lab Cloud Portal respectively).

Besides the mandatory requirements, there are some other optional components that can be installed, namely

- OpenStack Swift with 3 replication factor value.
- OpenStack Murano with OpenStack Heat for Platform as a Service capabilities.
- OpenStack Magnum with Swarm for managed docker.

4.3.5.2.1.2. OpenStack network configuration

FIWARE Lab has already defined a common name for the networks that must be used by all the nodes. It helps the different services deployed on top of OpenStack to work with the correct network without needing any special configuration [59].

- public-ext-net-01. This is the Public External network, a non-shared network with available subnets that provide public and routable IPv4 addresses. Additionally, nodes can configure IPv6 on this network. It is not visible to attach directly OpenStack Instances on it, but to allocate public IPs to be used by tenants.

- node-int-net-01. It is a shared tenant network providing DHCP IPv4 (and IPv6 in the future) addresses. This network is visible for all tenants and therefore anyone can attach OpenStack instances on it. Any node could choose its own network range since this should not collide with other node's networks.

4.3.5.2.2. Connecting to the FIWARE network

Then, once all the hardware and software requirements are met, and OpenStack properly set up, the node can be connected to the network.

The first step is to have a service account for each OpenStack service provided and an administrator account registered in the FIWARE Lab Keystone. An admin account for administrative tasks such as communicating with the users of your node or managing the users accounts status, and a service account to configure the OpenStack services and allow them to validate tokens with the FIWARE Lab Keystone. To create these accounts, the admin of the node has to send an email to the FIWARE Lab Keystone administrators providing the name of the node, the list of services to register and the email of the node admin.

The second step is to register the node endpoints in the FIWARE Lab Service Catalogue, providing the information to the FIWARE Lab Keystone administrators. Each service has its own endpoints and the information changes between them, but the list of endpoints must follow a given template. For example, the endpoints of OpenStack Nova must be given with the following format.

Nova: (service type: compute, service name: nova)

```
"adminURL": "http://<IP_ADRRESS>:8774/v2/$(tenant_id)s"

"internalURL": "http://<IP_ADRRESS>:8774/v2/$(tenant_id)s"

"publicURL": "http://<IP_ADRRESS>:8774/v2/$(tenant_id)s"
```

Figure 23. Template of Nova endpoints

After the endpoints are registered and the accounts obtained, the next step is to change the Keystone service to which the services are validating the tokens, because at the beginning, the OpenStack services are connected to the local Keystone and they have to be connected to the FIWARE Lab Keystone.

Finally, once all the previous steps are met, the node should be connected to the FIWARE Lab Keystone. A few key actions must be performed to verify the connection, such as deploying a virtual machine, deploying a network or deploying a floating IP to a VM and checking that it is reachable from the outside. If everything is correct, the node is ready to be published. The last step is to contact the FIWARE Lab Keystone administrators, and once they approve the submission, the node will be published.

4.3.5.3. Creating a virtual machine

Once the user has successfully logged in in the FIWARE Lab Cloud Portal, the resources available to use will vary depending on the type of account the user has. However, the options and menus will be practically the same for both types of account.

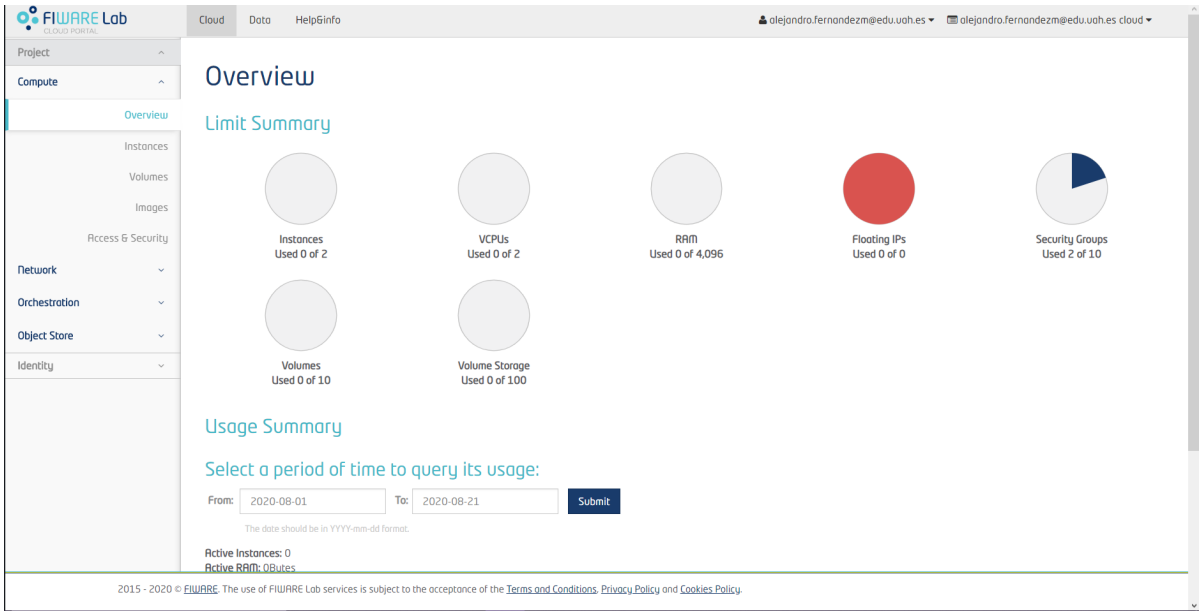

Figure 24. Main menu of FIWARE Lab Cloud Portal

In the main menu (Figure 24), in the middle of the screen, there is represented information with the instances and resources that are actually being used. On the left side of the screen, there is a vertical menu where all the options can be found, such as creating a virtual machine, defining security groups or choosing a pool of IPs for the projects. On the bottom of the screen there is a usage summary where it can be monitored the usage of the resources and their availability within a timeframe.

To create a virtual machine with all the Generic Enablers needed for the development of FIWARE-based applications, a few steps must be followed.

- First, there needs to be a security group created. A security group defines the ports that will be opened in the virtual machine. There is already a security group created called "default" with some predetermined ports open, but the user can choose to create new groups with new rules. In order to control the virtual machine, there are a few mandatory ports to open to ensure the machine works properly, such as port 22 (SSH) to control the machine remotely and port 80 (HTTP) to get access to the internet.

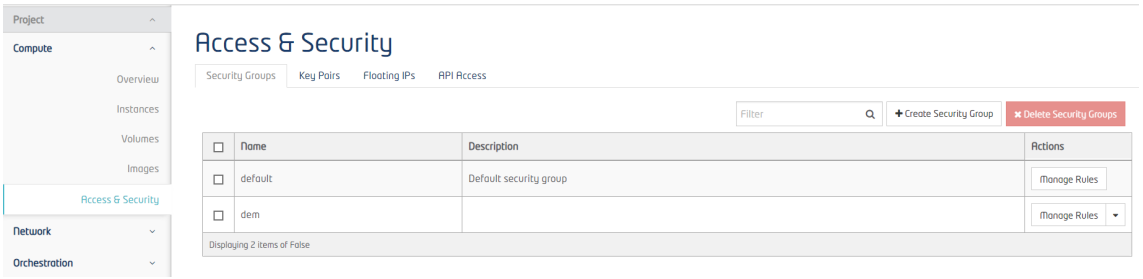

Figure 25. Security groups main menu

- Second, there needs to be a security key pair established to connect to the virtual machines via SSH. To download the key pair, in the key pairs section of the access & security group there is a button to create a new key pair, then it will be auto downloaded into the user computer. If the key is not downloaded in this step, the SSH connection may not properly work later.

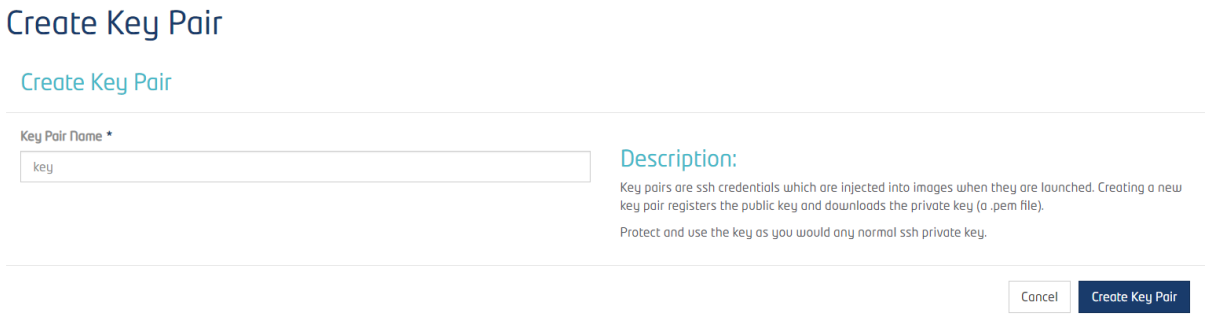

Figure 26. Creating a key pair

- Finally, the virtual machine can be created. From the Compute menu, the Instances submenu contains the tools to create a virtual machine. Selecting the launch instance button will prompt a wizard where the options of the desired virtual machine can be chosen. The availability of resources will determine the options of the machine, in the case of a Community account the resources available will be more extensive, while those of a Trial account will be reduced. One important option to take into account is Flavor. Flavor determines the basic resources of a virtual machine, such as virtual CPU cores, RAM memory and Hard Disk space. If the user account does not have enough resources to launch a Flavor, the wizard will not let the user select it, and will show a screen with the resources that are exceeding the quota.

Flavor	Memory (in MB)	Disk (in GB)	Virtual CPUs
m1.tiny	512	1	1
m1.small	2048	20	1
m1.medium	4096	40	2

m1.large	8192	80	4
m1.xlarge	16384	160	8

Table 4 Flavor options and the resources required [60]

After the flavor has been selected, the next option is the source from which the instances will boot. They can boot from a preexisting image (in ISO format), from a previous snapshot or from a volume. FIWARE Lab offers a few images to the user, like Ubuntu, CentOS and Debian, and other utilities useful for FIWARE applications like Orion Docker or MongoDB Docker [61].

