PlatformUptake.eu

ASSESSING THE STATE OF THE ART AND SUPPORTING AN EVIDENCE-BASED UPTAKE AND EVOLUTION OF OPEN SERVICE PLATFORMS IN THE ACTIVE AND HEALTHY AGEING DOMAIN

D2.2

European Open Service Platforms in the AHA Domain – Analysis Report

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No 875452.

PROJECT DESCRIPTION

Acronym:	**PlatformUptake.eu**
Title:	**Assessing the State of the Art and Supporting an Evidence-based Uptake and Evolution of Open Service Platforms in the Active and Healthy Ageing Domain**
Coordinator:	SYNYO GmbH
Reference:	875452
Type:	CSA
Program:	HORIZON 2020
Theme:	Healthy ageing, Internet Services & Application
Start:	01 January 2020
Duration:	26 months
Website:	www.platformuptake.eu
E-Mail:	office@platformuptake.eu

Consortium:

SYNYO GmbH, Austria (SYNYO), Coordinator

Institute of science and information technologies, Italy (ISTI)

Fraunhofer Institute For Computer Graphics Research, Germany (IGD)

Universidad Politécnica de Madrid, Spain (UPM)

Institute of Communication and Computer Systems, Greece (ICCS)

Jožef Stefan Institute, Slovenia (JSI)

AFEdemy, Academy on Age-Friendly Environments in Europe BV, Netherlands (AFE)

Cáritas Diocesana de Coimbra, Portugal (CDC)

Linköping University, Sweden (LIU)

Universitat de les Iles Balears, Spain (UIB)

Stitching Smart Homes, Netherlands (SMH)

Etablissementsa Lievens Lanckman, Belgium (LL)

DELIVERABLE DESCRIPTION

Number:	**D2.2**
Title:	**European Open Service Platforms in the AHA Domain – Analysis Report**
Lead beneficiary:	**CDC**
Work package:	WP2 Identify: Success Factors and Monitoring Methodology
Dissemination level:	Public (PU)
Type	Report (R)
Due date:	31.05.2020
Submission date:	07.06.2020

Authors: **Carina Dantas,** CDC
Joana Vieira, CDC
Alexander Nikolov, SYNYO
Andrea Carboni, ISTI

Contributors: **Frederic Lievens**, LL
Willeke van Staalduinen, AFE
Javier Ganzarain, AFE
Theodora Varvarigou, ICCS
Jani Bizjak, JSI
Silvia de los Rios, UPM
Silvia Rus, FNH

Reviewers: **Efstathios Karanastasis**, ICCS
Ad van Berlo, SMH

Acknowledgement: This project has received funding from the European Union's Horizon 2020 Research and Innovation Action under Grant Agreement No 875452.

EXECUTIVE SUMMARY

Within this document, the activities related to Task T2.2 "Observe common and differentiating features and characteristics of existing platforms that can act as success or hindrance factors in their uptake" are described and discussed.

The analysis departs from the ecosystem of platforms built under T2.1 and fully described in D2.1 European Open Service Platforms in the AHA Domain – Ecosystem Map. From a first group of 48 projects identified as some of the most representative in AAL/AHA (Active Assisted Living / Active and Healthy Ageing) research in the last 10 years, 18 were selected to become part of the ecosystem map.

The selection criteria in T2.1 concerned various factors such as the impact on research in the AAL / AHA sector, the European coverage and also other determinants that have led to select platforms that have laid the foundations for the subsequent ones.

Some refined selection criteria were added in T2.2 when performing the in-depth analysis, namely the development timeline and current status of the platforms, as well as their final scope and outputs, and on these grounds a three-layered investigation was performed, as detailed in the methodology section of this document.

Based on this revised list presented in section 1.1, this deliverable aims thus to provide a deeper analysis of eight platforms, to acquire a better understanding on possible success and hindrance factors based on their characteristics, existing networks and stakeholders.

Section 2 provides the three-fold analysis referred: a technical analysis of the platforms, including the description of the features, functionalities and services provided by each of them; a contextual analysis that includes legal, ethical and data concerned information; and a business analysis where the details concerning financial and exploitation aspects are described.

The outcome of these thematic investigations is articulated and combined in section 3, with the aim to understand which were the determining factors that supported or contributed to the current state of art.

This report concludes with a proposal of a scheme displaying the success and hindrance factors of each platform and that will be further elaborated on within the subsequent tasks of the PlatformUptake.eu project.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1. Introduction

The ageing population is the net result of the decline in birth rates and the increase of life expectancy. The improvements in welfare and medical care are contributing to longer lives of the European citizens; however, the increase in the number of years of life is not directly translated into an increase in Healthy Life Years (HLY) [7]. The latter measurement is now recognized by the World Health Organisation (WHO) as the main measure for health care success.

Currently, the main care models implemented to support older adults rely on formal and informal caregivers providing help and care in different living environments, from residential homes to individual houses. This is, though, clearly recognised to be a model with reduced sustainability, as the demographic change pyramid evolves [7].

Technologies have the potential to be essential tools to help older people to Age in Place, in a healthy and independent way, enhancing socialization and autonomous routines for as long as possible. Likewise, they also have a high potential for making caregivers' work easier and more effective in providing social and health care, facilitating their usual tasks, better managing the situation, their routines and relieving their burden as well as the stress symptoms associated to their work that make them feel depressed, emotionally weakened and isolated.

From a business perspective, technology has much to offer to, what is generally considered, the Silver Economy Market by opening new pathways to a more preventive and empowering approach to healthy ageing. Recognising this, a huge investment is being done at the European level to promote the Digital Single Market Strategy and the implementation of digital tools for health and care.

Despite the investment in several projects and platforms, there is still a gap in accomplishing broader implementation and uptake. In the specific area of open service platforms in the Active and Healthy Ageing (AHA) domain, the investment started around 2005 and it is essential to uncover which are the fruits and the lessons to be learned.

Thus, the present report intends to provide a more in-depth analysis of such platforms to acquire a better understanding on possible success and hindrance factors, by delivering the description, features, functionalities and services provided by each of the platforms (technical aspects) as well as legal and financial aspects, describe their development timeline and shed light on possible factors that have supported or contributed to the current state-of-the-art.

2. Methodology

The starting point for the work in T2.2 was the ecosystem map derived from D2.1 and the different selection criteria and descriptions used to reach that final platform list.

Departing from this map, the present analysis derived from the task description in the DoA and on desk research and discussion around the different methodologies portrayed in the PlatformUptake.eu proposal as inspirational examples:

- MAST | The Model for the Assessment of Telemedicine - is built following six perspectives of assessment ((1) health problem and characteristics of the application, (2) safety; (3) clinical effectiveness; (4) patient perspectives; (5) economic aspects; (6) organizational aspects; and (7) socio-cultural, ethical and legal aspects).
- OPEA: Open Platform Ecosystem Assessment Framework - is a three-dimensional model. The first axis includes the value network of the AAL platform provider, AAL application provider, Health Service or Social Service provider, the informal carers, assisted persons and society. The second axis marks the assessment domains of the evaluation: assistance problem and characteristics of the open platform & applications, technical aspects, user perceptions, outcomes, economic aspects, organizational aspects, and contextual aspects. The third axis relates to the three levels of assessing the AAL ecosystem: the platform, application, and service level.
- GLocal Evaluation Framework - is the ACTIVAGE reference evaluation framework for AHA Large Scale European pilots.
- Market Intelligence - also known as business intelligence, it provides several methods to analyse the platforms' maturity and business models:
    - Business Model Canvas, to analyse existing providers' business models.
    - ADL Matrix, for understanding how an industry's maturity and competitive position affects strategy, in terms of industry maturity (from embryonic to aging) and competitive position (from dominant to weak).

By analysing and discussing these different methodologies, the main features that were relevant for this task were extracted and a framework with the relevant areas and indicators was agreed on, combining three dimensions:

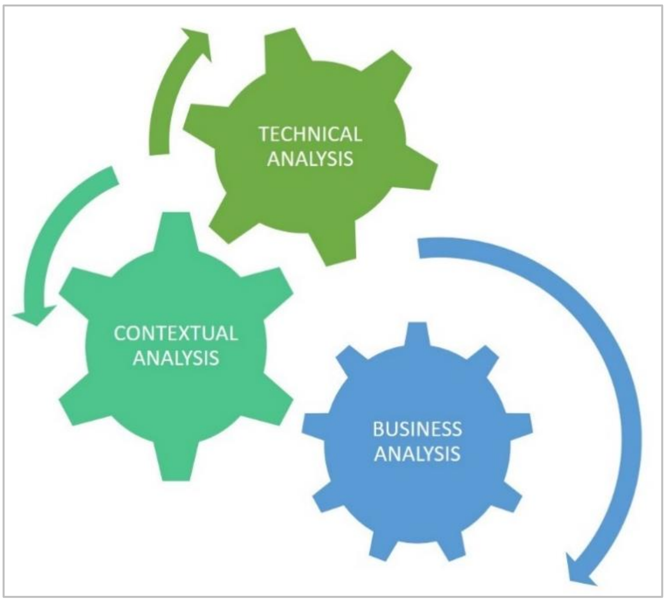

Figure 1: Three layers of in-depth analysis of the open platforms in AHA

Each of these dimensions encompasses a dedicated approach and several enclosed aspects, which are detailed and explained inside each of the three sections. Besides a detailed description of each platform in these areas, some comparative figures are drawn at the end of the section summarising and illustrating the main similarities and differences between them.

