COSIBAS

Cognitive Services for IoT-based Scenarios

D1.1- COSIBAS Requirements

***Delivery Date*:**	***Project Number*:**	***Responsible partner*:**
M12	ITEA3 Call 4 17022	PRODEVELOP

Document History

Version	Date	Author	Description
0.1	25/11/2019	Prodevelop	Table of Contents and first content related to methodology
0.4	13/01/2020	Prodevelop HI-Iberia	Technological State of The Art Functional requirements for Smart Operations
0.5	15/01/2020	Experis IT / USAL HI-Iberia Beia	Functional requirements for Smart Grid
0.6	27/01/2020	Prodevelop	Non-Functional Requirements
0.7	31/01/2020	HI-Iberia	Refinement of Functional and non-functional requirements
0.8	03/02/2020	Prodevelop	Minor adjustments
1.0	05/02/2020	Prodevelop Experis HI-Iberia	Final version approved for release

CONTENT

1 INTRODUCTION

COSIBAS targets the next step in IoT-based applications and solutions, namely the integration of semantic and cognitive AI technologies. The amount of data is rising and requires a sophisticated interpretation through analytics to generate business value in terms of faster detection, better forecasts, and improved decisions with an overall increased flexibility. Whilst current IoT-stacks are frequently focused on handling data or streams of information, semantic properties and ontologies are generally missing. The objective pursued by COSIBAS is to put the focus on the existing IoT-technologies to create an open, standardised approach as well as to define and develop cognitive enablers/services improving the state of the art of the current IoT-stack. Based on these enabling factors, cognitive services will be able to interpret and evaluate the status of the whole system to build and transact on a cognitive model, which is the basis for the interaction, decision-support as well as to support intelligent mechanisms to enable seamless interoperability respecting semantic. COSIBAS will rely on existing IoT-technologies and standards and will evolve existing semantic and cognitive technologies into a set of enablers and services based on proven elements of AI approaches and components. Besides that, COSIBAS will be validated through the implementation and deployment of two use cases, which will be running in real conditions during an agreed timeframe so that the Consortium will be able to gather feedback and measure expected KPIs.

The aim of this document is to carry out a detailed review of the state of the art of the current technological stack (i.e. IoT frameworks, AI platforms, AI cloud-based platforms, etc.) which will enable the set up of the COSIBAS platform as well as the list of functional and non-functional requirements whose development will enable the Consortium to deploy and validate both the Smart grid and Sea traffic management use cases.

2 Technological state-of-the-art

2.1 IoT Frameworks

The number of devices that are coming online and becoming part of IoT is exploding. Analysts expect that by 2020 the human population will grow to 7.6 billion with 50 billion devices connected to the Internet[1]. The challenge for this IoT framework is to ensure these emerging IoT devices can connect securely and efficiently over the Internet and to each other, since there is still a big need for shared IoT standards. There are several industry alliances and research projects that aim to deal with this challenge. Below we present and analyse the most relevant IoT frameworks. Figure 1 Figure 1: IoT platform comparisondepicts the different IoT platforms according to their capabilities.

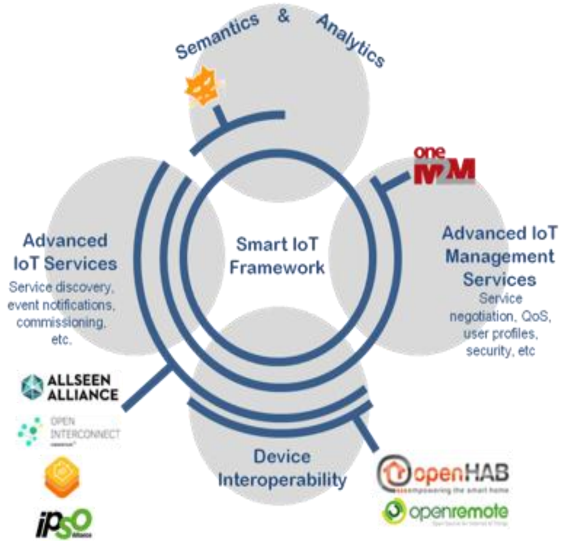

FIGURE 1: IOT PLATFORM COMPARISON

Device Interoperability

In order to enable interoperability between all the heterogeneous devices that surround us, several integration platforms have appeared, such as OpenRemote[2] or openHAB[3]. These platforms have no intention to replace existing solutions but rather enhance them with an additional abstraction layer.

- **OpenRemote** is an open source project whose first main objective was the integration of different protocols and solutions available for home automation as well as offering visualization tools. At first, this project focused on the home, but over time it was growing in other areas related to medical care, hospitality, entertainment and public spaces. To offer intelligent solutions to specific problems in various sectors, OpenRemote allows the connection of devices and services to custom-designed panels rendered on mobile

[1] Raymond James & Associates Inc., The Internet of Things, A Study in Hype, Reality, Disruption, And Growth, A Technology & Communications Industry report, January 24, 2014

[2] http://www.openremote.com/community (Accessed on February 2020)

[3] http://www.openhab.org (Accessed on February 2020)

applications. Moreover, OpenRemote Protocol agents facilitate the connection of different data sources, live sensors, and controls creating meaningful connections between the data. This allow the creation of cross domain applications which support a large variety of automation protocol standards. In addition, it provides APIs for the customisation and extension of its capabilities.

- **OpenHAB** is an open source automation software, vendor independent and home focused technology. It is characterized by providing a high degree of integration of technologies and different systems, and an engine for the design of rules, with triggers, scripts, actions and notifications and voice control based on times and events. It can be run on different operating systems such as Linus, macOS, Windows, Rasperry Pi, PINE64, Docker, etc., and accessed from different types of applications, including iOS and Android. It can be integrated into the most popular smart home cloud-based platforms, such as Google Assistant, Amazon Alexa, Appel HomeKit and IFTTT, and can be run on the hardware directly without the need for a cloud service for the user have the data in local, provide more control. This last point is an added value to highlight. On the contrary, its high flexibility and customization possibilities require people with knowledge of the technology to configure the system. Besides that, OpenHAB is a modular product that facilitates integration with other systems, even at runtime, it also allows data to be extracted from devices, which facilitates the replacement of technologies without modifying user rules and interfaces. The latter not only enables interoperability between the different devices, but also provides additional services such as definition and execution of control rules, generation and management of event notifications, tools for manually commission installed devices, or tools for designing and generating mobile GUIs. It must be highlighted that these platforms allow the integration of any device, independent of its protocol nonetheless, the corresponding controller must be implemented so that the communication between device and platform can be performed.

These integration platforms lack of methods for automatic discovery and commission of devices and require a gateway or hub that transforms device messages into platform messages.

Advanced IoT Services

To solve these drawbacks, several collaborative software frameworks from different industrial alliances have been developed such as AllJoyn[4] (AllSeen Alliance), Homekit[5] (Apple) or IoTivity[6] (Open Interconnect Consortium). These frameworks define and implement common services and interfaces, which are embedded in devices and applications in order to enable device discovery, control, monitoring, and reception of notifications. Note that devices still operate with their embedded communication technology (Bluetooth LE, LowPAN, Zigbee, Z-Wave, etc.), since framework services and interfaces are implemented on top of them.

- **AllJoyn** is a collaborative Open Source project of the AllSeen Alliance that provides a general networking framework and system services for interoperability between connected devices related to home, smart TV, smart audio, automotive, broadband gateways and software

[4] https://allseenalliance.org/developers (Accessed on February 2020)
[5] https://developer.apple.com/homekit (Accessed on February 2020)
[6] https://www.iotivity.org (Accessed on February 2020)

applications to create dynamic proximal networks. The project 's main goal is the creation of an open, universal, secure and programmable software and frameworks which enables the creation of interoperable smart home solutions that can be used by all major platforms and operating systems. For that purpose, AllJoyn offers support to hardware manufacturers and software developers to create interoperable products compliant with the framework. In favour of this framework we find the variety of services provided. Among the services provided by this framework are the incorporation services to connect a device to a Wi-Fi network, configuration to define the attributes of a device, notification to send and receive notifications from other devices in the network, control panel for administration remote device through a single application and common device model for monitoring and management of devices regardless of manufacturer. On the other hand, this framework has limitations related to communication because the communication layer is limited to WI-FI.