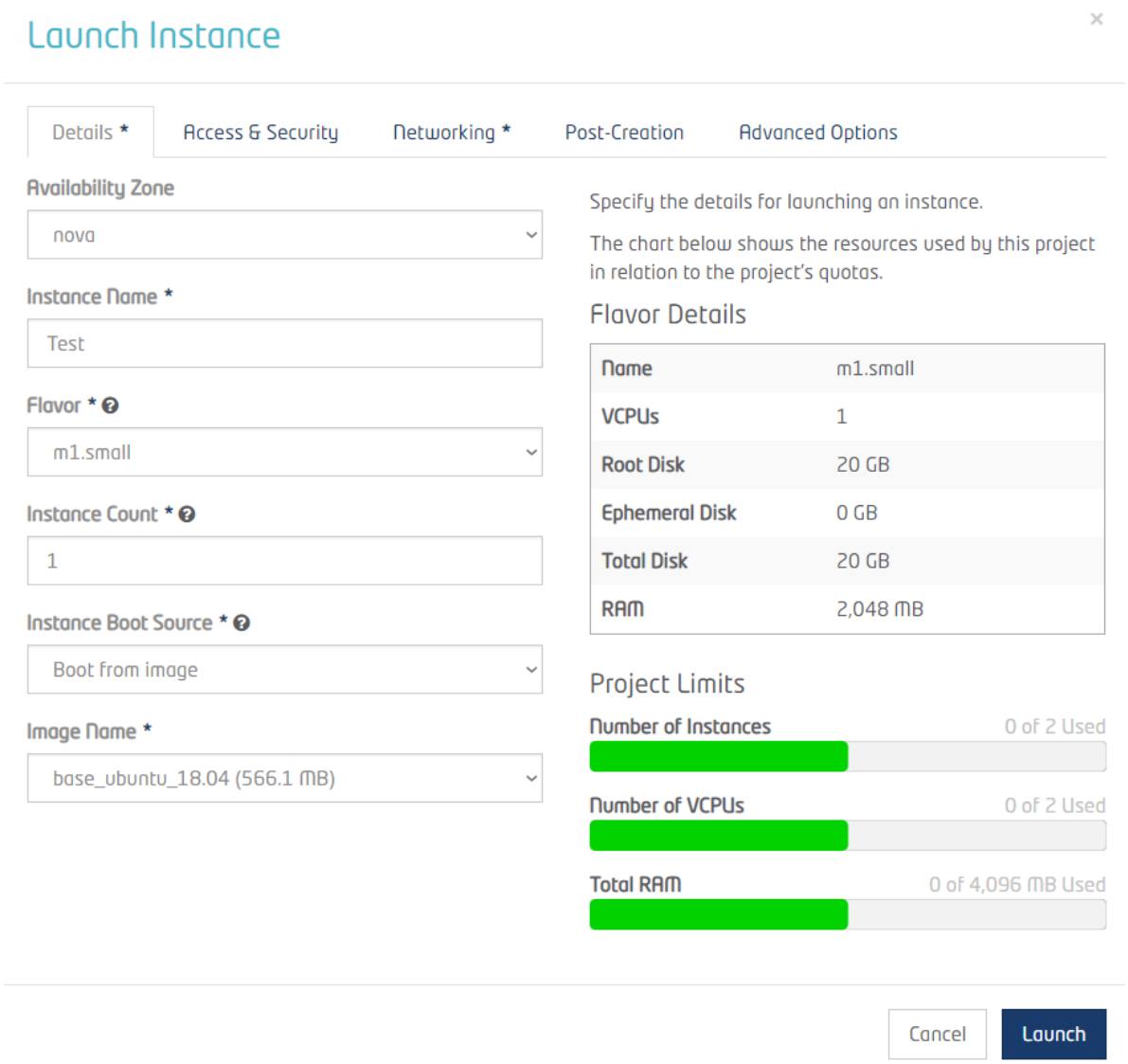

Figure 27. Launch instance options

Next, once the desired flavor and image has been selected, the previously created security group should be chosen to apply the port configuration, as well as the key pair downloaded to allow the user computer to be able to access the machine with a SSH

connection. Finally, a network must be selected from the available, the Trial user will have just one while the Community user may have more to choose. Once all the configuration has been done, the instance is ready to be launched. It will take some time for the machine to turn on and apply the configuration given, but then the user can connect to it with any SSH client as long as the key is present and start working on the FIWARE-based application.

4.3.5.4. Datasets

Creating virtual machines and applications is useless without data to test the solution developed and the proper functioning of the modules installed. FIWARE Lab also provides different datasets coming from various cities, businesses and universities, all free of use for users whether they have a FIWARE Lab account or not. At the time of writing, there are about three thousand datasets from twenty-three organizations and nineteen groups of classification [62]. Those datasets represent information from energy plants, city maps of districts, demographic trends, and many more. Some of the cities that provide data are Seville and Málaga in Spain, Amsterdam and Rotterdam in the Netherlands, and Florence and Trentino in Italy; while some universities such as Universidade de Sao Paulo or RWTH Aachen University also support the platform with useful data.

Reinforcing the open nature of the FIWARE ecosystem, the datasets are provided with the help of CKAN, the world's leading open-source data portal platform, used by governments and user groups to facilitate the accessing of data to end users [63].

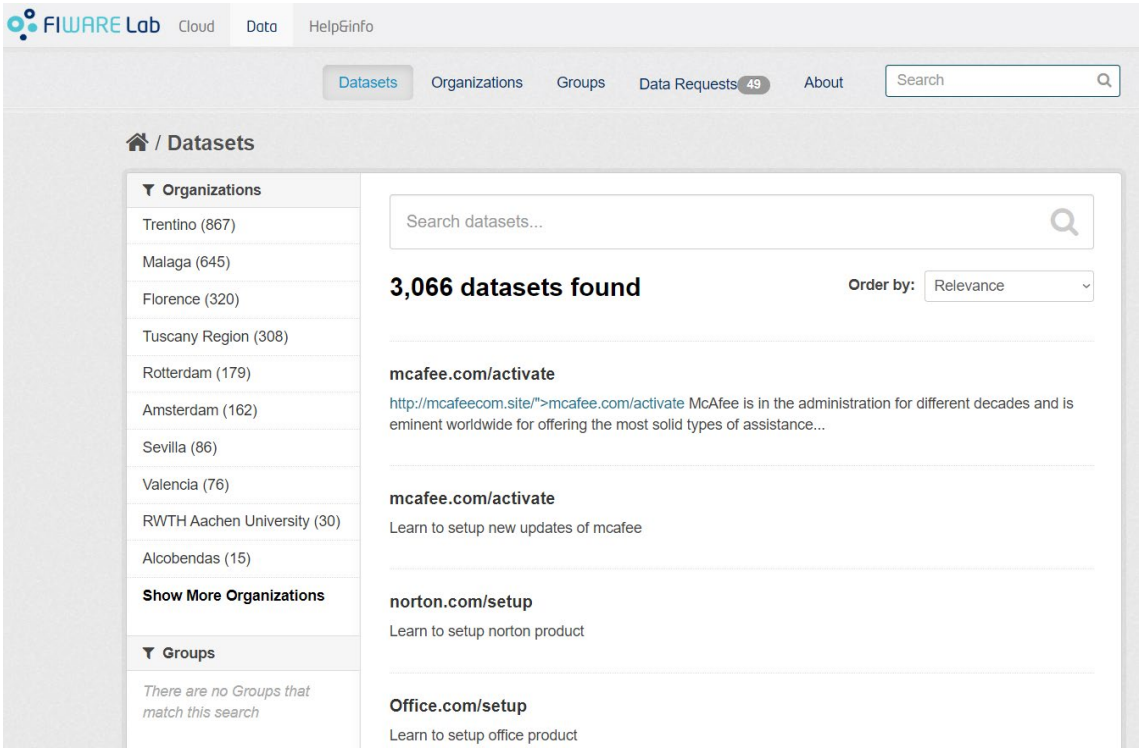

Figure 28. Datasets available in FIWARE Lab

After selecting a desired dataset, a screen appears with some useful information about such dataset, like a description of the providing organization, the type of license of the data and the proper dataset to download with its filetype. Some filetypes that can be found are the most common types to store information such as XML, CSV, JSON and PDF. Moreover, certain datasets will have an option to preview the data without downloading it, which can be useful to know especially for the biggest sets.

Finally, there is a dedicated option for users to request specific datasets that they may need for their projects if they can not found them in the given datasets [64].

4.4. FIWARE technologies

Regarding the software technologies and protocols that FIWARE is based on, besides being open source, the standards are very tested and used in a lot of smart projects nowadays, so the documentation is easily available, and the software has been tested enough to guarantee an optimal usage of resources. In essence, the methodology when using FIWARE-related software is working with context information, which contains all the information needed to identify entities, attributes and metadata. This is done with Orion Context Broker, the core of the FIWARE project. It implements one API, NGSI (Next Generation Service Interface) to manage the data coming from the IoT devices or applications and the representation of entities and results from the data collected, stored in JSON format. There are also Generic Enablers (GE), pieces of software based on open source interfaces used as libraries and tools. They are used in the communication between IoT devices and the context server in the form of bridge adapters, such as an IoT agent for JSON, which acts as a bridge from MQTT to NGSI and the JSON format, but also in the publication and monetization of data, the security of the different APIs and the visualization of context information.

Figure 29. FIWARE API logo

4.4.1. Orion Context Broker

The most important and mandatory component in any FIWARE-based application or solution is the Orion Context Broker (OCB), which has a main function of managing context information, updating and checking it. Orion allows the publication of context information by entities (called context publishers) like sensors, so the context information is visible and available to other entities (called context consumers) that want to use such information to process it and obtain a result, like smartphone apps using sensor data to control the traffic. Context publishers and context consumers can be any application or other components from inside the FIWARE platform [65].

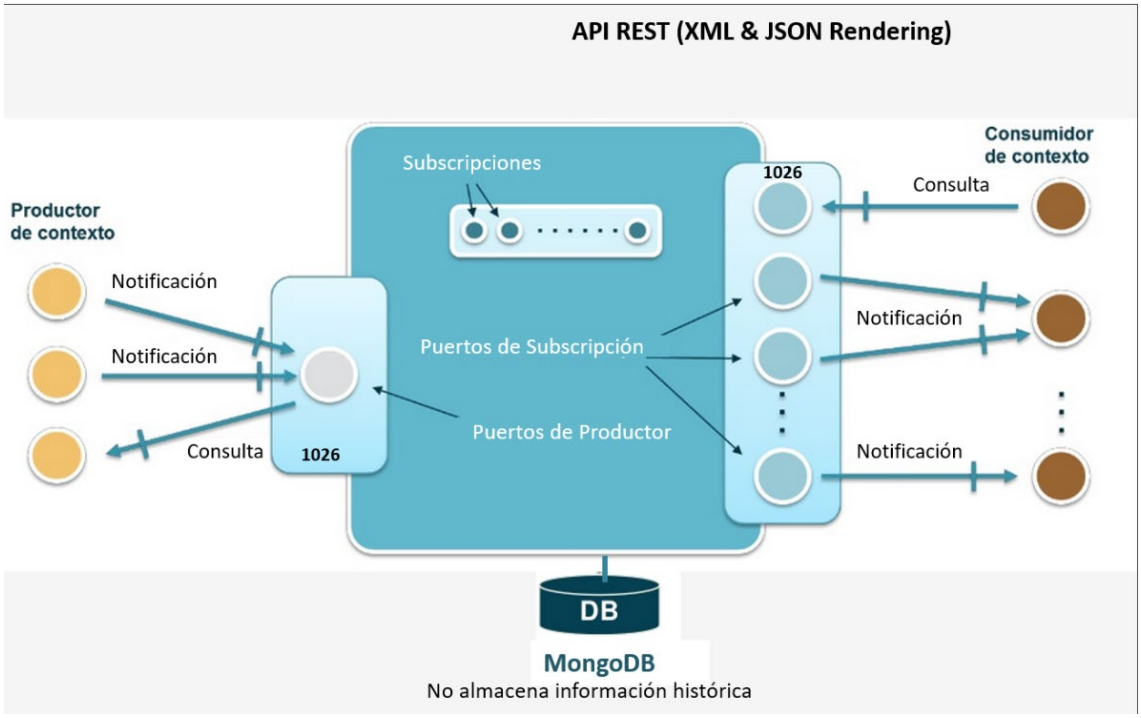

Figure 30. Diagram of the structure of a FIWARE application

Orion Context Broker is a C++ server that implements a Publisher/Subscriber Context Broker providing an interface based on the NGSIv2 REST API. Some of the operations that can be performed are:

- Register context publisher applications, like a temperature sensor inside a room.
- Update context information, for example sending temperature updates.
- Notify when there are changes in the context information (for example, when temperature changes) or with a predefined frequency (for example, update every minute).
- Consult context information.

Orion is always listening on the port 1026 by default but can be changed to any other port by the user without problems. OCB utilizes MongoDB as its database to store the actual entity status, but the historical information cannot be saved. In order to save it, an external database must be used. One of the most important characteristics of Orion is that publishers and consumers are disassociated, which means that publishers can generate context without knowing what consumers will use the information, when and where will they use it. Likewise, consumers will not know which publisher provides the information, they only care about its content.

4.4.1.1. Installing Orion Context Broker

The first step when installing Orion Context Broker is check the system requirements and compare them with the ones the user has installed. The minimum requirements recommended from FIWARE are these [66].

- System resources. It has been tested that a computer with at least 2 CPU cores and 4 GB of RAM is enough to run a server with Orion Context Broker and MongoDB as the database. It can work with less, but the main bottleneck will be the RAM because MongoDB requires a decent amount of memory to perform its operations, and the more RAM available, the faster it will work [67].

- Operating System. CentOS or RedHat. CentOS 7 is the reference system, but it will work in any later version.

- Database. MongoDB. It is required to be installed in the same host as Orion Context Broker or in a host accessible through the network. The recommended version is 3.6, it may work in later versions although it is not recommended.

- RPM dependencies. There are a few packages that must be installed in order for Orion to work properly. Some are not in the official CentOS or RedHat repositories and must be installed through EPEL[6] repositories.