After the collection and investigation of these data, a comparative analysis is drawn in the last chapter of the report, inspired by the Critical Success Factors methods. Starting by a collection, through desk research, of the main success and hindrance factors of digital platforms reported in literature, a comprehensive table of the stakeholders, dimensions and factors is proposed, that may be used as the departure point for the understanding of missing points and to frame the proposal KPIs in the subsequent tasks.

2.1. Projects list

During Task 2.1 "Map the ecosystem by collecting existing open-source platforms in the Active and Healthy Aging Domain, their end-users and related stakeholders", more than thirty projects and platforms were initially identified and selected and later on, reduced to eighteen (Table 1), as follows:

Table 1: The selected projects in D2.1

ID	Project name	Project logo	ID	Project name	Project logo
1	ACTIVAGE		10	M-Power	
2	Amigo		11	Oasis	
3	AmIVital		12	Persona	
4	BeyondSios		13	Reach2020	
5	EkoSmart		14	ReAAL	
6	FiWare		15	Soprano	
7	Giraff+		16	UNCAP	
8	In Life		17	universAAL	
9	InterIoT		18	VAALID	

These projects were discussed and analysed within the scope of D2.2. The main outcome of the preliminary search was that some of the older platforms, inactive for many years, were not suitable candidates for any analysis that concerns business or legal aspects.

Therefore, as a first criteria, the timeline of development of the different platforms built for D2.1 (figure 1) was followed to discard from a thorough analysis in this report all the platforms that fall under the blue period, that is, before 2010.

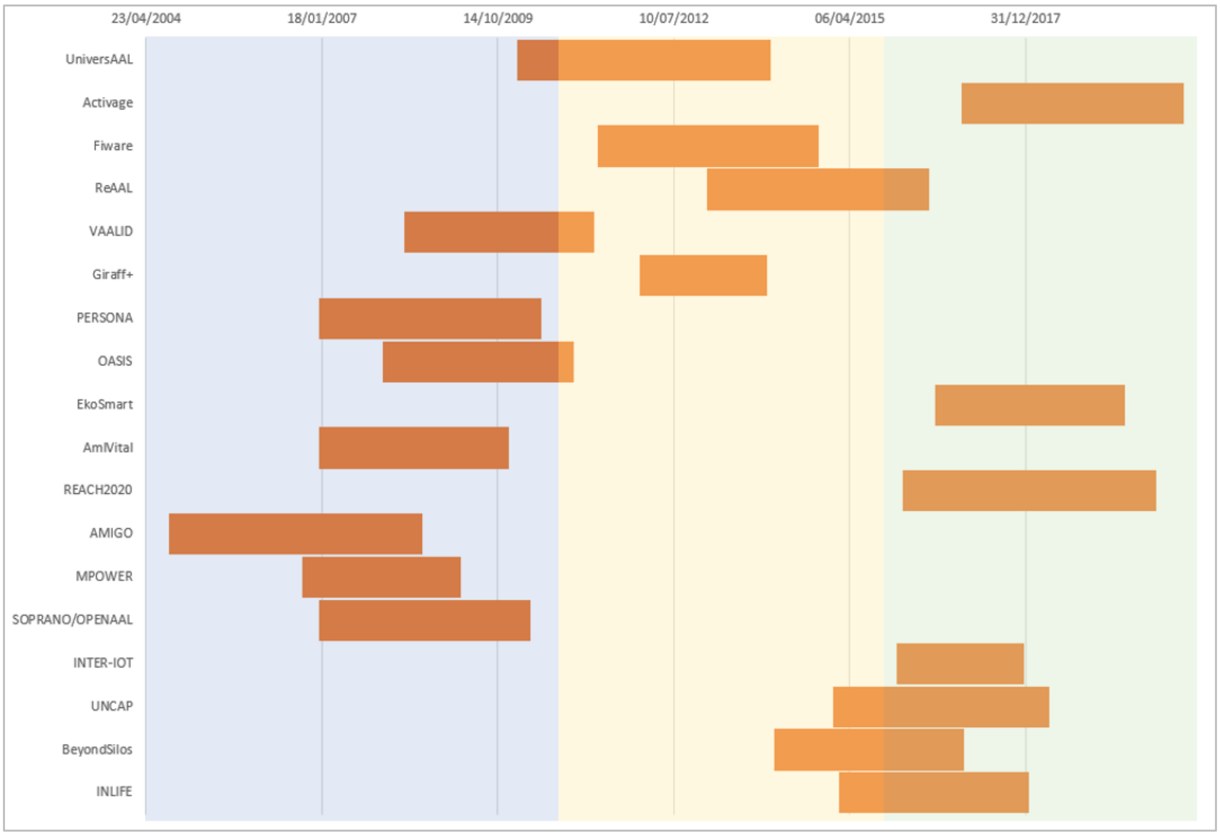

Figure 2: Development timeline of the platforms analyzed in D2.1

Despite this, the projects that were predecessors of UniversAAL IoT and Activage AIOTES were considered, when and where relevant, for the technical analysis, as this evolutionary pathway could be one of the factors to influence its current existence and therefore be considered for the success factors to be concluded.

On the other hand, during the analysis undergone for T2.2, some platforms not previously uncovered in the search undergone for the previous task were brought to the list. From these, a few were discarded (e.g. IoTivity, SeniorSome) and two of them integrated the core group to be analysed in depth: Onesait and Sensinact.

The final list and the summary of the inclusion/exclusion criteria is presented in Figure 3.

<table>
<tr><th>PLATFORMS TO BE ANALYSED IN FULL</th><th>PROJECTS THAT LED TO MORE RECENT ONES</th><th>ONLY SUMMARISED WITH CRITERIA FOR EACH GROUP</th></tr>
<tr><td>UniversAAL</td><td>Mpower | Soprano
Persona | Amigo | Oasis</td><td>Out of Europe:
- IoTvity</td></tr>
<tr><td>Activage_AIOTES</td><td>InterIoT
ReAAL</td><td rowspan="3">Platforms already closed:
- AmIVital
- Inlife
- Vaalid
- SeniorSome</td></tr>
<tr><td>Ekosmart</td><td rowspan="6"></td></tr>
<tr><td>Reach 2020</td></tr>
<tr><td>Sensinact</td><td rowspan="2">Out of scope:
- Beyond Silos
- OpenThings</td></tr>
<tr><td>UNCAP</td></tr>
<tr><td>FiWare</td><td rowspan="2">Only a project:
- Giraff</td></tr>
<tr><td>Onesait</td></tr>
</table>

Figure 3: Summary of the platforms analyzed in D2.2

For the in-depth analysis of these platforms, a template was built and distributed by the consortium partners to collect the information needed in the three areas:

Technical analysis

Table 2: Questionnaire for the technical analysis

TECHNICAL ANALYSIS
IN DEPTH PLATFORM ANALYSIS
PHYSICAL LAYER Analysis of the solutions structure and implementation
SERVICE LAYER Analysis of the solutions structure and implementation
APPLICATION LAYER Analysis of the solutions structure and implementation
SEMANTIC LAYER Analysis of the solutions structure and implementation
INTEROPERABILITY LAYER Analysis of the solutions structure and implementation
FEATURES ANALYSIS
DEVICE MANAGEMENT The IoT platform should maintain a list of devices connected to it and track their operation status; it should be able to handle configuration, firmware (or any other software) updates and provide device-level error reporting and error handling. At the end of the day, users of the devices should be able to get individual device level statistics.
INTEGRATION/INTEROPERABILITY The API should provide access to the important operations and data that needs to be exposed from the IoT platform. It's common to use REST APIs to achieve this aim.
INFORMATION SECURITY Measures required to operate an IoT software platform are much higher than general software applications and services. Millions of devices being connected with an IoT platform means we need to anticipate a proportional number of vulnerabilities. Generally, the network connection between the IoT devices and the IoT software platform would need to be encrypted with a strong encryption mechanism to avoid potential eavesdropping.
TYPES OF PROTOCOLS An IoT platform may need to be scaled to several (up to millions or even billions) devices. Lightweight communication protocols should be used to enable low energy use as well as low network bandwidth functionality.
DATA ANALYTICS The data collected from the sensors connected to an IoT platform needs to be analyzed in an intelligent manner in order to obtain meaningful insights. There are four main types of analytics which can be conducted on IoT data: real-time, batch, predictive, and interactive analytics.
SUPPORT FOR VISUALIZATION Typically referred as visual interfaces, they can be simple web portals with some kind of visualization of the system, its components and the data, they can allow for the management of Iot Ecosystems and, in optimal solutions, provide the capabilities of visual data analytics.