- **HomeKit** is a framework developed by Apple focused on the configuration, communication and control of smart home devices. One of the main advantages offered by this framework lies in the possibility of configuring the actions of the devices through voice commands, in addition to the applications. This framework incorporates the Siri artificial intelligence application that allows devices to be managed through web services using natural language. Furthermore, HomeKit configuration is synchronized across all user's apps and devices, and it's integrated into iOS, watchOS and tvOS. However, it is still an Apple product focused primarily on its devices and, therefore, has certain limitations, such as the need for the use of a specific Homekit accessory protocol, an encryption coprocessor in the products or, failing that, the use of a gateway for the connection of the devices to the HomeKit service.
- **IoTivity** is an Open Source project hosted by the Linux Foundation[7], sponsored by the Open Connectivity Foundation[8] (OCF), a group of industry leaders such as Samsung Electronics and Intel who together are developing standard specifications, promoting a set of interoperability guidelines, and providing a certification program to enable the Internet of Things. However, IoTivity is independent from OCF. Its architectural objective is to create a new standard that allows devices to connect to each other and to the Internet. Although it implements an Open Source reference of the OCF specifications, it is not limited to its requirements. It provides discovery, data transmission, device and data management capabilities. This software framework is independent of the device, uses the restricted application protocol (CoAP) as an application layer, but requires that the network layer be IP, being compatible with protocols such as WI-FI, Ethernet, Zigbee, Bluetooth, Bluetooth low energy (IPSP) and Z wave.
- Similarly, but assuming that all objects are reachable through an IP address, the IPSO alliance[9] is defining guidelines in order to allow that every object is able to expose its resources, properties, and services through URIs that can be discovered and retrieved by client applications. IPSO objects, which use many CoAP directives, must construct the URIs describing their resources following a defined namespace. This namespace is communicated to the client apps before it starts querying the object.

[7] https://www.linuxfoundation.org (Accessed on February 2020)
[8] OCS Website: https://openconnectivity.org (Accessed on February 2020)
[9] IPSO Alliance Website: http://www.ipso-alliance.org (Accessed on February 2020)

Advanced IoT Management Services

All the aforementioned frameworks are proximity-based, i.e. devices dynamically discovered and communicated within each other over direct peer-to-peer (P2P) connections, whether they are part of a homogeneous or heterogeneous network (remotely via internet). However, remote communication between different networks cannot be controlled or managed, making impossible to manage and maintain the non-proximal networks or to ensure a good QoS for the provided services. The oneM2M initiative[10] aims to cover these aspects and has recently released their specifications, which define a software layer seated between M2M applications and communication HW/SW and provides data transport. For example, oneM2M definition adds to the previous frameworks a communication middleware between proximal network and non-proximal networks, which provides standard means for remote monitoring, management and control functionalities.

oneM2M provides a middleware or IoT service layer that provides a series of functions through uniform APIs, vendor independent and globally standardized, towards IoT applications. Among the main features provided by this set of functions are the identification, authentication and authorization of users and applications, end-to-end data encryption, device management, remote provisioning and service activation, connectivity configuration and transmission programming. data, group management and application and data discovery functions, data aggregation, buffering in case of lack of connectivity and activation when restored.

The architecture used in the oneM2M standard uses a three-layer model: applications, services and networks. The application layer resides within the applications of the devices and provides a standardized interface for administration. Between the application layer and the network layer is the service layer composed of common service entities (CSE) that play a role similar to that exposed in the application layer. Finally, the network layer ensures that the devices, sensors and applications function independently of the network. All of this intrinsic functionality allows developers to focus on the development of IoT applications for devices.

Semantics & Analytics

Interoperability between heterogeneous devices can be achieved using the presented integration solutions. However, when looking at the IoT ecosystem from the developer's or user's point of view, resources cannot easily be discovered and accessed by client applications since every integration platform defines its own information model (ontology) and semantics, required to be known by client applications. The HyperCAT[11] project aims to make the data resources of IoT devices machine searchable, so that applications can commit resources on their own among the IoT ecosystem. HyperCAT defines an open, lightweight JSON-based hypermedia catalogue format for exposing collections of IoT resources along with linked-data descriptions of their capabilities over the web (as collections of URIs). Each HyperCat catalogue may expose any number of URIs, each with any number of RDF-like triple statements about it. As a result, applications will be able to find resources from connected devices across multiple data hubs on their own. HyperCAT solves the interoperability problem by allowing data from a sensor to be

[10] OneM2M, Global standard initiative, http://www.onem2m.org (Accessed on February 2020)

[11] HyperCAT Website: http://www.hypercat.io (Accessed on February 2020)

used by multiple types of IoT devices. Its operation is simple. Applications send an HTTP request message to the HyperCAT server and return a file with a catalogue of available resources with which applications can interact following HyperCAT protocols.

Regarding the reasoning capabilities of IoT models and interaction with natural language, there have been some advancements in recent years, both in academia and in industry. Wit.ai[12] develops interfaces to IoT devices for natural language communication. Their models constitute an example of similar projects in the field of human-machine interfaces. However, the connection between IoT and natural language data is severed due to the decoupling of both. Hence, evidence from IoT does not influence the natural language understanding, and vice versa. Another area of academic research is dedicated to the use of neural networks in IoT[13] as they constitute the state-of-the-art in many related disciplines. However, even when they are capable of delivering state-of-the-art solutions, the interpretability of neural models remains only marginally tackled.

Nowadays, the traditional controlling strategies are moving towards the distributed control system and more advanced technologies, such as IoT. This upgrading is more evident in the industrial sectors. In these sectors, the companies always tend to have a stable system by considering flexibility and minimizing the operational costs. These are the reasons that IoT is considered as a solution to fulfil the requirements in the industrial sectors[14]. In fact, by using IoT, the system would be able to have data transmission as fast as is possible, and the decision making would be performed locally and based on this transmitted information. Moreover, an IoT based system would be able to perform big data analysis by utilizing several smart facilities in order to provide some recommendations automatically, namely predictive maintenance time. For implementing such system, a high-performance intelligent model is required for employing semantic knowledge base, in order to have high precision data classifications[15]. This is the set point that shows the need of these semantic based system for data analysing.

In fact, the proposed IoT model is a cost effective and flexible system comparing with the typical and traditional automation infrastructures. In the current model, the solutions are isolated and limited for collective intelligence. The main performers contribute to overcome these problems and focus on the same issues comparing with IoT, which shows higher necessities of availability, security, integrity and flexibility. Several companies and industrial sectors invested on the traditional automation systems, are now interested on the more advanced technologies, such as IoT actors and system. However, so far all provided industrial solutions do not fulfil all

[12] Wit.ai Website: https://wit.ai (Accessed on February 2020)

[13] IotsWorldcongress: From chatbots to deep learning – How artificial intelligence drives IoT to the next level: A panel discussion

[14] A. Sajid, H. Abbas and K. Saleem, "Cloud-Assisted IoT-Based SCADA Systems Security: A Review of the State of the Art and Future Challenges", IEEE Access, vol. 4, pp. 1375-1384, 2016

[15] A. Mazayev, J. Martins and N. Correia, "Interoperability in IoT through the Semantic Profiling of Objects", IEEE Access, 2017

requirements, especially regarding the operational costs, model architecture, flexibility and availability [16], and the need of innovation and changes are obvious [17].

Despite the current drawbacks, the Industrial IoT (IIoT) is also experiencing a strong growth[18], hand in hand with the Industry 4.0 revolution, so much so that traditional automation systems (e.g. SCADA), are being replaced by these models and big data engines that, not only provides a gateway to read data from the facilities, but also incorporates smart features, enriching the decision ability with contextual information. This new paradigm is replacing the traditional industrial model, popularizing the analysis of big amounts of data, but also applying the intelligence component able to transform this knowledge into predictive-maintenance recommendations, corrective actions for anomalies detected when tracking the parameters of certain processes or documentation generated automatically by the system. All these characteristics need of a precise cognitive system, able to transform all this data into a semantic knowledge base, with an accurate ability to cluster the information, demonstrating a high capacity to difference slight aspects given by the environment of the data.