After checking the requirements, it is time to install Orion Context Broker. There are two possible packages of Orion. One is the Nightly release, which are builds from the master branch every night; and the other is the Release package, which is an official release every one or two months. Nightly will always be ahead of Release packages, but it is only recommended to install the first for testing purposes.

The recommended installing method is using yum, by previously adding the FIWARE repository to the list of sources with the operation *sudo wget -P /etc/yum.repos.d/ https://nexus.lab.fiware.org/repository/raw/public/repositories/el/7/x86_64/fiware-nightly.repo*. Once the FIWARE repository has been added to yum, with the operation *sudo yum install contextBroker,* Orion Context Broker will begin its installation in the computer [68].

When Orion Context Broker is successfully installed, the first step is to set-up an authorization token for all the calls of the REST API to Orion. In order to get the token, a valid FIWARE Lab account must be created, for then to be linked with orion-gi (Orion global instance at FIWARE infrastructure) through the execution of the script token_script-sh. The script will ask for the FIWARE Lab email and password of the account, and if the process is successful, the user will be authorized to use the FIWARE Orion context Broker services [69].

---

[6] Extra Packages for Enterprise Linux (or EPEL) is a Fedora Special Interest Group that creates, maintains, and manages a high quality set of additional packages for Enterprise Linux, including, but not limited to, Red Hat Enterprise Linux (RHEL), CentOS and Scientific Linux (SL), Oracle Linux (OL) [70].

4.4.1.2. Working with Orion Context Broker

Before working with Orion Context Broker, it is necessary to start the service with the operation */etc/init.d/contextBroker start.* Once Orion is running, the user can start sending requests and working with the API. In the following example, some data from the city of Santander will be retrieved from the server, then some operations with such data will be performed. All of the different HTTP requests to the server will be explained later in the section 4.4.2 with more detail, but for now some basic operations with Orion Context Broker and the data from Santander will be done. CURL[7] will be used in all the examples, but any REST client can be used without problems, in fact, in a real-life scenario, the requests will probably be done through a programming language that implements a REST library [71].

First, in order to get the data from the FIWARE Lab server, the following operation is performed for a sound level sensor.

```
curl orion.lab.fiware.org:1026/v2/entities/urn:smartsantander:testbed:357 \
   -X GET -s -S --header 'Accept: application/json' \
   --header  "X-Auth-Token: $AUTH_TOKEN" | python -mjson.tool
```

Figure 31. Getting the data from the server using cURL

In Fig. 31, the URL followed by the curl operation is the location of the Orion Context Broker server with the information of Santander, as well as the entity that has the information followed by the sensor identifier, *urn:smartsantander:testbed:357*. The second line of the operation represents the HTTP GET request with the header that indicates which data type to accept, in this case the information will be received in JSON format, so the type is *application/json*. The next header is the authorization token of the user acquired in the previous section from orion-gi, given stored in the *AUTH_TOKEN* terminal variable. It must be introduced in the request or else Orion will not give the user the information requested. The last option is the *python -mjson.tool* which serves as a way to represent the JSON information received in the terminal for the user to read it properly [72]. With this operation, the result will be a JSON document that includes information of the measure time, sound level, sensor battery charge and the location of the sensor.

In addition to retrieve data from the server, Orion Context Broker allows the creation of entities and their modification. It is important to know that the creation of entities must be done using a unique ID for the entity, because if the ID is already chosen, the information exist and was created by another user, as the Orion Context Broker instance is shared with the rest of users. A random number can be generated and chosen for the ID. The next figure represents how to create an entity with Orion Context Broker.

[7] CURL is an open-source software used in command lines or scripts to transfer data through the network [73].

```
curl orion.lab.fiware.org:1026/v2/entities -X POST -s -S \
   --header 'Content-Type: application/json' \
   --header "X-Auth-Token: $AUTH_TOKEN" -d @- <<EOF
{
  "id": "$ID",
  "type": "User",
  "city_location": {
    "value": "Madrid",
    "type": "City"
  },
  "temperature": {
    "value": 23.8,
    "type": "Number"
  }
}
EOF
```

Figure 32. Creating an entity in Orion Context Broker

In Fig. 32 the first line represents the HTTP POST operation requested to the *orion.lab.fiware.org* server that contains Orion Context Broker in the port 1026. The headers in the creation are first the type of data that will be introduced, in JSON format; and the authorization token of the user. Next is the body of the entity to be created. The *$ID* is the previously randomly generated unique number to identify the entity. The entity will have the type *"User"* and the attributes *city_location* and *temperature*. Each attribute has their own value and type field, with a String or a float as their value type. Once created, to confirm the creation of the entity, an HTTP GET request can be queried to the server. Moreover, the values introduced can be modified with an HTTP PUT operation, for example, to change the value of the temperature, the next operation is the most compact way of changing it.

```
curl orion.lab.fiware.org:1026/v2/entities/$ID/attrs/temperature/value \
   -X PUT -s -S --header  'Content-Type: text/plain' \
   --header "X-Auth-Token: $AUTH_TOKEN" -d 18.4
```

Figure 33. Modifying the value of an attribute

4.4.2. NGSI API

NGSI (Next Generation Service Interface) is a protocol developed by OMA to manage Context Information. It provides operations like the management of the Context Information about Context Entities, for example the lifetime and quality of information; and the access (query, subscribe/notify) to the available Context Information about Context Entities. The FIWARE version of the OMA NGSI interface is a RESTful API via HTTP. Its purpose is to exchange context information. The two main interaction types are one-time queries for context information and subscriptions for context information updates (and the corresponding notifications) [74].

Communication between components of the high-level Orion Context Broker architecture is done through the NGSIv2 RESTful API, so the context information when working with FIWARE will

be presented as generic data structures called context elements. A context element refers to the information that is produced, gathered or observed and can be relevant for further process, analysis and extraction of new knowledge. It has associated a defined value, which is a sequence of one or more triplets with attributes of a context element. FIWARE supports the basic data types, as well as the possibility of defining structured data types, maps and vectors, which elements can be other basic data types, maps or vectors. A context element gives useful information to a particular entity, whether it is a physical component or part of an application. Normally, a context element contains an id (EntityId) and a type (EntityType) that identify an entity from others, with the possibility of having metadata stored, linked to the attributes of that context element. To sum up, context information in a NGSI model is represented through data structures called ContextElements that have linked the following data:

- A unique EntityId and EntityType that identify the entity to which the data references.
- A sequence of one or more attributes of data elements (triplets).
- Optional metadata linked to the attributes (triplets).

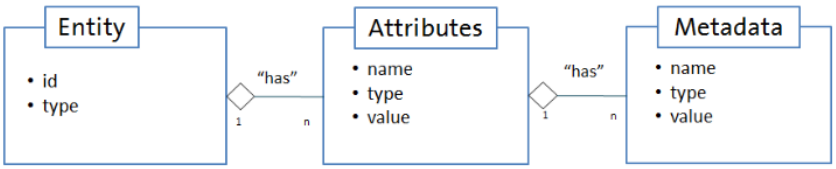

Figure 34. NGSI API diagram.

Finally, the FIWARE implementation of the NGSI model defines a data model for context information, based on a simple information model using context entities; a context data interface for the exchange of information through consult, subscription and update operations and an interface of context availability about how to obtain context information.

4.4.2.1. Representation of a JSON entity

The payload response of the API is based on application/json and text/plain for the operation types, so if the request is different from these types, a 406 error not acceptable will be received. The entity identifier is specified with the "id" property of the object, which value is a string with such identifier. Likewise, the entity type is specified with the "type" property of the object, which value is a string with such identifier.

```
{
  "id": "entityID",
  "type": "entityType",
  "attr_1": <val_1>,
  "attr_1": <val_1>,
}
```

Figure 35. Entity representation in the FIWARE NGSI API

Regarding entity attributes, they are specified with additional properties. Name of the attribute is represented as a string in the "name" field, the value of the attribute is contained in the field "value" as a string, the type of the attribute is in the "type" property as a string, and the metadata attribute is in the "metadata" field as a JSON object that contains one property for each metadata element. As a general rule, it is not allowed to use id and type as attribute names.

```
{
  "value": <...>,
  "type": <...>,
  "metadata": <...>
}
```

Figure 36. Attribute representation in the FIWARE NGSI API

4.4.3. Working with the FIWARE NGSIv2 API

In order to send HTTP requests, any RESTful client can be used, but it is required to specify the following information:

- HTTP method, which can be GET, POST, PUT, DELETE, OPTIONS, HEAD, TRACE or CONNECT.

- URL with the IP of the Orion Context Broker and the operation to perform, for example: *http://<IP_ADDRESS>:1026/v2/entities*

- Headers with the type of content and the types to accept.

```
Content-Type: "application/json"
Accept: application/json
```

Figure 37. Headers in a HTTP request

- Body of the request.

Next, an example of the operations that can be performed with the FIWARE NGSIv2 API will be shown. Assuming there is a building with two rooms named room1 and room2, each one with two sensors, temperature and pressure, Orion Context Broker will be used to manage the context information.

The first step is to create the entities and their attributes, done with the *POST /v2/entities* operation. For example, the following code represents the way of creating the entity room1 and setting the temperature and pressure values.

```
POST <cb_host>:1026/v2/entities
Header: Content-Type: "application/json"

{
  "id": "Room1",
  "type": "Room",
  "temperature": {
    "value": 23,
    "type": "Float"
  },
  "pressure": {
    "value": 720,
    "type": "Integer"
  }
}
```

Figure 38. Code to create the room1 entity and set its attributes

In the code of Figure 38, the first line represents the HTTP method to send to the server, in this case POST; followed by the URL of the Context Broker host, which will be different depending on the machine of the user; the port, by default 1026; and the type of operation, entities in this case because it is an entity creation operation. Next, the header of the message specifies the type of content it is being sent, application/json. Then, the body of the request contains the values and context information to add to the server in JSON format, in this example, the id of the entity is "Room1" and the type of entity is "Room". Finally, the attributes are added as well in the request, each one with their own value and type of value[8].

4.4.3.1. Get context information operations

Next, the entities created can be consulted and their information retrieved in different ways with a HTTP GET request to the Orion Context Broker server. The first option is to request access to the context information of all the entities stored, done with the operation *GET /v2/entities/.* However, there are other options to get only a selected entity, the operation *GET /v2/entities/{id}* retrieves the context information of an entity named {id}, in this example {id} would be Room1; and if there are more than one entity with the same name (there can be multiple entities with the same name but the type must be different), the operation *GET /v2/entities/{id}?type={type}* gets the context information of the entity named {id} with type {type}, in the example {id} would be Room1 and {type} Room. These requests will return the entity with all the attributes and details in JSON format, as it was previously defined that way.

[8] Besides the basic data types, Orion Context Broker allows for complex data structures such as vectors or key-value maps, retrieved at query time [75].

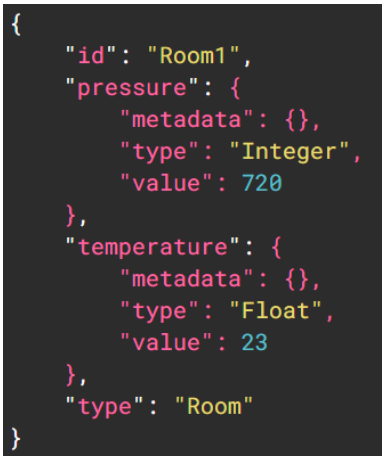

Figure 39. Result of the GET entity operation

In addition, the information presented in Figure 39 can be reduced and represented in a more concise way. The operation *GET /v2/entities/{id}?options=keyValues* will give the next result when applied to the {id} Room1.

Figure 40. Reduced GET entity operation

There is also the possibility of getting only the value of the attributes created with the entity with the operation *GET /v2/entities/{id}?options=values&attrs={attr_name1},{attr_name2},…,* in the example {id} would be Room1, {attr_name1} can be one of the two attributes, temperature or pressure, and {attr_name2} would be the other attribute. The order of the attributes will determine the order the attributes will be presented in the answer.

Other option available is to present all the context information of an attribute, done with the operation *GET /v2/entities/{id}/attrs/{attr_name},* where {id} would be Room1 and {attr_name1} the name of the attribute to get the information of, temperature for example.