Contextual analysis

Table 3: Questionnaire for the Contextual Analysis

CONTEXTUAL ANALYSIS
REGULATORY FRAMEWORK
In which country is it based
Is there information available on the regulations it complies with? (e.g. link to deliverables)
Is it a medical device? Is it certified? CE mark? Other?
Is there information available on how it was funded? If the services are reimbursed, if it was funded through procurement, projects, etc.?
ETHICS AND PRIVACY
Type of data collected
Information provided to the user on data collection, storage, processing and transfer
Is there an informed consent?
DATA SHARING GOVERNANCE
Which model of data sharing does it use?
How is data management ensured?
IPR
Is the platform registered - brand, trademark, patent, etc.
What is the access model? Open access, open source, close access

Business analysis

Table 4: Questionnaire for the business analysis

BUSINESS ANALYSIS
KEY PARTNERS
Suppliers, financiers, contractors, and marketing firms. Here it is important to mention if the creators of the open source platforms have used resources from external parties or outsourced certain activities. A list of key partners can be also added
KEY ACTIVITIES
Activities needed to create value, achieve benefits for the customers/end users and deliver successful services. The key activities are linked to the value proposition and key resources. Some sample activities might be marketing, distribution, research and development, customer service, revenue streams etc.
KEY RESOURCES
Key Resources looks at the staff, the processes, available money and equipment or applications needed to create the value for the customers/ end users of the open source platforms in AHA and AAL domains.
VALUE PROPOSITION
It defines the services supplied to the customers/ end users. The term "value" refers to the newness, performance, design, accessibility, etc. that the customer perceives.
CUSOMER SEGMENT
All the important (paying) customers /end-users or organizations for which the business model wants to create value need to be defined.
CHANNELS
Channels look at how the offerings/services/products can get to the customers and through what preferred channels.
CUSTOMER RELATIONSHIP

Customer Relationship focuses on getting, keeping, and growing the customer base (marketing communications, sales support, technical assistance and customer service).
COSTS
The costs for the creation of an open source platform need to be estimated and the money needed to get the business to a stage where it's providing the desired profit.
REVENUE STREAMS
Revenue streams, one focuses on how the customer pays for the provided value. Some examples are subscriptions, rentals, service sale, and asset sale.

The different platforms were distributed by the consortium partners for the information collection and the analysis of the three areas were firstly allocated to a leader: ISTI for the technical aspects, CDC for the contextual and SYNYO for the business and financial. CDC and AFE led the conclusions sections, uncovering the main success and hindrance factors.

Finally to foster general understanding on the terminology being applied in the course of the analysis, the glossary set out in D2.1 European Open Service Platforms in the AHA Domain – Ecosystem Map can be found in the Annex part of this document.

3. Analysis of technical aspects

As explained in Section 1, this chapter will describe the framework that was defined to analyse the technical aspects of the eight open platforms selected. With this framework as the overarching context, a detailed analysis is performed in section 2.2, presenting for each an introductory text detailing the layers, services and functionalities and a figure with the main aspects collected. In section 2.3 a collection of pictures summarises and compares the main aspects of each of the platforms in five different areas.

3.1. Definition of the technical framework and areas

With reference to the description of task 2.2, it is stated that "The report coming out from this task will involve the description of the features, functionalities and services provided by each of the platforms (technical aspects)". In the deliverable D2.1 five layers have been defined from the technical point of view of the developers; so, it is therefore ideal to continue starting from that angle.

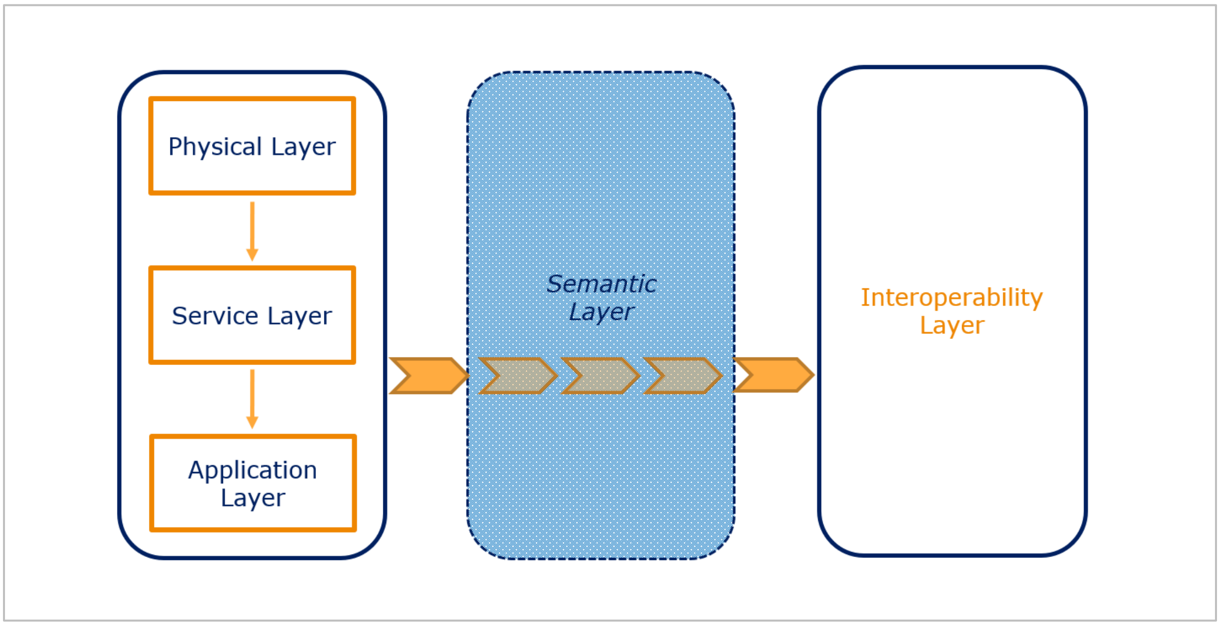

Figure 4: Platform layers from a developer point of view

In addition to this type of level-based description, six significant features for an IoT system have been identified:

- device management
- integration/interoperability
- security
- data analytics
- support for visualization

The description of the features, seen as the most significant implementation and operational aspects of the various platforms examined, allows to facilitate the understanding of the differences between the various platforms and also their schematization. In fact, as will be shown below, each type of feature will be accompanied by a comparative table between the various platforms in reference to the particular feature taken into consideration.

Features specification are specific for an IoT system and are defined as follows:

- **Device management:** the IoT platform should maintain a list of devices connected to it and track their operation status; it should be able to handle configuration, firmware (or any other software) updates and provide device-level error reporting and error handling. At the end of the day, users of the devices should be able to get individual device level statistics.
- **Integration/Interoperability:** the API should provide access to the important operations and data that needs to be exposed from the IoT platform. It is common to use REST APIs to achieve this aim.
- **Information security:** measures required to operate an IoT software platform are much higher than general software applications and services. Millions of devices being connected with an IoT platform means we need to anticipate a proportional number of vulnerabilities. Generally, the network connection between the IoT devices and the IoT software platform would need to be encrypted with a strong encryption mechanism to avoid potential eavesdropping.
- **Types of Protocols:** an IoT platform may need to be scaled to several (up to millions or even billions) devices. Lightweight communication protocols should be used to enable low energy use, as well as low network bandwidth functionality.
- **Data analytics**: the data collected from the sensors connected to an IoT platform needs to be analysed in an intelligent manner in order to obtain meaningful insights. There are four main types of analytics which can be conducted on IoT data: real-time, batch, predictive, and interactive analytics.
- **Support for visualization:** typically referred as visual interfaces, they can be simple web portals with some kind of visualization of the system, its components and the data, they can allow for the management of IoT Ecosystems and, in optimal solutions, provide the capabilities of visual data analytics.

The features can, in some ways, be seen as a summary of the main features of an IoT platform. To make the understanding of the features more immediate, five specific characteristics of each feature have been identified, which will be presented initially in the form of a table relating to the single platform and subsequently as a comparison of the solutions adopted by each of them. *Types of Protocols* feature is not included as a table but is discussed within each paragraph. In platforms where the analysis of the features has led to significant contents that cannot be limited to a table, a paragraph "Features notes" will be added to the description.

3.2. Technical analysis of the platforms

In this paragraph, the technical analysis of each of the platforms is described, consisting of two main parts: a first descriptive part where the solutions adopted for the implementation of the layers, whenever possible, are described in detail; and a second part that focuses on features and provides some details on those.

3.2.1. UniversAAL IoT

The *Physical layer* of universAAL contains sensors and actuators, as well as intermediary hubs, which ease the connection between them. In the universAAL world, a single device that is connected to an assistive system is referred to as being a "node".

In principle, there are two ways of how to integrate a device into an universAAL-based assistive system, assuming that the device in question is networked (can send and receive data using a network protocol, either wired or wireless). The first way is to install a specific piece of the universAAL platform on the device, the so-called "Middleware". The Middleware software contains the communication infrastructure of the universAAL platform and all devices that run the Middleware can actively participate in the communication of the system. The second way of connecting devices to an universAAL-based assistive system does not require a given device to run the universAAL Middleware.

The device in question is rather connected to a node that runs this Middleware, and this node is used as an intermediary by the system in order to control the additional device. For many devices, such as low-power wireless sensors, this is the only way of connecting them to the system, simply because they cannot run any additional software beyond their firmware. And although these slave-devices cannot actively participate in the communication with the rest of the system (as they are just queried for data), their advantage over regular nodes is that they can (oftentimes) simply be "plug-and-played" into a running system.