This IIoT model, despite of being a much more flexible and cost-saving tool than traditional automation systems, where solutions are isolated, heterogeneous, with limited collective intelligence and at a high cost in terms of investment and development, is far from being a wide-implanted solution. Main actors in this sector, share these concerns and aim to the same specific problems than in IoT, with the added problem of higher requirements in terms of security, availability, integrity and resilience. Industry, that traditionally has demonstrated an important funding capacity, has attracted the attention of important technological competitors as GE (Predix[19] platform based on Microsoft Azure), Siemens (MindSphere[20]) or even other IoT actors like Samsung (that has recently joined the EdgeX Foundry[21] supported by Linux). But these alternatives do not meet all needs in terms of cost, open architectures, support, integration capacity, etc. Industry stakeholders are not unconnected to this reality and they are sensitive to the barriers that restrain the development of smart automated environments in its sector, so they are demanding a reliable solution able to meet all these requirements. Companies' managers and industry administrators are aware of these barriers, and they are attempting to overcome these issues and provide a reliable solution, especially concerning the advancement of intelligent and automated management and control on different sections of their companies or industry.

---

[16] J. Li, J. Jin, D. Yuan and H. Zhang, "Virtual Fog: A Virtualization Enabled Fog Computing Framework for Internet of Things", IEEE Internet of Things Journal, vol. 5, no. 1, pp. 121-131, 2018

[17] H. Khaleel, D. Conzon, P. Kasinathan, P. Brizzi, C. Pastrone, F. Pramudianto, M. Eisenhauer, P. Cultrona, F. Rusina, G. Lukac and M. Paralic, "Heterogeneous Applications, Tools, and Methodologies in the Car Manufacturing Industry Through an IoT Approach", IEEE Systems Journal, vol. 11, no. 3, pp. 1412-1423, 2017

[18] https://www.accenture.com/us-en/insight-industrial-internet-of-things (Accessed on February 2020)

[19] https://www.ge.com/digital/predix-platform-foundation-digital-industrial-applications (Accessed on February 2020)

[20] https://www.siemens.com/global/en/home/products/software/mindsphere.html (Accessed on February 2020)

[21] https://www.edgexfoundry.org (Accessed on February 2020)

2.2 Reference IoT Architectures

From an architectural perspective, FIWARE[22], IOT-A[23] and OpenIoT[24], can be considered as starting points for expanding towards achieving additional features. These projects have defined architectural reference models for enabling interoperability and integration of heterogonous IoT systems, but they do not consider implementing intelligence, context aware and adaptation capabilities such as, self-configuration, end-to-end security, VOs composition or data quality, capabilities that will guarantee the performance of services and applications residing at a higher-level of abstraction.

- **FIWARE** is a middleware platform driven by the European Union under the Future Internet Public Private Partnership Programme[25], for the development and global deployment of Smart Applications for Future Internet in multiple vertical sectors. FIWARE easily allows the development, deployment and configuration of global Internet applications whose maximum objective is to become a standard platform with reusable solutions. This platform is a future-proof inasmuch as the new technologies or devices can be integrated effortlessly and logically. In fact, FIWARE stands for open source platform for an intelligent digital future. FIWARE platform provides a rather simple yet powerful set of APIs that facilitate the development of Smart Applications in multiple vertical sectors. An open source reference implementation of each of the FIWARE components is publicly available. The difference between this platform and others is that there is a development community of application components that can be used freely and for free. FIWARE has the seal of European Union and has been developed by large companies such a Telefónica, Orange and NEC, and many universities. These features make FIWARE an ideal platform for COSIBAS objectives.
- The **Internet of Things-Architecture** (IoT-A) was designed in the framework of the European Lighthouse Intergraded Project from 2010 to 2013. It is part of several EU-funded research project aimed at achieving Standardisation in the Internet of Things that have as their objective to achieve interoperability between different IoT systems. This project developed an Architectural Reference Model[26] (ARM) for generating reference architectures based on domain specific requirements in the Internet of Things domain together with the definition for the technical design of its protocols, interfaces and algorithms to achieve its objective. This interoperability had to be developed from the communication level as well as at the service and knowledge levels across different platforms established on a common grounding. The IoT-A project developed common tools and methodologies to achieve this. A major benefit of the IoT-A ARM is the capability of generating architectures for specific systems. This architecture generation is done by providing best practices and guidance for helping to translate the ARM into concrete architectures. The benefit of such a generation scheme for IoT architectures is not only a certain degree of automatism of this process, and

[22] FIWARE Foundation Website: https://www.fiware.org (Accessed on February 2020)
[23] IoT-A Website, Internet of Things architecture, http://www.iot-a.eu (Accessed on February 2020)
[24] OpenIoT Website: http://www.openiot.eu (Accessed on February 2020)
[25] https://www.fi-ppp.eu (Accessed on February 2020)
[26] IOT-A Project Consortium. Final architectural reference model for the IoT. Technik GmbH, Berlin, Germany, Tech. Rep. ed.; 2013

thus the saved R&D efforts, but also that the decisions made follow a clear, documented pattern.

- **OpenIoT** is an open source Internet of Things platform, which allows the collection, storage and processing of data stemming from heterogeneous set of IoT entities and resources. The platform introduces Sensing as a Service paradigm by interweaving the cloud computing with Internet of Things technologies. In particular, the OpenIoT platform provides the functionalities for different purposes. First, it allows to collect and process data from multifarious IoT sources including physical and cyber sensors, APIs and other sources. Secondly, it includes semantic annotation in sensor data streams as the OpenIoT data model is based on the W3C Semantic Sensor Networks (SSN) specifications. Moreover, it allows storing and enabling access to sensors data via cloud computing infrastructure as well as visualization of the sensor data using charts, graphs, maps etc.

2.3 AI Platforms

With the success of IBM Watson proving that natural language processing and understanding has made, a great leap[27] the development of AI components on the market is funnelling in AI platforms. The most prominent is IBM Watson[28] followed by those from Google, nVIDIA, Microsoft, and Amazon to name the most prominent players. These AI platforms form a stack of themselves[29] as is depicted in **¡Error! No se encuentra el origen de la referencia.**. The contained knowledgebases, ontologies, and semantic models are not shown in this figure.

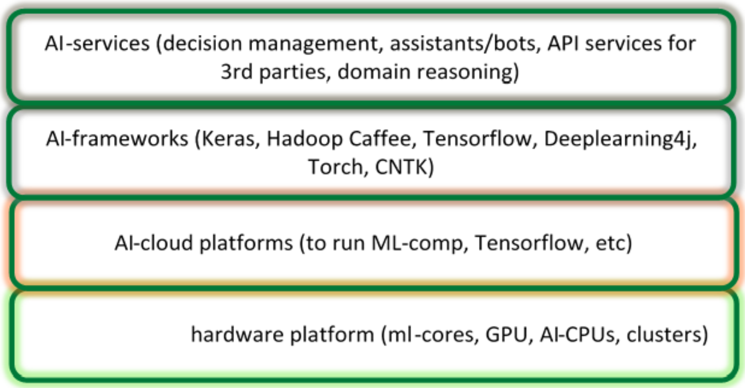

FIGURE 2 - THE AI TECHNOLOGY STACK

AI-CLOUD PLATFORMS

In the last years AI has reached a high level of popularity and, in parallel, the number of AI services has increased. Cloud service providers (CSPs) have started to offer services dedicated to specific tasks, such as facial recognition, detection of objects in video or text analysis. Some of these providers have even started to offer a more useful solution: an AI as a Service (AI PaaS) platform. Instead of offering a separate service leveraging AI capabilities, they provide full-scale AI platforms. AI PaaS solves the intrinsic problem of AI, which is that it consumes a large amount

[27] https://www.ibm.com/midmarket/us/en/article_Smartercomm5_1209.html (Accessed on February 2020)
[28] IBM Watson: https://www.ibm.com/watson (Accessed on February 2020)
[29] http://insightaas.com/how-ai-platforms-are-stacking-up (Accessed on February 2020)

of resources, thus offering money, time and effort saving. Following Gartner's words[30], we can define *AI PaaS as a set of AI and machine learning (ML) platform services for building, training, and deploying AI-powered functionalities for applications*.