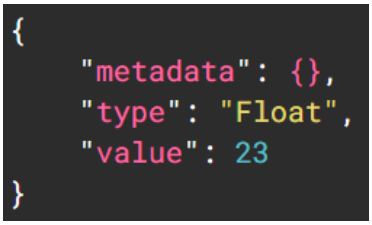

Figure 41. GET attribute operation

As done previously with the entities, it is possible to request only the value of the attribute. However, it is necessary to specify a header with the type of the value returned, normally it would be *text/plain*. The full operation for the attribute temperature of Room1 is:

```
Accept: text/plain

GET <cb_host>:1026/v2/entities/Room1/attrs/temperature/value
```

Figure 42. GET value of attribute operation

Finally, the FIWARE NGSIv2 API will show error messages if the operation cannot be completed. For example, if the entity requested has not been created, an HTTP error 404 not found will be presented to the user. The same error will be shown if instead of an entity requested it is a non-existent attribute.

```
{
    "description": "The requested entity has not been found. Check type and id",
    "error": "NotFound"
}
```

Figure 43. HTTP 404 Not Found error

Sidenote: Orion Context Broker allows for a representation of up to twenty elements in a single GET operation. Nonetheless, this can be extended until a maximum of a thousand elements, and the user can decide how many to show adding a modifier at the end of the GET operation, *?limit={n}* where n is a number from zero to one thousand [76].

4.4.3.2. Modify content operations

In the FIWARE NGSIv2 API it is possible to modify the content of the information created previously. Following the Room example, the value of the attributes can be modified with the operation PATCH /v2/entities/{id}/attrs followed by the body of the attributes, the values updated and even the type of the attribute. In this example the following image represents the operation of changing the values of the temperature and pressure of Room1 and the type of pressure to float.

```
PATCH <cb_host>:1026/v2/entities/Room1/attrs

{
  "temperature": {
    "value": 26.5,
    "type": "Float"
  },
  "pressure": {
    "value": 763,
    "type": "Float"
  }
}
```

Figure 44. HTTP PATCH operation

However, if the value of the attribute is the only field that wants to be changed, the PUT operation is much more concise and serves as a way to update an attribute without changing the type but also to replace all the attributes of a given entity. The following figure represents the PUT operation to update the temperature value of Room1. It is necessary to specify the header with the content of the value as with the GET operation.

```
Content-Type: "text/plain"

PUT <cb_host>:1026/v2/entities/Room1/attrs/temperature/value

28.4
```

Figure 45. HTTP PUT operation

4.4.3.3. Subscription operations

Operations up until now have interacted with the context information in a synchronous way, but one of the most powerful traits of Orion Context Broker is working with information asynchronously, subscribing to the context information so that when it changes, the application receives an asynchronous notification. This means that there is no need to constantly send HTTP requests to the server, Orion Context Broker will send the context information when it is available[9].

The operation to create a subscription is *POST /v2/subscriptions.* A header must be specified to set up the information about the accumulator server. In the next figure, it is shown how to create a basic subscription for the temperature following the Room example.

[9] Orion Context Broker must be configured as an accumulator server to be able to answer asynchronously, with the URL required when configuring the subscription, for example the server can be in http://localhost:1028/accumulate.

```
Content-Type: application/json

POST <cb_host>:1026//v2/subscriptions

{
  "description": "A subscription to get info about Room1",
  "subject": {
    "entities": [
      {
        "id": "Room1",
        "type": "Room"
      }
    ],
    "condition": {
      "attrs": [
        "pressure"
      ]
    }
  },
  "notification": {
    "http": {
      "url": "http://localhost:1028/accumulate"
    },
    "attrs": [
      "temperature"
    ]
  },
  "expires": "2040-01-01T14:00:00.00Z",
  "throttling": 5
}
```

Figure 46. HTTP POST operation

In Figure 46, the first line is the header, indicating that the content of the HTTP POST operation will be in JSON format. Next, the proper POST operation, where <cb_host>:1026 is the IP of the Orion Context Broker followed by the port where it is listening for requests. Then, the body of the operation contains a few fields:

- First, a description of the operation. It can be whatever the user wants to write, it is only for documentation purposes.

- Second, the entity where the operation will apply. It is a vector field, so it can apply to more than one entity at the same time, but they must be differentiated with the *id* and *type* fields.

- Third, the condition for the notification to be sent. When the attribute changes its value, it will act as a trigger to update the value of the attribute specified in the notification section. In this example, when the pressure changes, the temperature will be notified. It may seem useless in this example, but it is just to show how powerful the subscription function can be. It can be left empty as well.

- Next, the notification and its configuration. The first thing to set is the URL where the accumulator server is configurated. There can only be one URL per subscription, but there can be more than one subscription and URL with the same context elements. The attrs field indicates which attribute[10] to notify when the trigger is activated, in this case temperature. It is not present in the example, but an entity can also be notified the same way as an attribute, but in an entity field.

- The last two fields are modifiers of the notification system. Expires field indicates when will Orion Context Broker stop sending updates of the attribute or entity. It must be given using the ISO 8601 standard for dates. Throttling field stablishes the minimum time that must pass between one notification and the next one, even if there is a change in value in the meantime. In the example, if a notification is sent within five seconds of the previous update, it will not be sent no matter how many changes of value have happened in that period. Both these fields can be left empty.

The result of the operation is an HTTP code 201 Created, as well as a header with the subscription ID, a twenty-four hexadecimal character string used to identify the subscription among the rest, and it is used to update or cancel the subscription.

Finally, the subscription can be paused for later, done with the PATCH operation and modifying the status field from active to inactive.

```
{
PATCH <cb_host>:1026/v2/subscriptions/57458eb60962ef754e7c0998
  "status": "inactive"
}
```

Figure 47. Pause a subscription

In Figure 47, the <cb_host>:1026 is the IP of the Orion Context Broker server and its port, and the string of hexadecimal characters at the end of the operation is the ID of the subscription that wants to be modified.

4.4.4. Generic Enablers

In addition to the whole FIWARE NGSIv2 API and Orion Context Broker, FIWARE offers the possibility of using some specific pieces of software called Generic Enablers. A Generic Enabler (GE) is an independent open-source component that can be assembled with other Generic Enablers or third-party platform components to accelerate the development of Smart Solutions, offering reusable and common shared functions with multiple use cases that serve in various sectors [77]. Generic Enablers act in four different scenarios built around the FIWARE Context Broker, the manager of the context information and the cornerstone of any basic “Powered by FIWARE” solution. Those four groups are Core Context Manager, Context Processing, Interface

[10] Because the field is a vector type, more than one attribute can be specified for the notification.

to IoT devices and API Management, each one with their own set of Generic Enablers developed specifically to address the issues present in such groups [78].

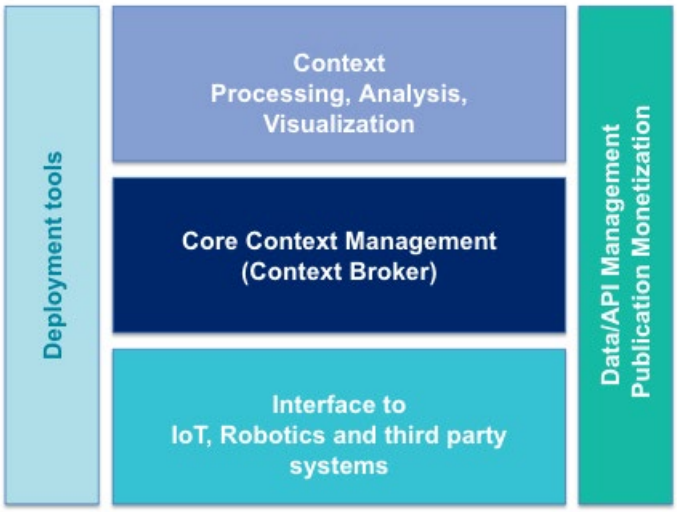

Figure 48. Classification of FIWARE Generic Enablers

4.4.4.1. Core Context Management

The first group of Generic Enablers belong in the Core Context Management group. A context Broker Generic Enabler is the mandatory component in any FIWARE-based solution. It enables to manage the context information in a decentralized way. The most important Generic Enabler in this group is Orion Context Broker, which provides the FIWARE NGSI v2 API, a RESTful API that allows updates, queries or subscriptions to changes on context information, previously explained thoroughly.

In this group there are also under incubation a few Context Brokers that support the newest ETSI NGSI-LD API[11] specifications, and those Generic Enablers are Orion-LD Context Broker, a NGSI-LD Broker that supports both the NGSI-LD and the NGSIv2 APIs; Scorpio Broker and Stellio Context Broker, two alternatives to Orion-LD.

[11] ETSI NGSI-LD is a Context Information Manager API that enables close to real-time access to information coming from many different sources, based on Linked Data and the NGSI standard and developed by the company ETSI [79].

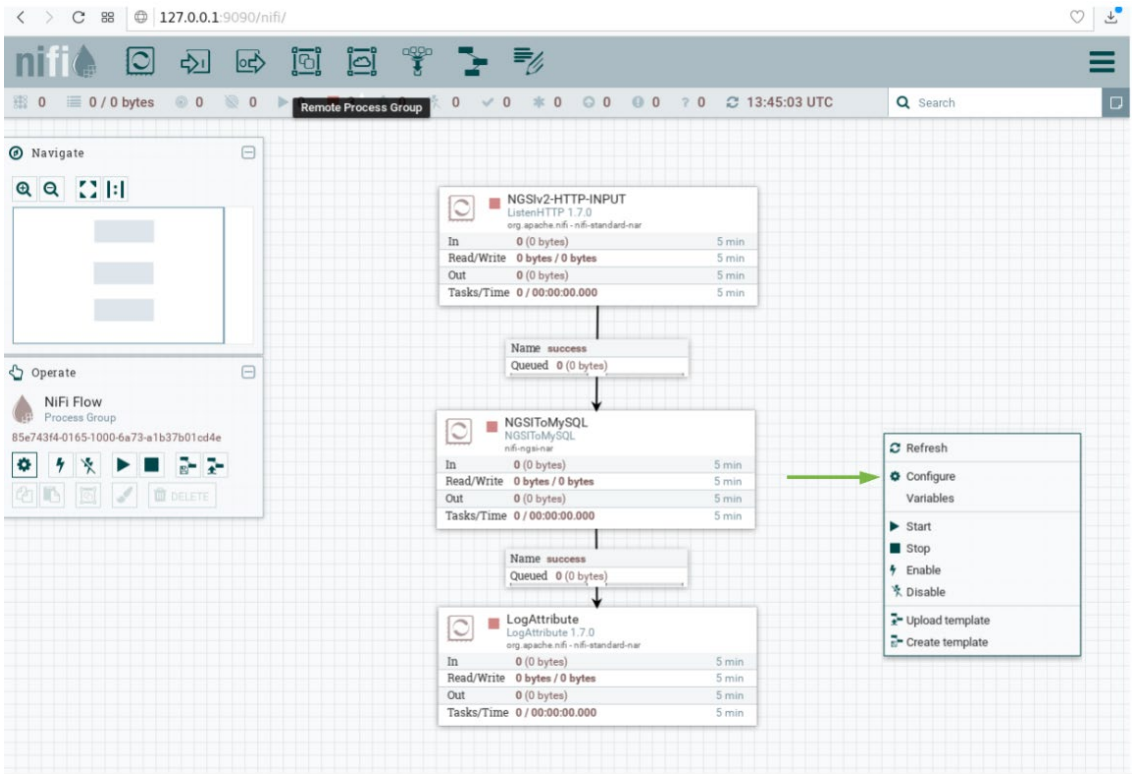

Figure 49. Draco Generic Enabler interface [80]

Some others Generic Enablers that accompany the Context Broker are Cygnus and Draco. Cygnus is a GE that gives means for managing the history of context created as a stream of data that can be interpreted by most popular database management systems like PostgreSQL, MySQL, MongoDB or AWS DynamoDB as well as Big Data platforms like Hadoop or Storm. Draco GE (Figure 49) gives an alternative data persistence mechanism to manage the history of context information based on Apache NiFi[12] and the concepts of flow programming, supporting scalable directed graphs of data routing and offering an intuitive graphical interface.