Regarding hardware sensor and actuator devices, these are connected through “exporters”, which are just like an application exporting the devices interfaces and information into universAAL platform. There would be a different exporter per technology (KNX, ZigBee, etc.).

Additionally, to the physical devices and the “exporters”, the physical layer of universAAL is composed of three pieces of software, serving specific purposes. These three buses form the heart of the universAAL platform. All communication between universAAL-based applications should happen only in a round-about way via one of them, even if physically, these applications are located on the same node (= are running on the same device). Each of the buses handles a specific type of message/request, and the way that a bus operates is based on the characteristics of this category of information. The three buses are: Context, Service and UI (User Interface) bus.

The Context Bus is used for publishing information about the state of the environment and/or the assistive system. On the Service Bus, an application that offers a service (= can do something) announces this by registering a corresponding service profile, that is a description of what it is capable of doing, with the Service Bus. The counterpart to this are applications that require a service, the "service callers". They send a service request to the Service Bus, asking for a specific service (as in "I need someone to turn off the lights in the living room, please"). It is up to the Service Bus to then find one or more matching service profiles to the service request and, if a match is available, to forward the request to the corresponding service. The purpose of the UI Bus is to deliver messages that are somehow related to explicit user interaction. For example, an application that wants the user to be notified about a certain event would use the UI Bus. There is actually a fourth bus, the Control Bus, which is responsible for managing the nodes in an uSpace (group of nodes), discovery of new nodes, and deployment of software artifacts.

The *Middleware* does not provide functionality that an end-user would find very useful. It is rather the basis for the higher level platform components and the functionality providing applications. The Middleware is capable of hiding the distribution and heterogeneity of the diverse devices that make up the system at its core.

The *Service layer* of universAAL is composed of different "Managers", which build upon the Middleware. They can be considered low level applications. Together with the Middleware they form the universAAL execution platform and are required for its proper operation. Some are tied to certain buses, while others are more widely used. They usually also provide functional APIs to the above final applications.

On the *Application layer*, the universAAL platform is not limited to a specific application. Any application can be created to interact with the heterogeneity of devices it is connected to. One understands an application - any piece of software - that can run on the Container and that makes use of the universAAL Buses or Managers, whether by consuming them or providing into them, in order to provide a service or a part of it. An application, in addition to its own business logic, and regardless of its structure, needs one or more of the universAAL "wrappers" presented until now: Context Publisher, Context Subscriber, Service Caller, Service Callee and User Interaction Caller. Each of this must be created at some point during the application execution, at which they will be connected to universAAL. When the application stops, these must be closed.

The *Semantic layer* of universAAL uses an approach called "goal-based interoperability". It is based on the principle idea to formulate requests in a semantical and not in a syntactical way, thus stating what is supposed to be done - the "goals" that are to be achieved - rather than how this should be done (which would include the specification of an addressee). The universAAL-platform achieves this through the use of ontological descriptions. The task of finding the appropriate recipient for a message is then left to a mediator that needs to know of all possible recipients that are currently available and must be able to decide, which one of them (if any) is the right one for this specific request. On the downside, however, this means that applications also require well-thought-out strategies for fault tolerance as there can be no guarantees that the dynamic resolution of dependencies through the mediators will actually be successful. In case of failure, when a mediator cannot deliver a message, an application should not simply crash but rather adjust its functionality to this situation, for instance by suspending its execution until a suitable recipient is available.

FEATURES NOTES

UniversAAL IoT has the ability to connect to heterogeneous devices. It offers the possibility to build applications with which to access, control and maintain the devices. These are however not a direct part of the platform. Some elementary functionality is offered for device lifetime management and auto diagnostic features. An application to see device level statistics could be built for universAAL but is not directly a part of it.

Due to the distributed nature of universAAL, integration is present at all layers. It is implemented using a semantic approach. It offers device and service interoperability. The interoperability also offers the according security. Security mechanisms are implemented in universAAL at all layers.

Currently universAAL supports devices using protocols such as KNX, Continua devices (IEEE-11073 via Bluetooth), ZigBee, EnOcean, FS20, MQTT and Modbus.

The universAAL platform does not include the applications on top of it. However, it offers the services enabling to build such applications which target data analytics, enabling real-time, batch, predictive, and interactive analytics. The same rationale applies to visual interfaces for visual data analytics. Specific solutions were available in the uStore and further extended in ReAAL and Activage projects.

TECHNICAL ANALYSIS				
universAAL IoT				
DEVICE MANAGEMENT				
Connectivity of heterogeneous IoT devices	Remote access to IoT devices	Remote control of IoT devices	Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Autodiagnostic features
Yes	Yes	Yes	No	Yes
INTEGRATION/INTEROPERABILITY				
Interoperability is implemented using a semantic approach	Interoperability is implemented using a syntactic approach	Implements mechanisms able to make available platform services to third parties outside the Platform	Implements Restful / SOA web services mechanisms to access interoperability feature	Implements interoperability between services and functions defined inside the Platform
Yes	No	Yes	Yes	Yes
INFORMATION SECURITY				
No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Data is encrypted on the network	Communications between Platform to the Internet are secured	Protocols and cryptographic schemes ensure end-to-end data integrity	Only authorized devices can be connected to the Platform
Yes	Yes	Partial	Yes	Yes
DATA ANALYTICS				
Implements real-time data analytics	Implements predictive data analytics	Implements data analytics for anomaly detection	Data analytics are included in a Marketplace	Data analytics are accessible using REST/SOA API calls
No	No	No	No	No
VISUAL INTERFACES				
Visualization of data	Manipulation of data	Creation of analytics	Dashboard available for each user group (caregiver, patient, etc.)	Web application or standalone
Yes	Yes	Yes	Yes	STANDALONE

Figure 5: Technical analysis summary of UniversAAL IoT

3.2.2. Activage_AIOTES

The *Physical layer* is in charge of performing sensing information from the Physical world and actuating to modify the behaviour of the connected objects. The main building blocks constituting this layer are:

- Sensor/Actuator Node: a sensor/actuator node is the device interfacing with the physical world. This node or end-device can be a device sensing data (sensor) from the Physical word or a device controlling the parameters of the devices (actuator) interacting with the Physical world. Both of them have a connectivity element (in general wireless transceivers supporting BT/BLE, Zigbee, ZWave, … protocols) and support different use cases:
    - Home automation monitoring (i.e., detectors of CO, gas, smoke flood, door contact, presence, temperature, and humidity).

  - Home Monitoring Clinical Data (i.e., Blood pressure, blood sugar, heart rate, weighting scale).
  - Home automation control (i.e., blind actuator, heating actuator, Wall Plug, light actuator).

- Tracker: it is a device in charge of acquiring the position of an object at a given time.
- Tags: they are used to identify some activities or behaviours. These tags can be based on different technologies such as Near-field-communication (NFC), quick response (QR) code, and Bluetooth low energy (BLE).
- Connectivity Devices: they are communication devices such as Aggregation points and Gateways, and the communication infrastructure (for instance, the switches, routers, firewalls, proxies needed to connect the Device layer to the Internet).

Each DS of the project uses its protocols and communication technologies to interconnect devices. However, most DS has chosen mainly wireless connectivity devices such as Bluetooth, BLE, ZigBee, Z-wave, Wi-Fi, and wired communication.

The *Service layer* is composed mainly of three components:

- Data Lake: it offers an intermediate component between the ACTIVAGE core of the semantic interoperability layer and high-level analytic components and end-users. The Data Lake allows the search and retrieval of IoT collected data, integrating the information gathered by all IoT platforms such as row data and extracted features and intermediate analysis results. The Data Lake exposes an API through which end-users and other components can have access to its functionalities;
- The AIoTES Data Analytics methods: provide mechanisms to extract high-level information and knowledge from the raw data, which can be used by users and applications through web services and provide functionalities including feature extraction and selection, anomaly detection, classification, prediction, clustering, and hypothesis testing;
- AIoTES information visualization methods: provide the end-user with means to explore large volumes of collected data in intuitive visual representations, in order to detect groups and patterns, and subsequently focus on the desired level of detail. Data analytics and visual analytics provide the user with aggregate information or extracted features, hiding details about the raw data, thus avoiding private data exposure.

In the *Application layer*, applications are stored, managed, and organized using a Marketplace: a shop for developers to provide and monetize their applications, and for end-users to discover, obtain, and deploy them. Marketplace implements most functions for users to browse, search, register, and install applications, keep a wish list, and get overall insights. Marketplace provides semantic search, recommendations with the AHA Advisor, Semantic Interoperability Layer capabilities, compliance with the Security and Privacy policy. Although there are many different needs and requirements related Applications specifically for each Deployment Site, applications can belong to the following categories:

- Efficient monitoring of the health status and activity of the elderly;

- Efficient monitoring of environmental parameters;
- Presentation of the information in a personalized, accessible, usable, and intuitive manner;
- Enable online data storage by also ensuring security and privacy.