Whereas Gartner sees AI PaaS as a set of separate AI services, they also can be understood from the perspective of the classic Platform as a Service (PaaS) model. In a PaaS, a cloud vendor provides an environment for building, deploying, and maintaining applications. Similarly, although there is no general architecture or model for AI PaaS, we can identify a number of elements that are common to the majority of today's AI PaaS and AI service platforms: infrastructure, data storage, pre-trained ML models and AI APIs. The market is still forming, but it can be affirmed that an IA platform is composed of these 4 elements. AI Cloud platforms offer affordable environments with high performance for AI. Cloud computing provides high scalability and resource availability, as well as the ability to manage and store large datasets. Of all AI cloud platforms stand out, for their higher degree of maturity, Amazon AWS AI, IBM Watson, Microsoft Azure AI and Google AI platform:

- **IBM Watson**[31] is a business-oriented AI platform that offers services based on natural language processing, visual recognition and machine learning. Watson offers pre-built services, as well as an API to develop new AI solutions and tools to manage the full lifecycle of an application based on AI. Among other AI services, Watson offers natural language processing, automatic translation, AI assistants and text-to-speech and speech-to text conversion.
- **Google AI Platform**[32] is an end-to-end machine learning as a service (ML PaaS) platform targeting data scientists, ML developers, and AI engineers. The Cloud AI Platform has services to tackle the lifecycle of machine learning models. The platform offers building blocks to develop and deploy sophisticated machine learning models including, among others, flexible and powerful training and deploying ML models.
- **Microsoft Azure AI**[33] offers a set of Python-based machine learning services for business that can be used across Cloud and Edge. It includes a set of pre-trained AI services, which can be accessed through an API to build applications with cognitive features. Some of these services are language and speech processing, computer vision, cognitive search and decision management. It also offers a set of development tools for the creation of custom Machine Learning models.
- **Amazon AWS AI**[34] is a collection of public cloud computing services that together form a cloud computing platform, offered over the Internet by Amazon. It provides a set of abstract technical infrastructure and distributed computing building blocks and tools. Moreover, it also offers an integrated development environment to build, train and deploy custom machine learning models. Examples of some services are speech and text processing, computer vision and cognitive search, among others. Regarding the access to the services,

[30] https://www.gartner.com/doc/3883863/hype-cycle-artificial-intelligence (Accessed on February 2020)
[31] https://www.ibm.com/watson (Accessed on February 2020)
[32] https://cloud.google.com/ai-platform (Accessed on February 2020)
[33] https://azure.microsoft.com/en-us/overview/ai-platform (Accessed on February 2020)
[34] https://aws.amazon.com/ai (Accessed on February 2020)

instead of exposing directly the services to end users, the access is provided by means of APIs. Web services are accessed via HTTP, using REST and SOAP protocols.

- **Oracle AI**[35] is a Cloud platform aimed for business, data scientists and developers. It offers pre-trained AI models and development tools, which cover the different steps of the AI development lifecycle, from data management to application development and data science. It also offers pre-built Adaptive Intelligent Apps, which are cloud-based AI applications designed for business. AI-frameworks.

AI-Frameworks

The most important and recognized AI frameworks for both industrial and academic partners are described below:

- **TensorFlow**[36] is an open-source Machine Learning framework developed by Google and focuses on neural networks and deep learning. It provides a comprehensive and flexible ecosystem of tools (which rely mostly on graphs and data visualization), libraries and community resources. TensorFlow is distributed under Apache License 2.0 and is available for the most common desktop and mobile platforms. It has stable APIs for Python and C++, as well as third-party APIs for several other languages. Since TensorFlow was developed as a modular system, it is possible to use only parts of the framework.
- **Keras**[37] is an open-source Python library for neural network creation. Keras has been designed as a complement to other Machine Learning libraries to offer more intuitive high-level tools to develop AI applications. It can be run on top of TensorFlow, Theano or CNTK. Keras supports a wide range of neural networks layers, such as dense layers, recurrent layers and convolutional layers.
- **Microsoft Cognitive Toolkit (CNTK)**[38] is an open-source framework for neural networks. It provides a low-level API, intended for neural network components creation, and a high-level API, which contains a set of building blocks for neural network definition and training. It also includes components that facilitate the management of large datasets. It has APIs for Python, C++ and C#.
- **Caffe**[39] is an open-source deep learning framework oriented to image processing implemented in C++ and has a Python API. Caffe was developed by Berkeley AI Research (BAIR), The Berkeley Vision and Learning Center (BVLC) and community contributors. It is lightweight, scalable and fast and offers pre-trained models for demo purposes.
- **Scikit-learn**[40] is an open-source Python Machine Learning library built on top of SciPy. It supports supervised and unsupervised learning.
- **PyTorch**[41] is a Python package for tensor computation and deep neural networks building. It has been designed to be efficient, intuitive and easy to use. It offers dynamic computation graphs, which can manage inputs and outputs that vary in length.

---

[35] https://www.oracle.com/artificial-intelligence (Accessed on February 2020)
[36] https://www.tensorflow.org (Accessed on February 2020)
[37] https://keras.io (Accessed on February 2020)
[38] https://github.com/Microsoft/CNTK (Accessed on February 2020)
[39] https://github.com/BVLC/caffe (Accessed on February 2020)
[40] https://scikit-learn.org/stable (Accessed on February 2020)
[41] https://github.com/pytorch/pytorch (Accessed on February 2020)

- **Deeplearning4j**[42] is a deep learning framework for Java and Scala. It is integrated with Hadoop and Apache Spark and provides methods for the definition and training of deep neural networks. This framework has several submodules, which cover different aspects, such as model training or visualization.
- **Theano**[43] is a Python library that allows the definition, evaluation and optimization of mathematical expressions. It is based on Python and has been one of the most widely used AI frameworks. However, the development of this framework is currently stopped.

AI-SERVICES

AI services are based on the application of machine learning techniques to emulate human thought processes. AI platforms offer a variety of pre-trained services, which can be accessed through an API and used to build applications. Some of these services are the following:

- Computer vision services: image and video processing. This includes image classification, content identification, object detection and object tracking, among others.
- Natural language processing: application of machine learning models to understand the meaning of texts and speech.
- Automatic translation.
- Speech-to-text conversion.
- Text-to-speech conversion.
- AI assistants: allow the creation of conversational interfaces for devices or applications.
- Cognitive search: machine learning applied to web search to obtain more relevant results.
- Decision management using predictive analytics.

2.4 Reference OWL-based Ontologies

There are several semantic descriptions designed for the IoT domain. The SSN ontology[44] is one of the most significant and widespread models to describe sensors and IoT related concepts. The SSN Ontology provides concepts describing sensors, such as outputs, observation value and feature of interest. However, it is a detailed description, containing concepts and properties that enable flexible descriptions over a very wide range of applications, but including non-essential components for many use cases that can make the ontology heavy to query and process if it is used as it is. The IoT-A model[45] and IoT.est[46] are some of the projects that extend the SSN ontology to represent other IoT related concepts such as services and objects in addition to sensor devices. IoT-A provides an architectural base for further IoT projects. The IoT-A model is overly complex for fast user adaptation and responsive environments. The IoT.est model extends the IoT-A model with extended service and test concepts.