4.4.4.2. Context Processing, Analysis and Visualization

In this group, the Generic Enablers are focused on providing methods to process, analyze or visualize context information in an easier and simpler way to have a better scope of the application being developed and check for possible design flaws. The following Generic Enablers are the core of this group and contain all the features to represent information and visualize it in a simple manner.

Wirecloud GE gives a powerful web mashup platform for users to develop highly customizable and operational dashboards to visualize information. Mashups are built as workspaces where widgets and operators can share data through data flows and are mashed up together.

[12] Apache NiFi is an easy to use, powerful, and reliable system to process and distribute data, supporting powerful and scalable directed graphs of data routing, transformation, and system mediation logic [81].

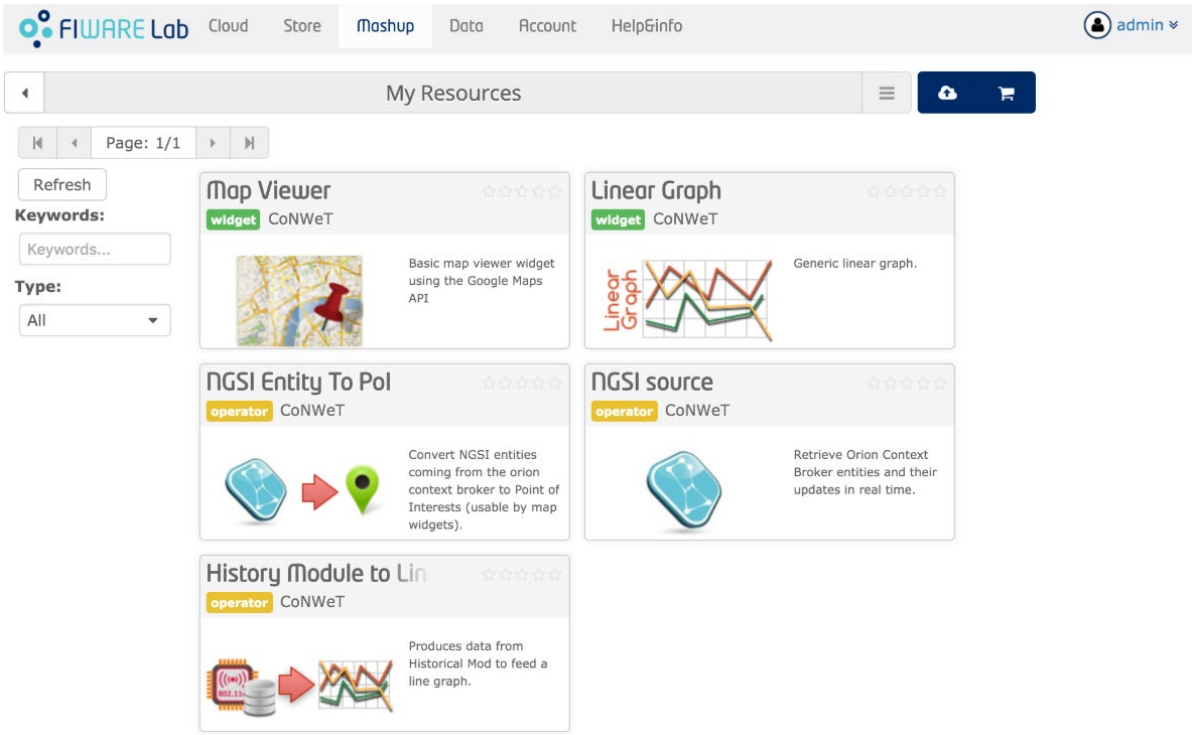

Figure 50. Wirecloud Generic Enabler interface

Knowage Generic Enabler is a powerful Business Intelligence platform enabling to perform business analytics over traditional sources and Big Data systems built on context history. It has a few sub-products each one related to a specific field of knowledge related to data and its different interpretations through multiple analysis, from the newest Big Data modules with clusters and NoSQL databases to modern Predictive Analysis, with other modules such as Location Intelligence to relate the data to the geographical location, Performance Management to manage KPIs, Enterprise Reporting to produce static reports and the usual Smart Intelligence business solution for structured data and agile methodologies [82].

Kurento Generic Enabler provides real-time processing of media streams supporting the transformation of video cameras into sensors as well as the incorporation of advanced application functions such as integrated audiovisual communications, augmented reality or flexible media playing and recording. Its core element is a Media Server, responsible for media transmission, processing, loading and recording, providing networked streaming protocols like HTTP and RTP, and automatic media transcodification with many codecs supported such as VP8 or H.264 [83].

Lastly, there are some Generic Enablers that are in an incubated state, they are being developed and some functionalities may not work properly. These Generic Enablers are Perseo, a Complex Event Processing software compliant with the NGSIv2 API designed to listen to context information to identify patterns and rules [84]; and OpenVidu, a Generic Enabler that provides an abstraction layer for Kurento, making media processing easier to program.

4.4.4.3. Interface with IoT, Robots and Third-party systems

Generic Enablers in this group provide an easier interface with the Internet of Things, Robots and Third-party systems for the purpose of gathering valuable context information or trigger actuations in response to context updates. The core of this group is the IDAS Generic Enabler, an implementation of a Backend Device Management software that offers a wide range of IoT Agents making it easier to interface with devices using the most widely used IoT protocols. The protocols that IDAS supports are the following.

- IoT Agent for JSON. A bridge between HTTP/MQTT messaging (with a JSON payload) and NGSI.
- IoT Agent for LWM2M. A bridge between the Lightweight M2M protocol and NGSI.
- IoT Agent for Ultralight. A bridge between HTTP/MQTT messaging and NGSI.
- IoT Agent for LoRaWAN. A bridge between the LoRaWAN protocol and NGSI.
- IoT Agent for OPC-UA. A bridge between the OPC Unified Architecture[13] protocol and NGSI.
- IoT Agent for Sigfox. A bridge between the Sigfox protocol and NGSI.
- IoT Agent library. A library for developing your own IoT Agent.

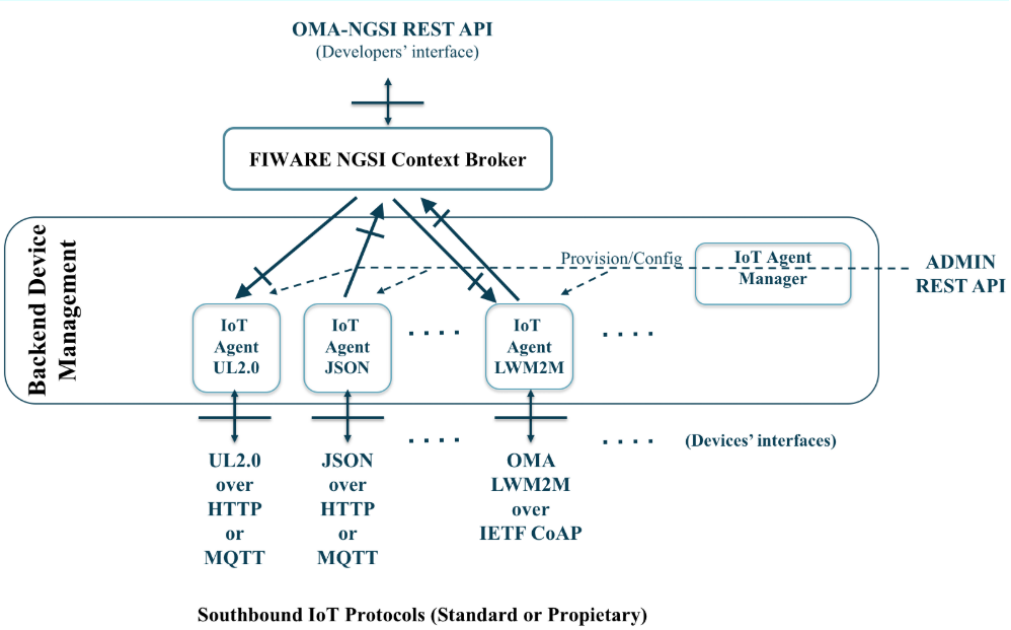

Figure 51. IoT Agents of IDAS Generic Enabler [85]

[13] Open Platform Communications Unified Architecture is a platform independent service-oriented architecture that integrates all the functionality of the individual OPC Classic specifications into one extensible framework. OPC Classic is the interoperability standard for the secure and reliable exchange of data in the industrial automation space and in other industries [86].

The next Generic Enablers are all in an incubated state, with reduced functionalities as they are still in development. For connections with robotic system, Fast DDS Generic Enabler has been adopted as the default middleware in ROS2[14], therefore it helps to interface with ROS2-based robotics systems; Micro XRCE-DDS Generic Enabler, a lite version of the previous GE adapted to run in microdevices; and FIROS Generic Enabler, which works transforming ROS messages into NGSIv2 and vice versa. Finally, Domibus Generic Enabler allows the users to exchange information and documents with one another in a secure way.

4.4.4.4. Context Data, API Management, Publication and Monetization

Generic Enablers in this section are divided in two groups: the GE of the first group are focused on granting secure access to the different APIs in any FIWARE solution, and the ones in the second group are focused on the publication and monetization of context data resources that the Orion Context Broker of the user application produces.

Regarding the first group, there are two main Generic Enablers. The first one is Keyrock Identity Management, a software that provides secure and private OAuth2-based[15] authentication of users and devices, user profile management and Single Sign-On (SSO). The other security Generic Enabler is Wilma Proxy, which gives support to proxy functions within OAuth2-based authentication applications, with the possibility of combining it with other security components such as Keyrock [87].

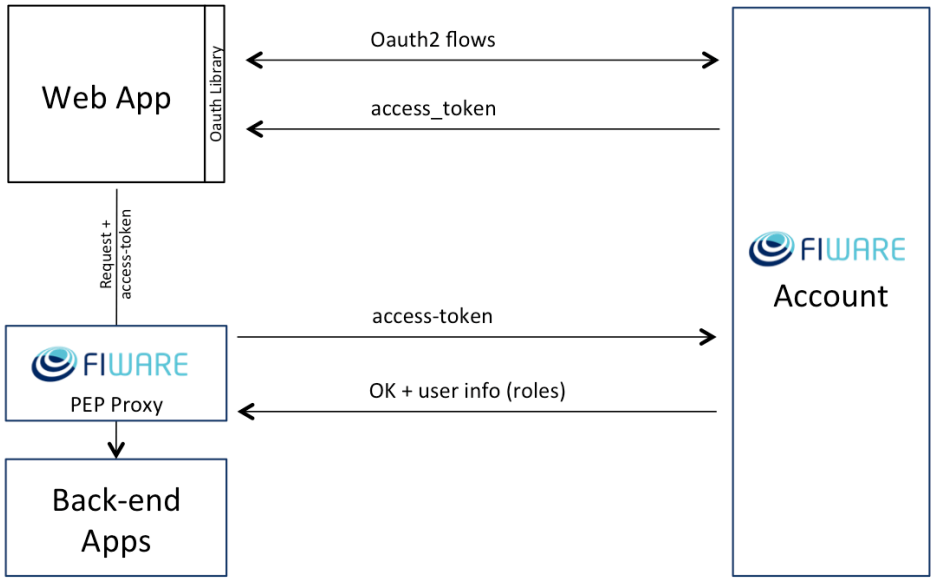

Figure 52. Wilma Proxy Generic Enabler and the authentication process

[14] Robotic Operating System is a set of software libraries and tools for building robot applications from basic drivers to state-of-the-art algorithms [88].
[15] OAuth2 is a protocol that enables a third-party application to obtain limited access to an HTTP service by orchestrating an approval interaction between the resource owner and the HTTP service or by allowing the third-party application to obtain access on its own behalf [89].

In the second group, there is one Generic Enabler for context data publication and monetization, CKAN Extensions, which gives some add-ons to extend the capability of CKAN Data publication platform, for example allowing the publication of datasets matching right-time context data, the assignment of access terms and policies to those datasets and the assignment of pricing to datasets.