Each DS in the project has developed its applications and analytics to monitor patients in AHA domain. In general, the analytics on IoT data used in the project are of a different type: real-time, batch, predictive, and interactive. According to its created applications, each DS has developed its human interfaces that are designed for desktop, web, and smart devices like smartphones, tablets, and smart TV devices.

The *Semantic layer* is used to implement IoT platform interoperability and for data modelling. ACTIVAGE subdivides the work of data modelling into several domain areas:

- IoT platforms/systems and services where sensors and sensor measurements are the focus, Active and Healthy Ageing (AHA) services where end-user applications are the focus;
- Data Security and Privacy systems and services where the IoT platforms/systems device protection and access control and data protection and privacy preservation are the focus aspects;
- Healthcare Information Systems Support.

The *Semantic interoperability layer* is the core element responsible for enabling semantic interoperability between all the heterogeneous IoT platforms and an Open Framework, and additionally ensures security, privacy, and secure data. To implement interoperability, the AIoTES architecture uses mainly two components of the Semantic Interoperability Layer:

- Interoperability Layer: Interoperability solution at middleware level that enables syntactic interoperability among IoT platforms and AIoTES. This component provides an abstraction level at the middleware;
- IPSM: interoperability solution at middleware level that enables semantic interoperability among IoT platforms and AIoTES. This component provides an abstraction level at the middleware. Data is sent through the Interoperability Layer using messages expressed in the common syntactic format. Each message consists of two RDF graphs (metadata and payload). The metadata graph contains information for routing and processing the message, which includes the identifier of the message, the identifier of the sender/receiver platform, the type of message, and the conversation identifier, which identifies a group of messages and allows matching a response with its corresponding request. The payload graph contains the actual data that is being sent. IPSM can perform ontology-to-ontology data translation among two different platforms that employ different ontologies or data models. As a result, one platform can receive data from another platform understanding the semantic meaning of this data.

During the project development, a security analysis activity was performed for Device, Gateway, Cloud, and Application levels. This analysis considers technical details as communication links, storage capabilities, Trust boundaries, etc. to identify and describe each potential Cybersecurity Threats attacks (S-T-R-I-D-E) and then to find the solutions to foil them.

FEATURES NOTES

The main building blocks constituting the Device layer in the ACTIVAGE architecture are the following: Sensor/Actuator Node, Tracker, Tags, Aggregation Point and Gateway. A sensor/actuator node is the device interfacing with the Physical world. A tracker is a device in charge of acquiring the position of an object at a given time. Tags are various forms of electronic tags can be used, for instance, at home in the ACTIVAGE Deployment sites in order to identify some kind of activities or behavioural. Aggregation point is a device used to aggregate /regroup several network ports/links into only one with the goal to extend the communication reach. A Gateway is either a server with a gateway application installed or a device that connects a network of computers to another network and more specifically with Internet network (via wired (ADSL, optical fibre, etc.) or wireless channels (3G/4G, etc.).

The project integrates several heterogeneous platforms and deployment sites using different standards, data formats and semantics making them unable to interoperate and share data. Integration is created using an abstraction layer between the IoT platforms from DS and the ACTIVAGE system that provides both syntactic and semantic interoperability. This is done using specific REST API.

The security analysis was performed for Device, Gateway, Cloud and Application levels. This analysis considers technical details as communication links, storage capabilities, Trust boundaries, etc. with the goal to identify and describe each potential Cybersecurity Threats attacks (S-T-R-I-D-E) and then to find the solutions to foil them.

Each DS in the project has developed its own analytics to monitor patients n AHA domain. In general, the analytics on IoT data used in the project are of different type: real-time, batch, predictive and interactive.

Each DS, according to its created applications, has developed its own human interfaces that are designed for desktop, web and smart devices like smartphone, tablet and smart TV devices.

TECHNICAL ANALYSIS

ACTIVAGE PROJECT

DEVICE MANAGEMENT

Connectivity of heterogeneous IoT devices	Remote access to IoT devices	Remote control of IoT devices	Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Autodiagnostic features
Yes	Yes	Yes	Yes	Yes

INTEGRATION/INTEROPERABILITY

Interoperability is implemented using a semantic approach	Interoperability is implemented using a syntactic approach	Implements mechanisms able to make available platform services to third parties outside the Platform	Implements Restful / SOA web services mechanisms to access interoperability feature	Implements interoperability between services and functions defined inside the Platform
Yes	Yes	Yes	No	Yes

INFORMATION SECURITY

No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Data is encrypted on the network	Communications between Platform to the Internet are secured	Protocols and cryptographic schemes ensure end-to-end data integrity	Only authorized devices can be connected to the Platform
Yes	Yes	Yes	Partial	Yes

DATA ANALYTICS

Implements real-time data analytics	Implements predictive data analytics	Implements data analytics for anomaly detection	Data analytics are included in a Marketplace	Data analytics are accessible using REST/SOA API calls
Yes	Yes	Yes	Yes	Yes

VISUAL INTERFACES

Visualization of data	Manipulation of data	Creation of analytics	Dashboard available for each user group (caregiver, patient, etc.)	Web application or standalone
Yes	Yes	Yes	Yes	BOTH

Figure 6: The technical analysis of Activage AIOTES

3.2.3. Ekosmart

The EkoSmart platform collects, combines and connects different services, devices, service providers and users in a common platform. While the platform does not provide any specific devices or services it allows other devices, services and products to be added and interconnected through it. It serves both as a marketplace for services and products as well as hub for easy interconnection, extension and integration of existing devices and products. While the EkoSmart platform does not provide any specific device on the *Physical layer* it allows each service provider to add multiple physical devices that connect directly to the platform, without the need of an intermediate application. The data sent by the device is stored on the server and visualised out of the box, the platform allows the devices to be configured through the platform, if the device supports over the air configuration. The physical device can connect to the platform with secure or insecure HTTP connection, which is based on REST API. For each device metadata, that describes the device, must be provided by the device provider. The metadata contains information such as device unique ID, device type, data type (numerical or text), visibility (public, private).

Furthermore, for each device several alarms can be added through the application layer, alarms are notifications that are triggered by a specific set of rules based on arithmetical and logical operations. There is no limit which devices can connect to the platform; the only perquisite is that the device provider obtains authentication token for the device type. Currently the following devices are connected: smart watch, temperature sensor, GPS tracker (for tracking and showing city buses), blood pressure monitor.

The *Service layer*[1] is based on WSO2 API Manager framework, which makes the communication between different services, devices and applications through API calls easier to implement and more reliable. Each service or device connected to the platform must provide metadata at registration which defines type of communication (Custom API, SOAP, REST or Websocket) and description of data provided. The data provided by the devices and services can be stored either externally (device provider) or internally on the platform. The storage is implemented with CKAN[2] open source framework for data management. The data stored on the platform can be either in (JSON, XML, CSV) format.

The services and devices can publicly expose their API calls, or parts of it, which allows other partners to integrate and interconnect their services. Depending on where the data is stored, connection can go through the *Interoperability layer* or directly to device/service provider.

The main purpose of the *Application layer* on the platform is to provide users and service providers with one place for providing and subscribing to services, products and devices in the domain of e-health and smart cities. While each device or service can provide its own separate application that can be downloaded through the marketplace part of the platform, service providers are encouraged to use data visualisation tools integrated into the platform. For all IoT devices, the platform provides visualisation of the data on the map (for devices with location). This is especially useful for devices tracking ambient metrics (temperature, air quality) devices for tracking (show buses on the map, show location of a user who triggered an alarm). For the devices and services providing the numerical data, this can also be visualised on the platform automatically and it also works with time-series data.

[1] https://wso2.com/
[2] https://ckan.org/

The service providers can also set certain rules for the data received on the platform. In most cases these rules are different alarms, defined rules based on simple arithmetical and logical operations. When the alarm is triggered the subscribed user(s) are notified about the event. The devices, products and services are discoverable through marketplace, which is based on PrestaShop[3] framework. Each participating partner can add one or more services or products to the platform with accompanying metadata (image, title, description, type, reference number, quantity and price). The types of products on the marketplace can be separated into 3 groups: a) Physical devices (e.g. temperature sensor); b) Intermediate services (e.g. not meant for the end users as machine learning algorithm that detect unusual patterns in data); c) Services for end user (e.g. home care for older adults).