[42] https://deeplearning4j.org (Accessed on February 2020)
[43] https://github.com/Theano/Theano (Accessed on February 2020)
[44] Compton, M., et al. (2012). The SSN Ontology of the W3C Semantic Sensor Network Incubator Group. Web Semantics: Science, Services and Agents on the World Wide Web 17: 25-32, 2012
[45] IoT-A Website, Internet of Things architecture, http://www.iot-a.eu (Accessed on February 2020)
[46] Wang, W., et al. (2012). A comprehensive ontology for knowledge representation in the internet of things. In Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on, pages 1793–1798. IEEE, 2012

Another initiative is the Spatial Data on the Web Working Group[47], a joint effort between the World Wide Web Consortium (W3C) and the Open Geospatial Consortium (OGC) that aims to standardise key ontologies for spatial, temporal and sensor data on the web. Several projects also work on semantic descriptions for the IoT, such as FED4FIRE[48] that currently has a semantic model focus on communications, VITAL[49] for smart cities, CityPulse[50] with more focus on data and OpenIoT[51] , which is an instantiation of SSN.

- **OneM2M** base ontology aims to provide a high-level ontology for the IoT market in order to provide a minimal set of common knowledge that enables the cross-domain syntactic and semantic interoperability. oneM2Montology is very abstract and general; thereby, oneM2M expects external ontologies that describe a specific domain of interest in a more detailed way to be mapped to the oneM2M base ontology. Additionally, oneM2M defines how to internetwork between devices and things from different domains is enabled.
- The **BIG IoT**[52] project follows the approach of reusing existing ontologies by combining them within an IoT data-sharing platform, also known as IoT data marketplace. There are three groups of the semantic modelling artefacts in BIG IoT. Two domain ontologies, that is, Mobility and Environment, provide terms to characterize the content of the data/service offering. Finally, the Application Ontology is used for organizing navigation through the model elements in the marketplace Web portal. All models follow Linked data principles and are published as schema.org custom extensions.
- **FIWARE** defined on the one hand, OMA NGSI interfaces to offer a homogeneous access to data, and on the other hand, a set of data models being standardized by ETSI ISG Context Information Management. Such Context Information provides the meta-data structure for sensors measurement and other data feeds from video, social media etc. Even when context is very simple to understand by human being, in order to provide artificial intelligence capabilities to smart systems, it is crucial to formalize and provide much more details about the context and make it available in conjunction with the data. A Context Information Management (CIM) system acts as a clearinghouse for publishing, discovering, monitoring and maintaining data according to relevant contexts for smart applications.

[47] http://www.w3.org/2015/spatial (Accessed on February 2020)
[48] FED4FIRE: http://www.fed4fire.eu (Accessed on February 2020)
[49] VITAL: http://vital-iot.eu (Accessed on February 2020)
[50] CityPulse: http://www.ict-citypulse.eu (Accessed on February 2020)
[51] OpenIoT: http://www.openiot.eu (Accessed on February 2020)
[52] BIG IoT: http://big-iot.eu/ (Accessed on February 2020)

3 COSIBAS Use cases and Pilots

3.1 Smart Grid Use Case

Description and goal

The main objective of the use case is to carry out simulation processes of a P2P energy trading system within a Smart City, so that the transfer of energy between producers and consumers can be optimized to make a Smart City sustainable. In order to achieve the above objective, it is necessary to calculate the energy needs of homes, office buildings, shopping centres, etc., as well as the estimate of the production of energy from solar panels, windmills, thermal power plants, etc. so that the energy exchange takes place. Machine Learning and Deep Learning algorithms will be used, which will make use of real data sets from external services. With these data the algorithms will be able to simulate how the energy negotiation process would be carried out in a real environment in the most efficient way. The results of the work packages will be tested in a mock-up that allows simulating these processes, so that it can be demonstrated with real data in a simulated environment the validity of these hypotheses.

One of the main objectives of Smart Cities is to become more sustainable through minimizing and optimizing the use of resources. Electricity is one of the most demanded resources. In this respect, one of the proposals that has aroused much interest in Europe is the process of buying and selling electricity (P2P model). This model proposes a paradigm shift in which the energy exchange between consumers helps them obtain energy at a lower price. This paradigm shift can be easily adapted thanks to the fact that the number of sensors deployed in cities continues to increase and therefore devices can connect even if the distance between them is large. The platform of that case study allows any user with an intelligent meter to enter the market as a buyer or seller, configuring their energy purchase and sale tariffs *vis-à-vis* the suppliers of energy distributed through the platform. The intelligent meter allows its user to control how much energy to buy or sell at any given time. The platform will also provide knowledge on the profit that will come from investing into storage systems (if the price of energy fluctuates during the day, the user may save money by using electricity at the times when its price falls).

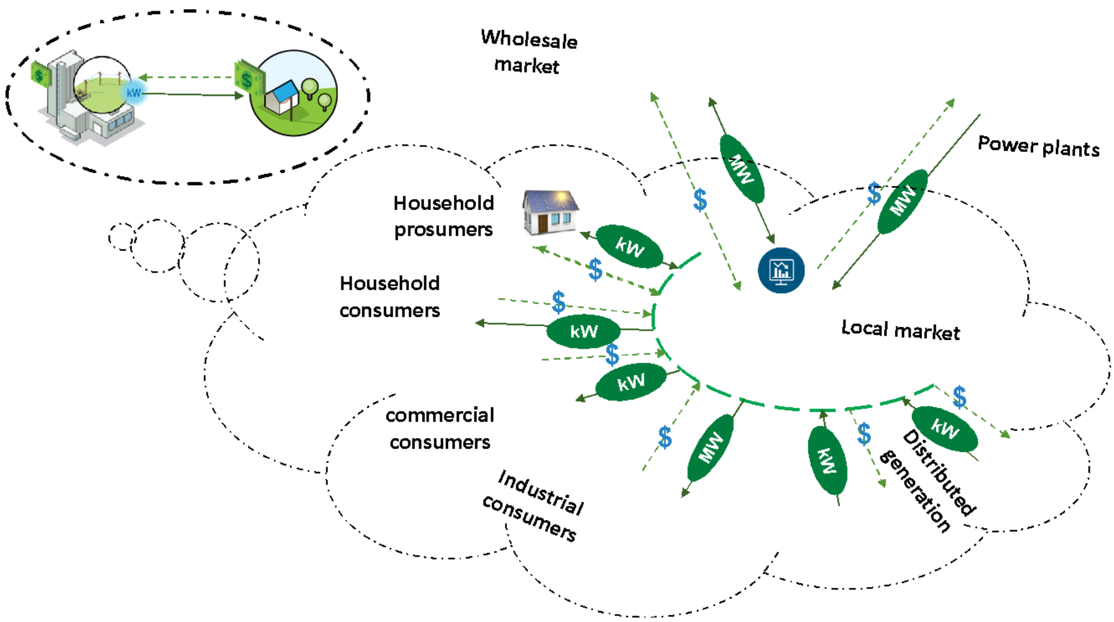

FIGURE 3 – SMART GRID USE CASE, AT A GLANCE

Similarly, the implementation of smart meters can help users acquire new energy-saving habits associated with demand response, in this way lowering their electricity bill. This could contribute as a relief to energy storage, although it is important to note that smart grids have not always been synonymous with improvements in electricity prices.

The case study will characterize the market and know the behaviour of consumers and producers, as well as the trend of prices. The trend is that solar energy auctions will close at lower prices as technology has become cheaper and projects have grown. Renewable generation already has a production cost much lower than the average wholesale market price that is set by marginal thermal technologies, mainly gas. This case study will allow the COSIBAS platform to characterize the energy market through simulation processes using real data to know how producers, consumers or prosumers behave in that market. COSIBAS will allow the establishment of rules to simulate how the application of certain marketing or market regulation rules affects and evaluate the degree of affection towards consumers, producers or prosumers before making decisions in a real city. In addition, it will make it possible to study how the rise or fall in the price of energy affects or the periods of time in which greater energy is generated or consumed.