Besides the main groups of Generic Enablers, there are two more that are currently in an incubated state. The first one, Idra, is able to federate existing Open Data Management Systems based on heterogeneous technologies (e.g. CKAN, SOCRATA, DKAN etc.) providing a single API and a standard metadata format to discover open datasets. The second one, API Management Framework, is a tool for API owners to manage their APIs, providing all the necessary features to run business with APIs.

4.5. FIWARE Marketplace

FIWARE solutions can be browsed by anyone in the FIWARE Marketplace. It serves the purpose of globally disseminating existing commercial offerings around FIWARE. It is a global shop that gives visibility to a wide range of "Powered by FIWARE" solutions, FIWARE-ready technologies as well as FIWARE-related training, integration and support services [90]. In the marketplace, besides external users looking for solutions or IoT devices among other things, business or startups can choose to publish their own offerings.

Figure 53. FIWARE Marketplace logo

4.5.1. Powered by FIWARE solutions

In the Powered by FIWARE section, solutions and platforms can be found. A FIWARE Smart Solution is based on FIWARE standards to manage context information at large scale. They use FIWARE technologies to gather Context Information (coming from different and highly distributed sources such as end users, sensor networks and all kind of information systems including social networks) and process it in order to perform smart actions [91]. Some examples of FIWARE-based solutions related to smart cities are the following.

- Línea Ciudadana. Platform for management, communication and citizen participation, as well as a self-managing mobile platform. It brings the administration and city governance closer to the citizens, offering information and supporting leadership and democracy in

the city. Some functionalities are bidirectional communication with the government and the citizens, information for tourists, reports and statistics of the city and a notification service for alerts. It is already working in some Spanish cities like Yecla, Puertollano or Torrelodones [92].

- Stratio Automotive. Product for in-vehicle data collection in real-time and a software product for Artificial Intelligence analysis over vehicle data, which allows to identify and predict engineering problems and malfunctions. The proprietary algorithms work with the vehicle sensor data, and extract predictions, measuring the performance of the internal components and detecting faults before they happen [93].

Besides FIWARE-based solutions, there is also the option of having a FIWARE Platform service, which is a much bigger solution and can be fit for more situations. FIWARE Platform Service Providers offer FIWARE as a Service on private and public clouds in which they operate with the possibility of having instances on premises for their customers. Anyone can create a FIWARE Platform instance for personal use or in their premises, but it is also possible to choose a company that offers FIWARE Platform services [94]. Some examples of FIWARE Platform services are the following.

- Orchestra cities. Platform designed to facilitate the management of data and services by different cities and stakeholders, analyze and visualize data of a Smart City from sensors, citizens and more. Orchestra Cities offers flexible dashboard co-creation, including map, graphs, and easy sharing; support for centralized management of IoT allowing the integration of live data from sensors, devices and other sources; and advanced data privacy control mechanisms, ensuring that the data owner is always in control of the data. It is in use in cities like Helsinki, Wolfsburg or Antwerp [95].

- Helix Sandbox. Platform fully compatible with FIWARE Generic Enablers (GE). Its purpose is to simplify the process of installation, configuration and use of the GEs through an easy-to-use graphical interface that allows to control all the elements in it. It uses Docker to instantiate the GEs, so it can be installed on any Cloud platform or virtualization agent. It is used in the Brazilian IoT Forum and USP (University of Sao Paulo) [96].

4.5.2. FIWARE-ready technologies

In this section, the products offered are segmented in three categories. First category is FIWARE-ready IoT devices. These devices come with easy-to-install drivers and instructions to help transform the data they gather from the outside into context information that is compatible with the FIWARE NGSIv2 standard [97]. Some devices are the following.

- Night Eye Sky Sensor. This device is able to measure the brightness background of the sky in different spectral bands and it is especially useful to evaluate light pollution. Through its interconnectivity, users can consult the data in real time, allowing to

download and visualize the data in the cloud, as well as generating a network of measures to control the pollution [98]. Some users are the University of Málaga and Junta de Andalucía [99].

Figure 54. Night Eye Sky Sensor physical device

- Smart POI. Point of Interest (POI) are strategic areas consisting of a set of Smart Spots devices that send a URL and create a physical space of information where everyone approaching can collaborate through a smart phone. Therefore, Smart POIs connect physical objects or places with the smart phone to offer an interactive and multimedia experience. This technology allows to directly open a responsive Web App that contains information designed to answer a specific topic, including text, videos, images and any multimedia material. Some cities that use this device are Mexico DF, Aarhus in Denmark and Ceutí in Spain [100]. Specifications appear with detail in the next figure.

Figure 55. Smart Spot device and specifications [101].

The second category is FIWARE-ready software enablers. They are base platform technologies, easily integrable with FIWARE, which extend the basic capabilities of the platform with advanced features such as location within buildings, representation of context information in maps and biometrics for access control. Their combination with FIWARE allows to build more sophisticated applications [102]. Some examples are the following.

- FI-ASSIST. System created to introduce a high-precision indoor navigation system and location infrastructure for restaurants based on acoustic localization. FI-ASSIST consists of an application for mobile phones, mobile acoustic senders and of the acoustic receivers installed in the building. The localization system enables to find the customer's position quickly and reliably, reducing the restaurant's staff overhead and workload [103].

- siGEUS. Platform composed of hardware modules to collect data from the environment mainly in waste management and software to manage the information. In waste trucks GPS devices track the route of the vehicles, RFID readers identify the containers picked up and the weight systems know how much waste has been loaded. Also, a filling level sensor on the containers is installed to know the percentage of waste they have. Then processes to optimize waste management are applied, such as route optimizations, predictive analysis or business intelligence. Some customers are Volvo, Still, Caterpillar or Ecoembes [104].

The last category is Building Blocks. A Building Block is an open and reusable digital solution that can take the shape of a framework, a standard, a software, a software as a service (SaaS), or any combination of them. Building Blocks are endorsed by the European Commission and ensure the digital service will be fully compatible with other on the market. Examples in this category are the Context Broker integrations with other technologies such as the European Data Portal or historical data management with the Generic Enabler Cygnus [105].

4.5.3. FIWARE-related services

In this group, the services offered can be FIWARE-related training and coaching for example from a technological point of view or more business oriented. Online training material is available on the FIWARE Academy, but many partners are offering training and coaching services to those organizations who are looking for dedicated sessions [106]. The best example is the FIWARE Zone iHub in Seville or Málaga, which works on promoting and spreading FIWARE technology around the region, delivering several courses and helping companies to develop and integrate solutions to make the ecosystem growth [107].

The other possibility of services is a business-oriented service, with experts that offer consultancy, integration or technical support for any future or actual project [108]. There are consultancy services in almost every country where FIWARE has an iHub, as one requirement of an iHub is providing consultancy and training services. However, some partners and startups offer consultancy. The best examples and the ones with wider presence are the following.

- FICODES. A startup of engineers and developers with deep experience with the FIWARE platform and API management, as this startup developed and maintained a few Generic Enablers for more than five years. Some customers are the city of Málaga and the organization Capelon in Sweden [109].

- StoneOne. It is the provider of the open and standardized S1 IoT platform, but also provides accompanying services such as the complete development of a customer or industry specific IoT platform or marketplace. StoneOne has in-depth knowledge of the FIWARE platform and supports companies and public institutions in providing IoT services [110].

4.6. Success stories

FIWARE as a platform has had many different applications developed from startups to SMEs and they have been deployed in a real environment with great success. Some of these applications will be detailed next to show that the FIWARE Orion Context Broker and the context information management is very useful when dealing with IoT devices and the development of Smart Solutions to convert cities into Smart Cities. Most of the following examples are available on the FIWARE Marketplace with information about the developers, so anyone can contact them for any inquiry.

4.6.1. Breeze Technologies

The first success story is about the German startup Breeze Technologies. The main objective was to use environmental data intelligently to make cities, buildings and communities more livable for citizens now and in the future with data obtained from the air pollution, as it is one of the biggest environmental health threats. There is also the problem of data-driven decision making, because it remains unfeasible as sensing equipment is expensive, stakeholders lack necessary knowledge to analyze the data and suitable interventions are unknown.

Figure 56. Breeze Technologies logo

To solve the problem, the solution consists of installing air quality sensors that are up to 50.000 times smaller and up to 1.000 times cheaper than traditional measurement devices, as well as offering a simple and easy-to-use UI. But there is also a factor of consultancy done for businesses, explaining the effects of the current air quality situation, predicting possible improvements using different tools and methods and providing a marketplace with vendors offering air quality improvement solutions. For cities, Breeze Technologies gives the planning and environmental departments access to high-resolution live data, offering a standardized API integrable with

standard industry solutions for its use in management systems and business intelligence software [111].

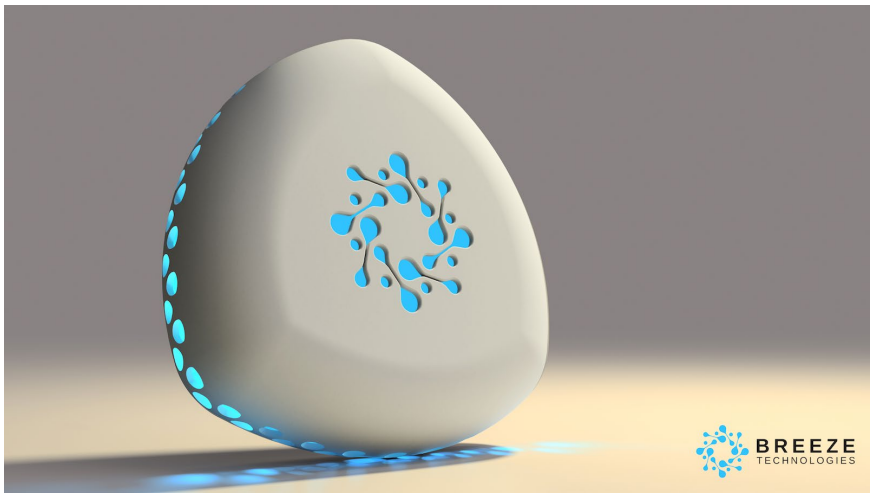

Figure 57. Breeze Technologies physical IoT device

Regarding the technology employed and the involvement of FIWARE, the Breeze Technologies analytics suite for Smart Cities is built upon several FIWARE components, some of them used in development and testing, and others in the live environment. Cloud APIs are realized with the standard NGSI using Orion Context Broker for the management of context information. In addition, the Open Data platform CKAN is used to publish and consume environmental data, and the CitySDK data models define how the system interacts with the data generated by the IoT devices in the city. When the project finished its initial development, FIWARE provided the right opportunities so that the solution could be validated in the market with a number of test users. As a result, an alpha version was developed, deployed and tested by partners of Breeze.

Breeze Technologies launched its first Smart City pilot in July 2017 in Hamburg. Breeze installed a number of urban air quality sensors and made the data available through its citizen portal so citizens can access real-time air quality information about any place they want [112].

The project received heavy support from the local community and district council, and nowadays, Breeze has become an integral part of the European Smart City community, working together with industrial partners in a number of research and development networks, as well as winning awards such as Science4Life: Idea Phase or Germany: Land of Ideas with a distinction from the German president.

4.6.2. Ubiwhere

Another success story is that of the startup Ubiwhere and their solution for Smart Cities Citibrain. The problem to solve was the management of different solutions for a single city as Smart Cities are inherently complex in nature. A single company cannot expect to monopolize every aspect of the ecosystem, so a diverse ecosystem must be expected, with parts of the data released freely, and other parts controlled by private stakeholders [113].

Figure 58. Ubiwhere logo

Ubiwhere is a Portuguese company dedicated to research and development of software for Smart Cities and the Future Internet sectors, with the actual focus being on the unified Smart City solution Citibrain, a consortium specialized in creating smart solutions for global challenges. The fields that the applications are focused on are City Parking, Waste Collection, Urban Cleaning and Urban Environment, the main challenges of municipalities in terms of Urban Mobility and Environment, and some of them will be detailed next [114].

- Smart Parking. The solution consists of a parking management solution which collects key performance indicators in real time and translates them into knowledge for better policy and optimized management. This system is also a simple way to provide citizens with the location of the available parking spaces. For cities, the solution decreases the operating costs of parking, reduces accidents and traffic congestion, while improving parking operations and reducing pollution peaks.