TECHNICAL ANALYSIS				
EKOSMART				
DEVICE MANAGEMENT				
Connectivity of heterogeneous IoT devices	Remote access to IoT devices	Remote control of IoT devices	Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Autodiagnostic features
Yes	Partial	Partial	Partial	No
INTEGRATION/INTEROPERABILITY				
Interoperability is implemented using a semantic approach	Interoperability is implemented using a syntactic approach	Implements mechanisms able to make available platform services to third parties outside the Platform	Implements Restful / SOA web services mechanisms to access interoperability feature	Implements interoperability between services and functions defined inside the Platform
No	Yes	Yes	Yes	Yes
INFORMATION SECURITY				
No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Data is encrypted on the network	Communications between Platform to the Internet are secured	Protocols and cryptographic schemes ensure end-to-end data integrity	Only authorized devices can be connected to the Platform
Partial	Partial	Partial	Yes	Yes
DATA ANALYTICS				
Implements real-time data analytics	Implements predictive data analytics	Implements data analytics for anomaly detection	Data analytics are included in a Marketplace	Data analytics are accessible using REST/SOA API calls
Partial	No	No	No	Yes
VISUAL INTERFACES				
Visualization of data	Manipulation of data	Creation of analytics	Dashboard available for each user group (caregiver, patient, etc.)	Web application or standalone
Yes	Yes	Yes	Yes	WEB APPLICATION

Figure 7: The technical analysis for Ekosmart

[3] https://www.prestashop.com/en

3.2.4. Reach2020

The Reach2020 platform is defined as a Product-Service-System (PSS). The "Touchpoints and Engine concept" structures the envisioned Reach2020 PSS architecture that integrates the different Personalised Intelligent Interior Units (PI²Us), which are defined as a set of smart modular furniture elements that serve as physical carrier elements of Reach2020 functionality. It includes different devices such as pressure mattress, thermal camera, ECG sensors, activity monitoring sensors (wearables).

The platform defines five "Touchpoints" that address different scenarios:

1. Touchpoint 1: Personal Mobility Device
2. Touchpoint 2: Active Environment
3. Touchpoint 3: Socializing & Nutritional Monitoring + Intervention
4. Touchpoint 4: Gaming & Training
5. Touchpoint 5: Wearables

On the *Physical layer*, Reach2020 provides connection with devices including ambient sensors, pressure mattress, thermal camera, ECG sensors, activity monitoring sensors (wearables) as well as other devices defined in the PI²Us. Touchpoint is modular in itself, also serving as a kind of physical product platform.

The services of the *Service laye*r are offered through the different Touchpoints in a modular way. The Touchpoints serve as data gathering scenarios through different devices (PI²Us) as well as mediator of services and interventions coordinated by the Engine towards the end user. Each Touchpoint is made up of several subsystems which allow to adapt the system both for a certain person or setting as well as over time.

Regarding the *Application layer*, Reach2020 platform offers the information through the following visualization tools: TV / Kinect interface, Data Dashboard (Philips), Ubiquitous large scale touch surface for gesture interaction, Interface for food intake app, Interface for food advice app, and "Nudging tablet". Reach2020 offers applications for mobilization and rehabilitation, physical activity, training, food and nutrition, mobility, and patient motivation. The platform also offers user management, authentication, communication and personalization.

There is no *Semantic layer*.

Interoperability is achieved by supporting a wide set of interoperability standards. The platform provides cross compatibility protocols to integrate with a variety of third party platforms, including Health Suite Digital Platform (HSDP) by Philips, and supports several interoperability standards (more info can be found in [4]).

In terms of information security, it allows secured access and control to devices. It supports data privacy tools, including pseudonymization, secure database mechanisms with access log, approval strategies for collection of non-invasive lifestyle data.

[4] Deliverable REACH System integration: https://mediatum.ub.tum.de/doc/1482708/1482708.pdf

TECHNICAL ANALYSIS				
REACH 2020				
DEVICE MANAGEMENT				
Connectivity of heterogeneous IoT devices	Remote access to IoT devices	Remote control of IoT devices	Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Autodiagnostic features
Yes	Yes	No	Yes	No
INTEGRATION/INTEROPERABILITY				
Interoperability is implemented using a semantic approach	Interoperability is implemented using a syntactic approach	Implements mechanisms able to make available platform services to third parties outside the Platform	Implements Restful / SOA web services mechanisms to access interoperability feature	Implements interoperability between services and functions defined inside the Platform
No	No	Yes	Yes	Yes
INFORMATION SECURITY				
No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Data is encrypted on the network	Communications between Platform to the Internet are secured	Protocols and cryptographic schemes ensure end-to-end data integrity	Only authorized devices can be connected to the Platform
Yes	Yes	Yes	Yes	Yes
DATA ANALYTICS				
Implements real-time data analytics	Implements predictive data analytics	Implements data analytics for anomaly detection	Data analytics are included in a Marketplace	Data analytics are accessible using REST/SOA API calls
Yes	Yes	Yes	No	Yes
VISUAL INTERFACES				
Visualization of data	Manipulation of data	Creation of analytics	Dashboard available for each user group (caregiver, patient, etc.)	Web application or standalone
Yes	No	Yes	Yes	BOTH

Figure 8: The technical analysis of Reach2020

3.2.5. Sensinact

In the Eclipse SensiNact *Physical layer*, emerging IoT devices, legacy systems, increasing number of social networks, mobile applications, open data repositories and web data are the potential exploitable data sources. SensiNact ships with southbound bridges for using a lot of common devices including Zigbee (motion sensors, force sensor, etc.), EnOcean (remote controls, windows opener detectors, etc.), CoAP (sliders, buttons, etc.). It also provides a bridge for retrieving context information using NGSi 9/10 protocol. Thanks to an OSGi based architecture, it is possible to add bridges on the fly, while the gateway is running, to allow communication with new kind of devices. The project uses third party software with various license types such as Apache v2.0 and EPL, JSON and MIT. sensiNact thus provides connectivity support to those data sources including today's IoT protocols and platforms such as LoRa, Zigbee, IEEE 802.15.4, Sigfox, enOcean, MQTT, XMPP, NGSI, HTTP, CoAP, etc. With its modular approach, connectivity support for new protocols can be rapidly developed and dynamically added to the platform, even at run-time. Those resources are accessible by generic and easy to use Application Programming Interfaces (API) providing synchronous (on demand) and asynchronous (periodic or event based) access to data/actions of IoT devices, as well as access to historic data.

Considering the *Service layer* and the *Application layer*, the Eclipse sensiNact platform aims at creating a common environment in which heterogeneous devices can exchange information and interact among each other in the IoT world. This environment is composed of two tools: the sensiNact Gateway, which refers to the *Service Layer* and aims at integrating devices and aggregating data from various sources and sensiNact Studio, which refers to the Application Layer, aiming at interacting with the sensiNact Gateway to visualize the devices and the data.

SensiNact Gateway interconnects IoT devices using different southbound IoT protocols such as Zigbee, EnOcean, LoRa, XBee, MQTT, XMPP, as well as platforms such as FIWARE and allows access to them with various northbound protocols such as HTTP REST, MQTT, XMPP, JSON RPC and CDMI. The gateway can also host applications and manage them using an application manager module.

The sensiNact Studio proposes an IDE (Integrated Development Environment) based on Eclipse to manage the existing devices, in addition to develop, deploy and manage IoT applications. It is a service composition tool which assists developers in building applications by binding the services via events and actions. With a Domain Specific Language, the developers can express the application logic in terms of ECA (Event-Condition-Action) rules, which is verified and validated by the tool before its deployment into the sensiNact platform. The application developers can then remotely monitor and manage applications (install, start, stop, uninstall, etc.). The tool also provides means to easily build support for new types of protocols and platforms and add it to the platform on-the-fly.

SensiNact adopts a generic and extensible data model to facilitate building adapters for various protocols. It is core model is based on 4 types of resources: sensor data, action, state variables, and properties. Those resources are accessible by generic and easy to use Application Programming Interfaces (API) providing synchronous (on demand) and asynchronous (periodic or event based) access to data/actions of IoT devices, as well as access to historic data.

A first level of security of the Sensinact platform is reached by the available security tools in the OSGi environment: ServicePermission and ConditionalPermissionAdmin. The data collected from the sensors connected to Sensinact gateway use Application Programming Interfaces (API) providing synchronous (on demand) and asynchronous (periodic or event based) access to data/actions of IoT devices, as well as access to historic data. This is real-time information provided. This data can be visualized with sensiNact Studio which allows managing devices/services connected to the platform and rapidly creating applications and deploying them to the platform.

Semantic interoperability support is not present in sensiNact, but it is supposed to be implemented in future updates. SensiNact has been successfully used in close to real-life environments in various application domains such as smart city, smart home, smart shopping, smart health care within the context of collaborative projects.

TECHNICAL ANALYSIS				
sensiNact				
DEVICE MANAGEMENT				
Connectivity of heterogeneous IoT devices	Remote access to IoT devices	Remote control of IoT devices	Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Autodiagnostic features
Yes	Yes	Yes	No	No
INTEGRATION/INTEROPERABILITY				
Interoperability is implemented using a semantic approach	Interoperability is implemented using a syntactic approach	Implements mechanisms able to make available platform services to third parties outside the Platform	Implements Restful / SOA web services mechanisms to access interoperability feature	Implements interoperability between services and functions defined inside the Platform
No	Yes	Yes	No	Yes
INFORMATION SECURITY				
No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Data is encrypted on the network	Communications between Platform to the Internet are secured	Protocols and cryptographic schemes ensure end-to-end data integrity	Only authorized devices can be connected to the Platform
No	Yes	Yes	Yes	Yes
DATA ANALYTICS				
Implements real-time data analytics	Implements predictive data analytics	Implements data analytics for anomaly detection	Data analytics are included in a Marketplace	Data analytics are accessible using REST/SOA API calls
Yes	Yes	Yes	No	Yes
VISUAL INTERFACES				
Visualization of data	Manipulation of data	Creation of analytics	Dashboard available for each user group (caregiver, patient, etc.)	Web application or standalone
Yes	Yes	Yes	Partial	WEB APPLICATION

Figure 9: The technical analysis of Sensinact

3.2.6. UNCAP

In the *Physical layer* the UNCAP certification suite can be connected to multiple devices such as blood pressure monitor, heart rate monitor, oximeter, scale, glucometer through drivers. This driver creates the link of the device to the UNCAP BOX. Drivers have to be written/created if a device does not support the already existing drivers. The project UNCAP has already developed a few drivers for some key communication protocols.