3.2 Smart Operations and PortCDM in sea traffic management Use Case

Description and goal

This use case will be carried out in Spain and it will be led by Prodevelop. The objective pursued by this use case is to aggregate several sources of data (included but not limited to AIS station, PMS, PCS, etc.) from different stakeholders and analyze them in real-time through AI algorithms and Deep Learning tools in order to improve the overall efficiency of the maritime transport chain. This scenario will be built on both innovative paradigms, STM (Sea Traffic Management) and PortCDM (Port Collaborative Decision Making) as a way of establishing not only a common view of all the available information regarding Port Calls in harbors, but also taking advantage of this data as a good opportunity to create a common awareness supporting the involved actors to take more efficient as well as collaborative decisions. Once implemented and deployed, this use case will enable Port Authorities to better plan arrivals and departures of vessels and improve the allocation of resources (both human resources and machinery/assets). To sum up, the objective pursued by this use case is to optimize the Port stay of a vessel in the harbour by better transparency, richer data exchange and more intelligent solutions assisting vessel port call operational processes. The focus is driven in following areas:

- **Artificial Intelligence (AI) and Machine Learning (ML)** are becoming one of the main factors in a successful digital transformation of the port industry. Larger ports are increasingly becoming aware of the value that is present in a daily collected operational data. The ability to harness operational insights from vast amounts of data that is collected in the ports, will be one of the main advantages of future ports to make them more efficient in terms of energy efficiency, hinterland multimodal transport needs and to better anticipate harmful actions.
- Vessel visit management and port processes control by **ETA prediction and optimization**. Provide all parties involved in a port call with a 'single source of truth' as it relates to vessel ETA by reconciling all available information sources including the vessel pro-forma schedules and real-time vessel position.
- **AIS data** is widely used in maritime domain and is becoming extremely useful for data analytics tasks, especially because of its quantity. In this task we visualize and analyze the data around the ports, provide port congestion indicators out of AIS data and to some extent ETA prediction for the incoming ships.

Automatic Identification System (AIS) was proposed and mandated by IMO (International Maritime Organization) by 2002 SOLAS (International Convention for the Safety of Life at Sea) agreement and its main intention was to prevent collisions on sea, by supplementing additional information, however not replacing existing solutions on board, such as radar and other means that are regulated in COLREG (Convention on the International Regulations for Preventing Collisions at Sea). Since December 31st, 2004 all vessels exceeding 300GT, are obligated to have an AIS transceiver installed and operational, while each has assigned a unique identifier, MMSI (Maritime Mobile Service Identity). Data such as navigational data, information about the ship and voyage related data, is transmitted via VHF (Very High Frequency) radio between ships and shore stations. The range is limited to the VHF range, which is about 10-20 nautical miles but S-AIS (Satellite-based AIS) is available, which can track ships on open seas. Having said that, the AIS data represents a rich source of data about maritime transport and its potential can be utilized to solve a plethora of different problems, in maritime domain, to improve safety, optimize logistics and reduce environmental impacts. In COSIBAS the rationale of the use case will be:

- **<u>ETA prediction from AIS data & other sources of information</u>**: Accurate estimations about estimated time of arrival of the ship is very important, as it has influence on the whole logistics chain, not only on the operations in the port. ETA information is provided in AIS messages but is often unreliable, as it is provided manually by the crew and often not updated. EU ships are requested to send ETA information 72 hours before arrival to the port, which is often too soon and can be quite unreliable. In COSIBAS focus will be on short term

ETA prediction, that is at most a couple of hours distance from the port. This is due to availability of AIS data, for the development and deployment, as it cannot be realistically expected that small and medium sized ports will invest the amounts of money, which is required to obtain live AIS data for a larger area. Another source information to use in this use case is provide from buoy. Buoy provides oceanography information how measure wave height, period and direction, wind speed and direction, air temperature, current profiles at different depths, etc. this information can be reached through Puertos del Estado website/API[53].

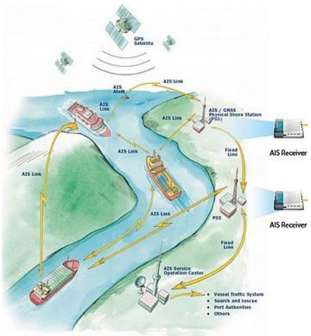

FIGURE 4 – SEA TRAFFIC MANAGEMENT USE CASE, AT A GLANCE

ETA prediction has recently become a very relevant topic in the maritime industry. Port of Rotterdam is one of the most active ports in this field and has also deployed solutions such as ShipTracker[54], which provides ETA estimation out of historical AIS data and other sources (e.g. weather information), with Machine Learning methods. Application was developed internally within Data Science department, which bigger ports can afford to have. ShipTracker tracks all the ships that will arrive to the port in the next 48 hours.

All this information will allow us to calculate an ETA with major precision, whose indirect financial impacts will be (to name only a few):

- Less costs for data and asset maintenance.
- Less discussions by having one single truth about e.g. arrival time.
- More accuracy berth to berth planning.
- More efficiency for terminal operations.
- Less emissions and paperwork.

[53] http://www.puertos.es/es-es (Accessed on February 2020)
[54] https://shiptracker.portofrotterdam.com/#!/ (Accessed on February 2020)

4 Requirements

4.1 Functional requirements

Common functional requirements

This section is aimed at listing the set of common functional requirements collected from the analysis of the different use cases exploited by COSIBAS project. Particularly, such list of common functional requirements is extracted from those scenarios which are common for both use cases. Thereby, common functional requirements are analysed in detail in order to extract relevant information, which could be useful to support the next phases (design, modelling and development) of the lifecycle of COSIBAS.

CFR-01	Register user
The COSIBAS platform will allow the registration of users of the case studies.	
Acceptance criteria	Availability to subscribe users from case studies through and API
Type of requirement:	Functionalities
Product:	COSIBAS Platform
Use case:	UC-0019 (Smart Grid), UC-014 (Smart Operations)

TABLE 1 – CFR-01: REGISTER USER

CFR-02	Unregister user
The COSIBAS platform will allow the cancellation of users of the case studies.	
Acceptance criteria	Availability to unsubscribe users from case studies through an API.
Type of requirement:	Functionalities
Product:	COSIBAS Platform
Use case:	UC-0020 (Smart Grid), UC-015 (Smart Operations)

TABLE 2 – CFR-02: UNREGISTER USER

CFR-03	Modify user
The COSIBAS platform will allow the modification of user information of the case studies.	
Acceptance criteria	Availability to modify user data of the case studies through an API
Type of requirement:	Functionalities
Product:	COSIBAS Platform
Use case:	UC-0021 (Smart Grid), UC-016 (Smart Operations)

TABLE 3 – CFR-03: MODIFY USER

CFR-04	User Login
The COSIBAS platform will allow users to log in to the case studies.	
Acceptance criteria	Availability of logging users on the platform through an API.
Type of requirement:	Functionalities
Product:	COSIBAS Platform
Use case:	UC-0022 (Smart Grid), UC-017 (Smart Operations)

TABLE 4 – CFR-04: USER LOGIN

CFR-05	User Logout
The COSIBAS platform will allow users to log out to the case studies.	
Acceptance criteria	Availability of log-out users on the platform through an API.
Type of requirement:	Functionalities
Product:	COSIBAS Platform
Use case:	UC-0023 (Smart Grid), UC-018 (Smart Operations)

TABLE 5 – CFR-05: USER LOGOUT

CFR-06	Consult historical data
The COSIBAS platform will make it possible to consult the historical data associated with any data analysis process managed by the platform itself.	
Acceptance criteria	Consultation of the data history through an API.
Type of requirement:	Functionalities
Product:	COSIBAS Platform
Use case:	UC-001, UC-002, UC-003, UC-004, UC-010, UC-0011, UC-0012, UC-0013, UC-0026 (Smart Grid), UC-004

TABLE 6 – CFR-06: CONSULT HISTORICAL DATA

CFR-07	Weather prediction
The COSIBAS platform will allow predicting weather conditions through an API for the rest of functionalities in both use cases.	
Acceptance criteria	Availability of an API to predict weather conditions.
Type of requirement:	Functionalities
Product:	COSIBAS Platform
Use case:	UC-0016 (Smart Grid), UC-013 (Smart Operations)

TABLE 7 – CFR-07: WEATHER PREDICTION

Functional requirements from Smart Grid Use Case

This section is aimed at listing the set of functional requirements collected from the analysis of the Smart Grid use case.