- Smart Air Quality. Through small IoT devices that capture air quality, noise pollution levels, temperature, atmospheric pressure, humidity and luminosity; the system can respond more effectively to the problems of citizens, improving urban planning and obtaining objective data on the quality of life provided.

- Smart Waste. The placement of sensors with low energy consumption and high durability in the traditional trash bins allows to keep a tight control on the state of the container, its location and security, increasing the effectiveness and efficiency of the waste management teams. Also, the application generates rates based on the contribution of each citizen, encouraging recycling.

Regarding the involvement of FIWARE in the development of the solutions, FIWARE technology allows to create interoperable, standards-based ways to unify all of Ubiwhere's Smart City solutions. For example, the application about transport exposes its transportation and tourism data via the NGSIv2 API, and the Smart Parking, Traffic, Waste and Environment solutions, collect their data through the Backend Device Management Generic Enabler, shared via Orion Context Broker and processed in the Complex Event Processing GE. Other Generic Enabler employed in the different applications is the Identity Management GE KeyRock, to provide an extra layer of security to the platform.

Citibrain is currently implemented by municipalities and City Service Providers worldwide. One of the cities using the Smart Air Quality solution is the city of Porto, Portugal, deploying 75 Air Quality stations in the city and monitoring the environmental conditions in real-time.

4.6.3. WiseTown

The next success story is the Italian application "Web Information Streams Enhancer for your Town" or WiseTown, focused on creating an ecosystem for the management of big chunks of data in Smart Cities. But cities are often small or medium-sized and cannot manage to sort and analyze all the information that is being produced. In order to solve that unevenness between goals and capabilities, it is important to develop a Smart Data plan as a central part of each Smart City strategy, as data is the key point to activate Civic Technologies and to make them really effective [115]. The goal of this solution is to collect information from different streams to identify the issues that affect the city in several areas: urban renewal, garbage collection, public safety, transportation, social services and environmental problems.

Figure 59. WiseTown logo.

WiseTown is not only a web application, but a whole ecosystem for web and mobile that offers crowdsourcing functionalities, applications and modules to perform mainly two tasks, create analytics of the town issues and other information in order to improve the real-time event management and future urban planning; and create a participative environment where citizens and town managers can discuss city development.

Regarding the use of FIWARE technologies in the solution, WiseTown collects information from several data sources. FIWARE Orion Context Broker connects the platform with CKAN Open Data platform, NGSI API to connect to the various IoT sensors, and Orion notifications to communicate with other APIs[16]. All the processed information goes into FIWARE Cosmos through the Cygnus connector, and supervisors of the platform will perform geographical analysis thanks to the ArcGIS Online geoprocessing capabilities and pattern analyses with the business intelligence functionalities of FIWARE Spago B-I GE.

WiseTown has been used in different cities, but the first deployment of the application was in the city of Perugia, in Italy. A specific feature of WiseTown was used, 'Crowd Planning', which allowed the local government to inform its citizens about several topics like possible improvements they would make to the city. WiseTown collected all the information from sensors, citizens, and third-party solutions, then analyze it together in order to detect potential issues and improve the quality of life of its citizens. With this information, the application created a data

[16] This communication is done through subscriptions, explained in section 4.4.3.3.

model that could be used by everyone, including private businesses for their own purposes. FIWARE helped WiseTown allowing to acquire data from different data sources and creating one single data model.

4.6.4. SaMMY

The final success story is the Greek application SaMMY, a smart app in the world of yachting marinas, aspiring to be the ultimate travel companion for yachters, skippers and marina administrators. Yachting nowadays is confusing, and many customers, from yacht owners to berth managers do not have a system where consulting prices, fees and yacht or berth availability is all in one place with easy access. The goal is to provide a centralized application with support for e-booking services, navigational and parking assistance features and mobile guidance to the nearby coastal areas [116].

Figure 60. SaMMY logo

SaMMY is an application with technologies that transform a regular Marina into a Smart one. The IoT sensors present in the platform include a multi-purpose sensor grid consisted of waterproof wireless sensors that communicate with the central system and provide booking and meteorological environmental information. These services are facilitated through the installation of different types of sensors: ultrasound, meteorological, water quality, wave measurement and more, which collect data and monitor the conditions within the Marina in real-time [117].

Regarding the FIWARE involvement and use of the technology, Orion Context Broker is configured to manage Context Information for the mobile application, Yachter's mobile app, and the Marina Management System, the web app. The mobile app interacts with the Context Broker through AJAX requests in order to publish the IoT sensor datasets. Additionally, a Complex Event FIWARE Generic Enabler is also configured for handling specific complex events, for example the berth space availability. In addition, the User Interface components interact with the FIWARE Generic Enablers via RESTful services, used both for the interaction with the Marina Management System and the Yachter's mobile app. Finally, the Wire-Cloud GE was employed to provide a working space for the marina officers and the system administrators integrating the data, applying business logic and creating the necessary UI components [118].

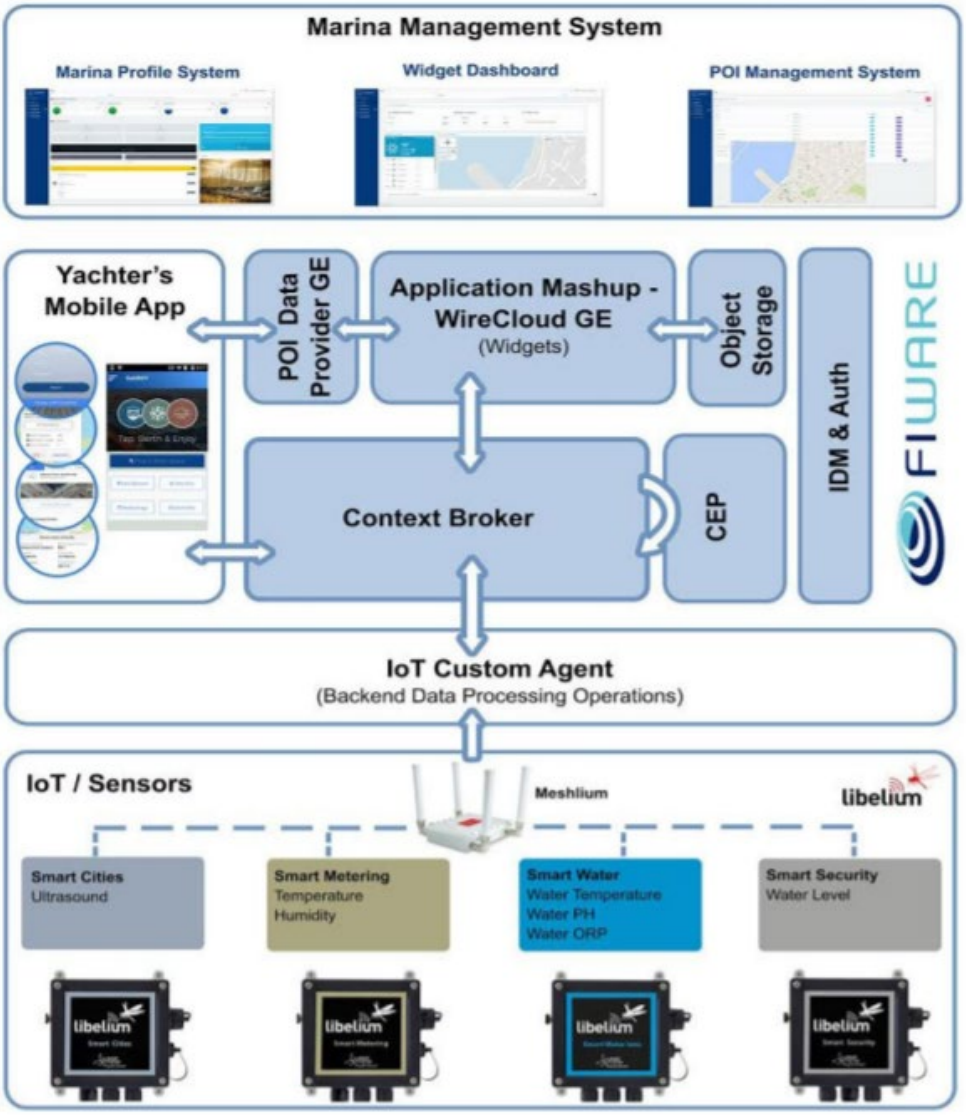

Figure 61. Diagram of all the components forming the SaMMY application

FIWARE has also helped SaMMY not only on the technological field providing an open platform, but creating a community of corporates, startups and experts around the application that work together with the developers and support the platform with new solutions. The accelerator frontierCities have supported the SaMMY application through coaching and targeted mentoring services, along with the funding of the program. Customers of SaMMY include more than fifteen marinas in Greece and Cyprus and is currently at the expansion stage reaching countries like Italy and France.

5. Conclusions

FIWARE is an ecosystem born in Europe to advance the Internet of Things technologies and promote new and open standards. It is built upon the concept of openness, from Open Source software and frameworks to Open Data available to the general public. The FIWARE Foundation is the main organism in decision making and its members choose the course of action of the ecosystem. It is formed by enterprises and individuals, and there are different tiers depending on how much enterprises contributes.

There are four main programs in the FIWARE ecosystem, and all of them are complementary. FIWARE Mundus is the program that takes care of expanding the mission and the knowledge about FIWARE to the rest of the world, organizing events and training sessions. FIWARE Accelerator promotes the use of the technology by giving SMEs and startups funding and coaching in the FIWARE technologies and framework to develop smart solutions. FIWARE Lab is the sandbox environment where anyone can try and test the different FIWARE technologies free of charge and develop their own solutions. FIWARE iHubs are centers of application of the FIWARE technologies in a regional level, and they combine all the characteristics of the previous programs into one centralized space. With all these programs working together and the help of the FIWARE community, the ecosystem can be spread and known by the general public and promote the open standards and philosophy of the ecosystem. Without these programs, FIWARE would not have extended as much as it has, and it would be much less known than it is now, especially outside Europe. FIWARE Lab is very important as well, because it allows anyone to test the technology for free, making FIWARE even more available. Also, the choice of open source philosophy makes FIWARE well known among the open source developers and those interested in it, in addition to the datasets gathered from the different supporters available to the public. Finally, the management of context information, very important in Internet of Things systems in cities, is centralized in Orion Context Broker, which, with the support of the various Generic Enablers, make the management easier and more complete than other decentralized systems that depend on many solutions to keep track of the huge amount of data collected by the sensors.

In conclusion, FIWARE has proved itself a great platform for the development of Smart solutions in regard to Smart Cities especially with its numerous success stories from startups and SMEs like Ubiwhere or Breeze Technologies. The open source Orion Context Broker framework and NGSI API standard guarantee that it can work with other third-party solutions already established or in development by other businesses, making it an ideal tool for cities that already have projects with other solutions, as there will not be a need to modify such solutions. FIWARE is part of the Future Internet and is a tool to keep in mind for now on when building Internet of Things systems in Smart Cities, as some cities like Santander or Málaga have already proved.

6. Future Work

Future work will consist in developing a FIWARE-based application that uses Internet of Things devices to get information and the results obtained can be consulted by anyone, following the open philosophy. On section 4.6, success stories, a few examples of FIWARE-based applications where shown. All those stories serve as future work examples, because they are all related to the field of Smart Cities. However, there are much more options available regarding Smart Cities once FIWARE has been chosen as the development framework.

For example, a possible future application that can be developed is a parking management solution that notifies the users of possible parking spots or when cars leave their parking. It will be deployed in a certain area and use IoT sensors to detect if a car is parked. With all the information of the sensors, a network of parking spots can be created, then the users will access the information with their mobile phones and check if the place where they want to go has parking available or not. FIWARE Orion Context Broker will be used to manage the context information from the multiple IoT sensors deployed across the city.

There are multiple options to choose from to continue the work in the future centered around Smart Cities. The requirements will often be the same, Internet of Things devices to gather information, or get the data from other sources such as the Open Data platform CKAN; FIWARE Orion Context Broker to manage the information, with the possibility of complementing and amplifying its functionalities with Generic Enablers; and an Smartphone application to consult the information, albeit optional.

7. Cost of the project

Creating this project has costed the following in terms of the personal costs.