In the *Service layer* all services are modularized and interoperable, accessible through REST API. A data broker and a storage module offer intermediary services. The Data Broker collects the data published by multiple devices (medical devices, non-medical devices, etc.) and applications. It also collects and dispatches different types of data, such as timeseries data and asynchronous events.

The storage is the lower layer where data will be stored and from where all modules can gather information to be processed. This level is responsible for managing all data uploaded to the cloud from the various users/applications and should guarantee accessibility to those when needed.

In terms of the *Application layer* UNCAP offers viewing the data, different dashboards for carers, medical professionals, elderly adults. It also offers functionality of reminders, alerts and emergency alerts. Location based geofencing is implemented as well as indoor and outdoor localization and navigation. Billing, user management, authentication, communication configuration functionality is also offered. More specifically exergames are offered as well as a bio-signal tracking module.

UNCAP does not have a *Semantic layer*, it uses semantic annotation to describe the services it offers.

In terms of device management, devices can be added and discovered. They can only be used as data sources. Basic protocols, for which drivers are already developed are KNX and Bluetooth Low Energy (BLE). The system is connected to hospitals through FHIR HL7 v2,3.

Security measures have been thought of, but no further details are mentioned on how this is achieved. Due to the distributed nature of the system on mobile devices, home servers, clouds it is especially important. Authentication and authorization are achieved by managing permissions and user groups.

UNCAP offers data representation in real-time and of historic data. It also offers rule creation functionality. Besides data representation, different dashboards for carers, medical professionals and older adults are readily available.

TECHNICAL ANALYSIS				
UNCAP				
DEVICE MANAGEMENT				
Connectivity of heterogeneous IoT devices	Remote access to IoT devices	Remote control of IoT devices	Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Autodiagnostic features
Yes	Yes	No	No	No
INTEGRATION/INTEROPERABILITY				
Interoperability is implemented using a semantic approach	Interoperability is implemented using a syntactic approach	Implements mechanisms able to make available platform services to third parties outside the Platform	Implements Restful / SOA web services mechanisms to access interoperability feature	Implements interoperability between services and functions defined inside the Platform
Yes	Yes	Yes	Partial	No
INFORMATION SECURITY				
No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Data is encrypted on the network	Communications between Platform to the Internet are secured	Protocols and cryptographic schemes ensure end-to-end data integrity	Only authorized devices can be connected to the Platform
Yes	Yes	Partial	Yes	Yes
DATA ANALYTICS				
Implements real-time data analytics	Implements predictive data analytics	Implements data analytics for anomaly detection	Data analytics are included in a Marketplace	Data analytics are accessible using REST/SOA API calls
No	No	Yes	No	Yes
VISUAL INTERFACES				
Visualization of data	Manipulation of data	Creation of analytics	Dashboard available for each user group (caregiver, patient, etc.)	Web application or standalone
Yes	Yes	No	Yes	WEB APPLICATION

Figure 10: The technical analysis of UNCAP

3.2.7. FiWare

FIWARE is a curated framework of open source platform components, also known as Generic Enablers (GEs), which can be assembled together and with other third-party components, so as to create custom hybrid platforms covering the needs of specific projects and to accelerate the development of Smart Solutions. The main and only mandatory component of any FIWARE-based platform or solution is a FIWARE Context Broker GE, bringing a cornerstone function in any smart solution: the need to manage context information, enabling to perform updates and bring access to context. A rich suite of complementary FIWARE GEs are available, dealing with interfacing with the Internet of Things (IoT), robots and third-party systems, context Data/API management, publication, and monetization, and context information processing, analysis, and visualization. Depending on the solution's requirements, the FIWARE GEs which enable the required functionality can be selected and integrated, or new ones can be implemented as long as they are compliant with the corresponding FIWARE GE Open Specifications. Any implementation of a FIWARE GE is, by nature, replaceable.

The *Physical layer* is composed of all the devices and gateways connected to the platform, which can be of different types, collect various kinds of data and communicate via various protocols, depending on the exact needs of the offered solution and the devices availability from their vendors. The deployment of the architecture of the physical layer is typically distributed across a large number of devices and several gateways. A device is a hardware entity, component or system that either measures properties of a thing/group of things or influences the properties of a thing/group of things or both measures/influences. Sensors and actuators are devices. An IoT gateway is a hardware device that hosts a number of features of one or several Gateway GEs of the IoT Service Enablement, e.g. provide inter-networking and protocol conversion functionalities between devices. This is an optional element in the FIWARE IoT model, which is usually located at proximity of the devices (sensors/actuators) to be connected and aims to optimize the network traffic sent to the Backend and IoT services to reach higher efficiency and reliability.

The *Service layer* mainly encapsulates a Context Broker GE, which is the core and mandatory component of any FIWARE-based solution, which enables to manage context information (perform updates and bring access to context) in a highly decentralized and large-scale manner. The Broker keeps virtual representations of the physical devices, hence interaction with devices happens by updating and modifying the virtual representations attached/corresponding to them. From an architectural point of view, the Context Broker acts as a blackboard in a typical blackboard architecture. It is the core and control piece of the platform, in charge of interacting with the other components and agglutinate data. Therefore, Context Broker plays a key role when developing a data/context scenario. A number of Data/Context Management FIWARE GEs are available or under incubation, such as the *Orion Context Broker GE* that currently provides the FIWARE NGSI v2 Restful API enabling to perform updates, queries or subscribe to changes on context information, and the *Orion-LD Context Broker GE* and the *Scorpio Broker GE* that support the ETSI NGSI-LD API specifications.

The service layer further includes components accompanying a Context Broker as part of Core Context Management, such as the *STH Comet GE* that brings the means for storing a short-term history of context data (typically months) on MongoDB, the *Cygnus GE* that brings the means for managing the history of context that is created as a stream of data which can be injected into multiple data sinks, including some popular databases like PostgreSQL, MySQL, MongoDB or AWS DynamoDB as well as BigData platforms like Hadoop, Storm, Spark or Flink, the *Draco GE* that is an alternative data persistence mechanism for managing the history of context based on Apache NiFi and is a dataflow system based on the concepts of flow-based programming which supports powerful and scalable directed graphs of data routing, transformation, and system mediation logic and also offers an intuitive graphical interface, and the *QuantumLeap GE* that supports the storage of context data into a time series database (CrateDB and Timescale).

The Service layer also includes a number of FIWARE GEs for making it easier to interface with the Internet of Things devices (IoT devices Backend) as well as Robots and Third-party systems for the purpose of gathering valuable context information or trigger actuations in response to context updates. Since devices have a lot of differences and particularities making it impossible to provide a solution where one size fits all, and the platform needs to be able to support and integrate a number of heterogeneous devices, a modular approach is followed. IoT Agents act as translators between the protocols that devices use to send or receive information and the common language and data model across all the platform, i.e. FIWARE NGSI. The *IDAS GE* offers a wide range of *IoT Agents* making it easier to interface with devices using the most widely used IoT protocols (LWM2M over CoaP, JSON or UltraLight over HTTP/MQTT, OPC-UA, Sigfox or LoRaWAN), while a number of incubated GEs bring open source implementations of modern standards for the communication with IoT devices and Robotics, as well as the exchange of electronic data and documents in reliable and trusted ways. FIWARE also enables solution builders to create their own IoT Agents in order to be able to connect to and use any type of device available now or in the future.

In the *Application layer*, we can distinguish between a) the FIWARE Marketplace platform that can be used to locate and purchase FIWARE-based solutions and b) the FIWARE platform components (GEs) which can be used by developers and providers of a solution to enable b1) the management and monetization of applications, services and data in a business framework across their whole service life cycle or b2) the processing, analysis and visualisation of context.

The FIWARE Marketplace serves the purpose of globally disseminating existing commercial offerings around FIWARE. It is a global one-stop shop that gives visibility to a wide range of solutions/ platforms powered by FIWARE, FIWARE-ready technologies as well as FIWARE related training/coaching or consultancy, integration and support services.

On the FIWARE GEs side, there are Generic Enablers that can be utilised to build an ecosystem of applications, services and data delivery that is sustainable and fosters innovation as well as cross-fertilization. In particular, these support managing of applications, services and data in a business framework across its whole service life cycle, starting from creation through composition of applications and data to publication, monetization and revenue sharing. The *CKAN extensions GE* brings a number of add-ons enabling to extend current capabilities of the world-leading CKAN Open Data publication platform to allow publication of datasets matching right-time context data, the assignment of access terms and policies to those datasets and the assignment of pricing and pay-per-use schemas to datasets, the *Biz Framework GE* brings backend support to Context API/Data monetization based on open TM Forum Business APIs, *Idra GE* 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 (DCAT-AP) to discover open datasets, the *APInf API Management Framework* is a tool for API owners to manage their APIs that provides all the necessary features to run business with APIs and makes it easy for API consumers to find and start using the standard APIs. The *Application Mashup GE* aims at allowing end users without programming skills to quickly compose beautiful web applications and dashboards/cockpits mashing up widgets, operators and data sources from a shared catalogue. The *Data Visualisation GE* aims at creating agile, beautiful visualizations and meaningful reports useful to present the large variety of datasets. Data stakeholders will bring in the play as well as providing customisable data analytics.