SG-FR-01	Prediction Algorithms
The COSIBAS platform will have a set of supervised, unsupervised and semi-supervised algorithms that can be used to perform prediction and learning processes.	
Acceptance criteria	Availability of use of the set of prediction algorithms through an API.
Type of requirement:	Functionalities
Product:	Smart Grid
Use case:	UC-006, UC-007, UC-008, UC-009, UC-025, UC-027

TABLE 8 – SG-FR-01: PREDICTION ALGORITHMS

SG-FR-02	Generate PDF documents
The COSIBAS platform will allow the generation of documents in PDF format for the export of information.	
Acceptance criteria	Availability to export information generated by the platform in PDF through an API.
Type of requirement:	Functionalities
Product:	Smart Grid
Use case:	UC-017, UC-018

TABLE 9 – SG-FR-02: GENERATE PDF DOCUMENTS

SG-FR-03	Negotiation process to sell/purchase energy
The COSIBAS platform will allow users (producers, consumers and prosumers) to sell or purchase energy through a negotiation process established by the system.	
Acceptance criteria	Availability of negotiation processes to sell/purchase energy in the platform through an API.
Type of requirement:	Functionalities
Product:	Smart Grid
Use case:	UC-014, UC-015, UC-024

TABLE 10 – SG-FR-03: NEGOTIATION PROCESSES TO SELL/PURCHASE ENERGY

SG-FR-04	Access to historical data
The COSIBAS platform will need availability and access to historical data of energy production (total energy production data, wind energy production data, thermal energy production data,	

solar energy production/generation data) as a training data set to allow training of the prediction algorithms.	
Acceptance criteria	Availability of historical data of energy production
Type of requirement:	Data requirements
Product:	Smart Grid
Use case:	UC-006, UC-007, UC-008, UC-009, UC-025, UC-027

TABLE 11 – SG-FR-04: ACCESS TO HISTORICAL DATA

REQUIREMENTS BY TYPE

The set of requirements described in the previous paragraph can be classified according to their type, as listed in the following table.

Type	Requirements
Functionalities	• Prediction Algorithms • Generate PDF documents • Negotiation process to sell/purchase energy • Access to historical data
Data Requirements	• Access to historical data

TABLE 12 – FUNCTIONAL REQUIREMENTS FROM SMART GRID USE CASE BY TYPE

FUNCTIONAL REQUIREMENTS FROM SEA TRAFFIC MANAGEMENT USE CASE

This section is aimed at listing the set of functional requirements collected from the analysis of the ETA prediction and PortCDM in sea traffic management use case.

SO-FR-01	Access AIS data
The COSIBAS platform must access the AIS data to obtain information on the vessels.	
Acceptance criteria	Obtain AIS data through an API
Type of requirement:	Functionalities
Product:	ETA prediction and PortCDM in Sea Traffic Management
Use case:	UC-001, UC-002, UC-003, UC-004, UC-005, UC-006

TABLE 13 – SO-FR-01: ACCESS AIS DATA

SO-FR-02	Access oceanographic data
The COSIBAS platform must access the oceanographic data to obtain information maritime climatology	
Acceptance criteria	Obtain information maritime oceanographic data through an API

Type of requirement:	Functionalities
Product:	ETA prediction and PortCDM in Sea Traffic Management
Use case:	UC-007, UC-008, UC-009, UC-010

TABLE 14 – SO-FR-02: ACCESS OCEANOGRAPHIC DATA

SO-FR-03	ETA Prediction Algorithms
The COSIBAS platform will have a set of supervised, unsupervised and semi-supervised algorithms that can be used to perform prediction and learning processes.	
Acceptance criteria	Availability of use of the set of prediction algorithms through an API.
Type of requirement:	Functionalities
Product:	ETA prediction and PortCDM in Sea Traffic Management
Use case:	UC-012, UC-013

TABLE 15 – SO-FR-03: PREDICTION ALGORITHMS

SO-FR-04	Configure system parameters
The COSIBAS platform will allow the user to configure the system parameters related to algorithm prediction.	
Acceptance criteria	Config system parameters through a user interface.
Type of requirement:	Functionalities
Product:	ETA prediction and PortCDM in Sea Traffic Management
Use case:	UC-011

TABLE 16 – SO-FR-04: CONFIGURE SYSTEM PARAMETERS

SO-FR-05	Visualize Port Call and related Operations
The COSIBAS platform will allow the user to visualize the Port Call whose starting point will be the ETA prediction	
Acceptance criteria	Complete Port Call of a specific vessel.
Type of requirement:	Functionalities
Product:	ETA prediction and PortCDM in Sea Traffic Management
Use case:	UC-012

Requirements by type

The set of requirements described in the previous paragraph can be classified according to their type, as listed in the following table.

Type	Requirements
Data Requirements	• Access AIS data • Access oceanographic information data
Functionalities	• Show AIS Data • Show oceanographic information • Prediction Algorithms • Configure system parameters • Visualize Port Call and related operations

TABLE 17 – FUNCTIONAL REQUIREMENTS FROM ETA PREDICTION AND PORTCDM IN SEA TRAFFIC MANAGEMENT USE CASE BY TYPE

4.2 Non-functional requirements

This section is aimed at listing the set of non-functional requirements that the COSIBAS platform should have after analyzing the different use cases and their corresponding scenarios.

NFR-01	Efficiency
COSIBAS should optimize the processing of information with respect to a cost function, e.g. communication, computation, energy, etc.	
Acceptance criteria: Cost functions should be defined, applied and evaluated. Depending on the involved use case a different cost function will be applied and must be defined and evaluated	
Type of requirement: Performance	
Rationale: Optimization of costs	

TABLE 18 – NFR-01: EFFICIENCY

NFR-02	Integrity
Much of the information stored and shared with other systems is critical. Only authorized users can access or modify data and the system must ensure data availability. Backups and procedures to restore the data must be implemented.	
Acceptance criteria: Establishment of security, backup and data recovery mechanisms. Conformity with laws and regulations.	
Type of requirement: Legal/Standard	
Rationale: The system must be protected from unwanted access and ensure data availability.	

TABLE 19 – NFR-02: INTEGRITY

NFR-03	Open source
The European Commission open source software strategy puts a special emphasis on contribution to open source software projects and providing more of the software developed within the Commission as open source. Thereby, software produced by COSIBAS will be open sourced and published	
Acceptance criteria: Source code of COSIBAS is published under an open source license	
Type of requirement: Legal/Standard	

Rationale: The European Commission prefers Open Source projects	

TABLE 20 – NFR-03: OPEN SOURCE

NFR-04	Portability
Ensure that COSIBAS solutions can run in different environments as well as guarantee that system elements may be accessed and may interact from two different environments. Thus, main components of a system can be reused for other tasks	
Acceptance criteria: Standardization of main components	
Type of requirement: Usability	
Rationale: High modularity promotes resilient, high quality, short development systems. Usage and disambiguation	

TABLE 21 – NFR-04: PORTABILITY

NFR-05	Interoperability
Data federation between different systems requires the possibility of exchanging data with unambiguous, shared meaning. Data should be exchanged with explicitly defined formats and semantics. Cognitive services must be able to understand the meaning of the data in order to operate properly. Moreover, the integration with legacy components and systems should be possible.	
Acceptance criteria: Use of standard semantic annotations. The meaning of the data can be understood by all the involved components.	
Type of requirement: Operation maintainability	
Rationale: Semantic Interoperability enables systems to combine received information with other information resources, to process it and to share it in a meaningful manner	

TABLE 22 – NFR-05: INTEROPERABILITY

NFR-06	Documentation
To ensure the usability of COSIBAS, detailed documentation should be made available. A proper documentation is essential to allow future use of the product	
Acceptance criteria: Complete documentation available	
Type of requirement: Operation maintainability	
Rationale: Documentation should be available	

TABLE 23 – NFR-06: DOCUMENTATION

NFR-07	Versioning
The different versions of the components of COSIBAS must be properly identified. Future versions should be compatible with prior versions.	
Acceptance criteria: Versions control	
Type of requirement: Operation maintainability	
Rationale: Management of updates and version upgrades	