Work	Hours	Cost/hour (€/h)	Total cost (€)
Research	115	17	1.955,00
Analysis	50	22	1.100,00
Redaction	175	25	4.375,00
Total			**7.430,00**

Table 5 Personal costs of the project

Research work includes all the searching done for information in multiple papers, webpages and official documentation of each technology or business. Analysis work is done to separate the useful information from the useless and get only the most relevant data for the following writing work. Redaction work is about the process of elaborating the memory of the project by using the previously selected information in the analysis phase.

Next, the hardware costs.

Hardware	Units	Cost (€)
Workstation, Ryzen 7 3700X 3.6 GHz, Nvidia GTX 970 4GB, 16GB RAM DDR4, 1TB SSD + 1TB HDD	1	1.200,00
Razer keyboard	1	100,00
Razer mouse	1	40,00
ASUS Monitor 24”	1	200,00
HP Printer	1	40,00
Total	**4**	**1.380,00**

Table 6 Hardware costs of the project

The software costs are the following.

Software	Cost (€)
Windows 10	145,00

Microsoft Office 365 personal	69,00
FIWARE Lab cloud	0
Total	**214,00**

Table 7 Software costs of the project

Finally, the total cost of the project.

Total costs	**Cost (€)**
Personal	7.430,00
Hardware	1.580,00
Software	214,00
Total	**9.224,00**

Table 8 Total costs of the project

The total cost of this project, with the personal, hardware and software costs result in a total of NINE THOUSAND TWO HUNDRED TWENTY-FOUR EUROS.

8. Bibliografía

[1] - Dieter Uckelmann, Mark Harrison, Florian Michahelles. An Architectural Approach Towards the Future Internet of Things.
[2] - Laura García, Lorena Parra, Jose M. Jimenez, Jaime Lloret, and Pascal Lorenz. IoT-Based Smart Irrigation Systems: An Overview on the Recent Trends on Sensors and IoT Systems for Irrigation in Precision Agriculture. (2020)
[3] - Barrionuevo, J.M., Berrone, P. & Ricart, J. E. Smart Cities, Sustainable Progress. IESE Insight, Vol. 14, pp. 50-57. (2012)
[4] - Internet of Things (IoT) Communication Protocols: Review Shadi Al-Sarawi, Mohammed Anbar, Kamal Alieyan, Mahmood Alzubaidi.
[5] - https://zigbeealliance.org/solution/zigbee/
[6] - https://www.thethingsnetwork.org/docs/lorawan/architecture.html
[7] - http://mqtt.org/faq
[8] - Ivan Ganchev, Zhanlin Ji, Mairtın O'Droma A Generic IoT Architecture for Smart Cities (2014)
[9] - https://eu-smartcities.eu
[10] - https://secmotic.com/fiware-smart-cities/
[11] - https://www.businesswire.com/news/home/20130412005065/en/IDC-Government-Insights-Unveils-Smart-City-Maturity
[12] - https://eu-smartcities.eu/clusters
[13] - Ivan Ganchev, Zhanlin Ji, Mairtın O'Droma A Generic IoT Architecture for Smart Cities (2014)
[14] - https://www.statista.com/statistics/809750/worldwide-popularity-ranking-database-management-systems/
[15] - https://www.fiware.org/2011/05/17/what-is-fiware/
[16] - https://www.sap.com/industries/smart-cities.html
[17] - https://dss.tcs.com/smart-city-solutions/
[18] - https://www.fiware.org/2015/03/25/fiware-a-standard-open-platform-for-smart-cities/
[19] - https://www.fi-ppp.eu
[20] - https://www.kowi.de/Portaldata/2/Resources/horizon2020/FI-PPP-final-report.pdf
[21] - https://cordis.europa.eu/project/id/285192
[22] - https://www.zhaw.ch/no_cache/en/research/research-database/project-detailview/projektid/957/
[23] - https://blog.zhaw.ch/icclab/fi-ppp-xifi-fi-ops/
[24] - https://oascities.org/wp-content/uploads/2019/01/20190117_OASC_FIWARE_Ahle_v2.pdf
[25] - https://www.fiware.org/foundation/
[26] - https://www.fiware.org/foundation/bylaws/
[27] - https://www.fiware.org/about-us/
[28] - https://www.fiware.org/community/evangelists/
[29] - https://www.fiware.org/community/fiware-experts/
[30] - https://www.fiware.org/community/fiware-accelerator-programme/
[31] - https://www.slideshare.net/FI-WARE/aceleradoras-image
[32] - https://www.fiware.org/startup-accelerator/
[32] - https://www.fiware.org/business-incubator/
[33] - https://www.fiware.org/success-stories/digitanimal/
[34] - https://www.fiware.org/success-stories/everimpact/
Mundus
[35] - https://www.fiware.org/community/fiware-mundus/
[36] - https://www.fiwaremexico.org/wp-content/uploads/2017/07/D4.2_FIWARE-Mexico_Mapping-FIWARE-good-practices-and-critical-success-factors.pdf

[37] - https://www.fiware.org/news/fiware-at-the-heart-of-the-smart-city-innovation-in-the-usa/
[38] - https://discoveryproject.eu/canada-next-step-fiware-mundus/
[39] - http://www.gatv.ssr.upm.es/index.php/2019/01/24/fiware-presentation-in-the-smart-cities-conference-during-the-start-me-up-hk-festival-hong-kong-24th-january-2019
[40] - https://www.eurocham.com.hk/event_galleries/building-an-open-ecosystem-for-smart-city-developments-in-hong-kong-and-europe
[41] - https://www.fiware.org/2019/02/05/happy-new-year-of-the-pig-fiware-mundus-visits-hong-kong/
[42] - https://www.fiware.org/news/fiware-at-the-smart-city-expo-in-casablanca-morocco-17-18-may-2017/
[43] - https://www.fiware.org/community/fiware-ihubs/
[44] - https://www.fiware.org/wp-content/uploads/2020/06/FF_iHubs_GuideforApplicants.pdf
[45] - https://www.udgalliance.org
[46] - https://fiwareinnova.org
[47] - http://www.fiwarerioja.org
[48] - https://fiware.zone
[49] - https://www.fiware.org/news/andalusian-regional-government-junta-de-andalucia-and-telefonica-to-drive-smart-cities-and-iot-through-the-fiware-zone-initiative/
[50] - https://www.fiware.org/news/fiware-foundation-presents-the-new-fiware-ihubs-network/
[51]https://www.fiware.org/news/new-fiware-ihubs-open-in-the-u-s-and-europe-to-accelerate-innovative-business/
[52] - https://www.fiware.org/developers/fiware-lab/
[53] - https://fiware-training.readthedocs.io/es_MX/latest/ecosistemaFIWARE/FIWARELab/
[54] - https://docs.google.com/forms/d/e/1FAIpQLSfIz2JxJ8zwytwYG9CJpauYPaOkiJEs88SMqBul9x1UWdzh3w/viewform
[55] - https://join-fiware-lab-guide.readthedocs.io/en/latest/1.essentials/2.requirements/
[56] - https://join-fiware-lab-guide.readthedocs.io/en/latest/1.essentials/3.steps_join/
[57] - https://www.openstack.org/software/
[58] - https://join-fiware-lab-guide.readthedocs.io/en/latest/3.process/3.upgrade/
[59] - https://join-fiware-lab-guide.readthedocs.io/en/latest/3.process/5.configuring/
[60] - https://join-fiware-lab-guide.readthedocs.io/en/latest/5.management/2.account/
[61] - https://cloud.lab.fiware.org/project/instances/
[62] - https://data.lab.fiware.org
[63] - https://data.lab.fiware.org/about
[64] - https://data.lab.fiware.org/datarequest
[65] - https://fiware-training.readthedocs.io/es_MX/latest/ecosistemaFIWARE/ocb/
[66] - https://fiware-orion.readthedocs.io/en/master/admin/install/index.html
[67] - https://fiware-orion.readthedocs.io/en/master/admin/diagnosis/index.html#resource-availability
[68] - https://fiware-orion.readthedocs.io/en/master/admin/yum/index.html
[69] - https://fiware-orion.readthedocs.io/en/master/quick_start_guide/index.html
[70] - https://fedoraproject.org/wiki/EPEL
[71] - https://fiware-orion.readthedocs.io/en/master/user/walkthrough_apiv2/index.html
[72] - https://www.cambus.net/parsing-json-from-command-line-using-python/
[73] - https://curl.haxx.se
[74] - https://knowage.readthedocs.io/en/6.1.1/user/NGSI/README/index.html
[75] - https://fiware-orion.readthedocs.io/en/1.11.0/user/structured_attribute_valued/index.html#structured-attribute-values
[76] - https://fiware-orion.readthedocs.io/en/1.11.0/user/pagination/index.html#pagination
[77] - https://www.slideshare.net/finodexproject/fiware-generic-enablers-introduction
[78] - https://www.fiware.org/developers/catalogue/

[79] - https://www.etsi.org/deliver/etsi_gs/CIM/001_099/009/01.02.01_60/gs_CIM009v010201p.pdf
[80] - https://fiware-draco.readthedocs.io/en/latest/quick_start_guide/index.html
[81] - https://nifi.apache.org
[82] - https://knowage-suite.readthedocs.io/en/7.2/
[83] - https://kurento.readthedocs.io/en/stable/
[84] - https://perseo.readthedocs.io/en/latest/
[85] - https://fimac.m-iti.org/5a.php
[86] - https://opcfoundation.org/about/opc-technologies/opc-ua/
[87] - https://fiware-pep-proxy.readthedocs.io/en/latest/
[88] - https://index.ros.org/doc/ros2/
[89] - https://tools.ietf.org/html/rfc6749
[90] - https://marketplace.fiware.org
[91] - https://marketplace.fiware.org/pages/solutions
[92] - https://marketplace.fiware.org/pages/solutions/9eae8fbd2402886a64f169f8
[93] - https://marketplace.fiware.org/pages/solutions/96afb6f8ccaef2f85aec6d77
[94] - https://marketplace.fiware.org/pages/platforms
[95] - https://marketplace.fiware.org/pages/solutions/c8a29a1f62a688fa58d8bc8d
[96] - https://marketplace.fiware.org/pages/solutions/fee10a6a7755e554686664fd
[97] - https://marketplace.fiware.org/pages/devices
[98] - https://marketplace.fiware.org/pages/solutions/48fcf7221056ca4734a980f5
[99] - https://www.greenglobe.es/night-eye/
[100] - https://marketplace.fiware.org/pages/solutions/e039c8de15e9c2afd86d9555
[101] - https://smartcities.hopu.eu//
[102] - https://marketplace.fiware.org/pages/enablers
[103] - https://marketplace.fiware.org/pages/solutions/5c48e1d037f08d143452189c
[104] - https://marketplace.fiware.org/pages/solutions/e86ee0ac0f75b7e6d86fadde
[105] - https://marketplace.fiware.org/pages/cefs
[106] - https://marketplace.fiware.org/pages/training
[107] - https://marketplace.fiware.org/pages/solutions/06b1e24b3ddd69cbe023a7ca
[108] - https://marketplace.fiware.org/pages/consultancy
[109] - https://marketplace.fiware.org/pages/solutions/66ebf0ff49c375ae4bb91dec
[110] - https://marketplace.fiware.org/pages/solutions/014ca03d5d967e296b9abb42
[111] - https://marketplace.fiware.org/pages/solutions/4cee5102717a2b0d36681b7e
[112] - https://www.fiware.org/success-stories/breeze/
[113] - https://www.fiware.org/success-stories/ubiwhere/
[114] - https://marketplace.fiware.org/pages/solutions/7905c2965f2dfc18a882ba82
[115] - https://www.fiware.org/success-stories/wise-town/
[116] - https://www.fiware.org/success-stories/sammy/
[117] - https://marketplace.fiware.org/pages/solutions/00d3fb3501261fdb5ab32dff
[118] - https://www.fiware.org/wp-content/uploads/2018/03/SaMMYs-Use-of-FIWARE-Tech.pdf

Universidad de Alcalá
Escuela Politécnica Superior

ESCUELA POLITECNICA
SUPERIOR

Universidad
de Alcalá