TABLE 24 – NFR-07: VERSIONING

NFR-08	Standard APIs

The COSIBAS solution should be able to communicate with other elements through standard APIs to retrieve data. The standard APIs will establish the basis for the creation of new services or functionalities that are compatible with the COSIBAS architecture.	
Acceptance criteria: Definition of standard APIs. Compatibility with standard APIs	
Type of requirement: Operation maintainability	
Rationale: Interoperability	

TABLE 25 – NFR-08: STANDARD APIS

NFR-09	Scalability
The COSIBAS architecture must have an optimal layer structure to maximize their quality of service, while still being scalable, to permit the addition of new elements in them during their exploitation phase	
Acceptance criteria: COSIBAS architecture must be generic and scalable. It needs to be scalable to at least thousands, or even millions of nodes	
Type of requirement: Performance	
Rationale: The system should support the interconnection of a large number of devices and services without degrading its performance	

TABLE 26 – NFR-09: SCALABILITY

NFR-10	Availability
The system should ensure availability. For this reason, the system should be designed to minimize failures and also be able to recover from a failure in a proper amount of time	
Acceptance criteria: Ability to prevent and react to attacks and failures	
Type of requirement: Performance	
Rationale: Maximum reliability possible should be ensured.	

TABLE 27 – NFR-10: AVAILABILITY

NFR-11	Access security
Provide relevant control functions such as access and transport resource control functions: authentication, authorization and accounting (AAA)	
Acceptance criteria: Secured environments can be created for designated users; any communication sensitive of secret data may be connected to IoT. Security levels do not allow third parties to take over control of a private system that is working over the IoT.	
Type of requirement: Security	
Rationale: Conformity with laws and regulations.	

TABLE 28 – NFR-11: ACCESS SECURITY

NFR-12	Security communication between components
Certain sections or sensors/actuators must be securely shielded from the public. This includes the implementation of access control policies and the encryption of the data that is being sent.	

NFR-12	Security communication between components
Acceptance criteria: Devices cannot be accessible for everyone. Access must be controllable for sensors. The network layer must provide with reliable and secure connectivity as required by the pilots	
Type of requirement: Security	
Rationale: Conformity with laws and regulations.	

TABLE 29 – NFR-12: SECURITY COMMUNICATION BETWEEN COMPONENTS

NFR -13	System privacy
Provide privacy protection for accessing information about physical entities, services or platforms connected to or integrated into COSIBAS. Privacy must be supported during data transmission, aggregation, storage and processing. To maintain this privacy, third party access to the private data or into the system is not possible. Also, the identifier or other critical information of a device (e.g., ID of an RFID tag or MAC address of Wireless Sensor) must not be tracked by unauthorized entities. Additionally, the avoidance of the integration and interaction of false nodes/sensors, or unauthorized smart objects must be ensured.	
Acceptance criteria: Security levels do not allow third parties to take over control of a private system that is working over the IoT. The IDs are only sent (and maybe stored) to/in other authorized entities, typically in the same subsystem, without any tracking purposes.	
Type of requirement: Security	
Rationale: Privacy protection	

TABLE 30 – NFR-13: SYSTEM PRIVACY

NFR-14	Services API
A services API must guarantee that the access to the services could be accomplished in a transparent manner from the upper layers point of view. The services API must be properly documented to ensure that it is understandable.	
Acceptance criteria: All exposed COSIBAS services are reachable from the API. The API documentation is complete and available.	
Type of requirement: Look and feel	
Rationale: Creation of new applications and services	

TABLE 31 – NFR-14: SERVICES API

NFR-15	Data provenance
It should be possible to identify the origin of the data	
Acceptance criteria: The system should allow tracking which device or service provides the data	
Type of requirement: Security	

NFR-15	Data provenance
Rationale: Data should be trustworthy	

TABLE 32 – NFR-15: DATA PROVENANCE

NFR-16	Extensibility
COSIBAS should be able to support updates and the inclusion of new modules in an easy way and be able to obtain data from multiple types of sources. The addition of new services and features should be supported.	
Acceptance criteria: The system can be extended with new functionalities, modules or data sources.	
Type of requirement: Operation Maintainability	
Rationale: The system must be capable of retrieving data from multiple sources and support new functionalities and updates.	

TABLE 33 – NFR-16: EXTENSIBILITY

NFR-17	Real-time support
COSIBAS must be able to obtain and process data in real time. The delays in the system output should be short enough to allow real-time operation.	
Acceptance criteria: Real-time data acquisition and processing is supported.	
Type of requirement: Performance	
Rationale: Some applications require real-time data processing.	

TABLE 34 – NFR-17: REAL-TIME SUPPORT

NFR-18	Maintainability
The system should include management functionalities that allow maintaining and updating the functionality. Maintenance operations should require a reasonable effort.	
Acceptance criteria: Management functionalities are available	
Type of requirement: Operation Maintainability	
Rationale: The system must be able to integrate updates	

TABLE 35 – NFR-18: MAINTAINABILITY

NFR-19	Accountability and auditability
Tracking performed operations to the entity (user or device) that generated them must be possible. This implies the existence of audit logs, which must include at least information about the type of operation (including failed login attempts), user, date and sensible data updates. Audit logs must be maintained for a period of six months minimum.	
Acceptance criteria: The information about past operations is available for internal audit.	
Type of requirement: Security	
Rationale: Internal audit	

TABLE 36 – NFR-19: EFFICIENCY

The set of non-functional requirements described in the previous paragraph can be classified according to their type, as listed in the following table.

Type	Requirements
Legal/Standard	Integrity
	Open source
Usability	Portability
Operation Maintainability	Interoperability
	Documentation
	Versioning
	Standard APIs
	Extensibility
	Maintainability
Performance	Scalability
	Availability
	Efficiency
	Real-time support
Security	Access Security
	Security communication between components
	Data provenance
	Accountability and auditability
	System privacy
Look and Feel	Services API

TABLE 37 – NON-FUNCTIONAL REQUIREMENTS BY TYPE

5 CONCLUSIONS

Cognitive Services for IoT based scenarios, COSIBAS, aims to mind the gap between complex IoT architectures and AI services, in both architectural and logical levels through the demonstration of two real use cases (i.e. Smart Grid and Sea Traffic management) where feedback will be gathered and business KPIs measured. The objective pursued by this document is to release the current State of the Art of the technological stack which will help the Consortium to build up the aforementioned solutions as well as to elaborate a list of both functional and non-functional requirements in order to progress with the development stages (i.e. user stories, design, backlog with functionalities, prioritize previous backlog and iterative development stage). Broadly speaking, the consequence of carrying out previous use cases will contribute also to:

- Develop cognitive layers that will support scalability and evolution of large-scale heterogeneous IoT-based systems and autonomously manage changes after deployment. The project will extend the benefits of the IoT concept to industrial applications, to enable interoperability for legacy systems by integrating legacy components and industrial systems into existing IoT platforms and by developing automatic driver generation tools that will facilitate integration of other legacy (and future) systems.

- Deliver intelligence and awareness for enhanced business applications and systems through cognitive enablers and services. These components of cognition will include encoders/decoders, IoT brokers enhanced with knowledge base, IoT planner, IoT human and environment centric monitor and IoT security and privacy risk oracle. On the higher level of the software stack, there will be cognitive services like decision making management, efficient data analysis and inference, and cognitive assistants with adaptive user-specific interfaces.
- Create specific reasoning modules which will enable self-organization in order to provide continuous auto-commissioning capabilities to IoT systems and services, fault forecasting to predict maintenance schedules of IoT devices and platforms, dependability for timely delivering the required quality of service during the entire system's life-cycle and self-adaptation to improve system performance during context changes.
- Create dedicated IoT decision support services and assistants, which will be able to evaluate the current semantic context and suggest possible decisions through cognitive assistants that communicate with maintenance or operator personnel. Harnessing IoT-relevant data from natural language texts can help engineers to access voluminous amounts of relevant data.