There are further FIWARE GEs available for making it easier to process, analyse or visualize context information for the purpose of implementing the Application layer of a solution and enabling the “smart behaviour” expected. The *Wirecloud GE* brings a powerful web mashup platform making it easier to develop operational dashboards which are highly customizable by end users, the *Knowage GE* brings a powerful Business Intelligence platform enabling to perform business analytics over traditional sources and big data systems build on context history, the *Kurento GE* enables real-time processing of media streams supporting the transformation of video cameras into sensors as well as the incorporation of advanced application functions (integrated audiovisual communications, augmented reality, flexible media playing and recording, etc.), the *Cosmos GE* enables a more easy Big data analysis over context integrated with popular Big Data platforms (Spark and Flink), the *FogFlow GE* is a distributed execution framework to support dynamic processing flows over cloud and edges, while the incubated the *Perseo GE* introduces Complex Event Processing (CEP) defined using a rules-based system, enabling you to fire events which send HTTP requests, emails, tweets, SMS messages etc., the *OpenVidu GE* is an abstraction layer for Kurento, making media processing easier to program.

Regarding the *Interoperability layer*, in order to implement a FIWARE-based solution, a number of FIWARE platform components (GEs) are utilised in an interoperable manner. All interactions between the platform components or applications and the Context Broker take place using the FIWARE NGSI RESTful API open standard. The same APIs (if chosen to be exposed externally) can take the form of an interoperability layer for the interoperation with external platforms or just for data consultation between platforms. However in case that the external platform is not also FIWARE-based, i.e. does not comply to the same FIWARE NGSI data model, an intermediate layer/component may be needed in order to perform the necessary mappings/ translations or any other operations needed for achieving compatibility between the heterogeneous platform APIs.

In the *Semantic layer*, semantics are supported by FIWARE in terms of data models and APIs implementing them. Data in FIWARE refers to information that is produced, generated, collected or observed that may be relevant for processing, carrying out further analysis and knowledge extraction. In terms of the traditional NGSI v2 FIWARE data model, a cornerstone concept is that data elements are not bound to a specific format representation. Also, the structural definition of Data Elements encloses its Data Type, a number of Data Element attributes (which enclose the following: Name, Type, Value) and, optionally, a set of Metadata Elements also referred to as semantic data (which have also data-like attributes: Name, Type, Value). Lately, the FIWARE NGSI v2 information model has been evolved to better support linked data (entity relationships), property graphs and semantics (exploiting the capabilities offered by JSON-LD). This work has been conducted under the ETSI ISG CIM initiative and has been branded as NGSI-LD. The main constructs of NGSI-LD are Entity, Property and Relationship. NGSI-LD Entities (instances) can be the subject of Properties or Relationships. Properties and Relationships can be the subject of other Properties or Relationships. Thus, in the NGSI-LD information model (and corresponding API) there are no attribute's metadata, but just "properties of properties" or "properties of relationships".

FIWARE further provides few GEs related to semantic interoperability, however these do not seem to have been updated to the latest NGSI version. As mentioned above, FIWARE is currently migrating to the NSGI-LD specification to enhance relationships between entities, but currently it is up to the logic of the application to navigate between entity relationships. The IoT Discovery GE acts as a meeting point for IoT Context Producers to register the availability of their Things and Sensor devices, and IoT Context Consumers to discover them, using either the OMA NGSI-9 messaging protocol, or the Sense2Web API that supports Linked Open Data. Also as part of the IoT Broker GE, an IoT Knowledge Server is provided, which contains a large amount of IoT semantic knowledge useful from the perspective of a project. The IoT Knowledge Server is a standalone component created for serving semantic information to the IoT Broker semantic ontologies.

FEATURES NOTES

FIWARE has the ability to connect to heterogeneous devices, which it handles in a device-agnostic manner, only based on the device model used. There is a dedicated FIWARE webpage for allowing the download of various harmonized data models to enable data portability for different applications, or a custom model can be created following the NGSI-v2 or NGSI-LD specification. A list of devices connected to the platform is maintained by the Context Broker along with their current status. Managing this information is possible since Context Broker keeps virtual representations of the physical devices. Interaction with devices will happen by updating and modifying the virtual representations attached/corresponding to them. From an architectural point of view, Context Broker acts as a blackboard in a typical blackboard architecture. FIWARE can handle any operation from/to a device, such as configuration, firmware (or any other software) updates and provide device-level error reporting and error handling, although these functionalities are not provided out of the box. Keeping track of the historic values of each device can be added with the addition of the *Short Term Historic (STH)* FIWARE component. Common operations, such as calculating the minimum, maximum, mean, bias or deviation of a set of data can be performed using the STH time series database without an intermediate step of processing the whole set of data in order to obtain the desired information. Hence statistics can be obtained easily, if the values of interest have been stored in STH. The usage of another FIWARE GE for data analysis could be of value. Hence statistics solutions are also implementable, but not directly available out of the box.

FIWARE has a fully distributed nature, pertaining to all layers of the platform. Integration and interoperability (inter- and intra-platform) are supported through the Restful API exposed by the each of the FIWARE components (GEs) used to build and integrate each specific solution, the core of which is the Context Broker. FIWARE can also support semantic interoperability, which is an inherited part of its data model specification. However, since this is a period of transition from the old to a new data model specification and not all FIWARE components have been updated to support the latest model, a custom approach and some additional effort may be required in order to enable semantic interoperability.

In the latest release of FIWARE and especially in what concerns the various IoT Agents included in the *IDAS GE* it seems to be possible to have fully encrypted connections between devices and the IoT platform for several protocols, e.g. MQTT over TLS. Even in the case of this not being available for a specific protocol, it could be implemented and added by a project's development team, albeit with some effort. Data transmission between the APIs of the various FIWARE components (GEs) can also be encrypted by installing TLS certificates in the servers and utilizing the HTTPS protocol. As far as identity management, authentication and authorization are concerned (including users, devices and applications), these are supported by FIWARE based on three main security elements: Identity Management (IdM), Policy Enforcement Point (PEP), and Policy Administration Point / Policy Decision Point (PAP/PDP). The *Keyrock IdM GE* carries the information about users, roles and profiles. It also manages authentication and authorization in applications and backend services. It sends and validates tokens, as well as authentication mechanisms. It further can be used for identifying and registering devices with the platform. A *PEP Proxy GE* is meant to catch the request from a certain component, and force the requirements specified in terms of identification and authorization for that specific component, before starting using it. PEP Proxy is in charge of orchestrating all the communications between the Keyrock element and the PDP element. *AuthzForce GE* is the reference implementation of the Authorization PDP Generic Enabler that provides an API to get authorization decisions based on authorization policies and authorization requests from PEPs. AuthZForce chooses the requests after selecting the allowed or banned actions for each role vs. component.

Real-time, batch, predictive, and interactive analytics can be conducted by FIWARE, based on the deployment and usage of the corresponding components (GEs). The FIWARE GEs dealing with Data Analytics enable, among others, to perform real-time analytics, to analyse data stored on big data clusters or NoSQL databases, to perform smart intelligence, i.e. the usual business intelligence on structured data, also oriented to self-service capabilities and agile prototyping, to aid with enterprise reporting, i.e. to produce and distribute static reports, to perform location intelligence, i.e. to relate business data with spatial or geographical information, to undertake performance management, i.e. to manage KPIs and organize scorecards, to perform predictive analysis, to undertake embedded intelligence tasks, to perform user behaviour analysis (e.g. for anomaly detection), to extract meaningful conclusions as to the state of a smart solution and bring value to the solution.

The existence of visual interfaces in FIWARE depends on whether a GE (covering one of the aforementioned data analytics areas) implements one.

As part of the *Keyrock Identity Management GE* there is a User Interface available which brings support to secure and private OAuth2-based authentication of users and devices, user profile management, privacy-preserving disposition of personal data, Single Sign-On (SSO) and Identity Federation across multiple administration domains.

The *Draco GE* is an alternative data persistence mechanism for managing the history of context. It is based on Apache NiFi and is a dataflow system based on the concepts of flow-based programming. It supports powerful and scalable directed graphs of data routing, transformation, and system mediation logic and also offers an intuitive graphical interface.

The *Wirecloud GE* is a tool which helps users to rapidly generate new application mashups based on NGSI and other data sources. To speed up development, the Wirecloud architecture has been defined to split mashup operations into a series of simple reusable tasks (widgets and operators). Each task has well-defined input and output interfaces, and the Wirecloud UI allows mashup creators to wire up a series of tasks into a complex chain of data processing and visualization events.

The Knowage GE, which brings a powerful Business Intelligence platform enabling to perform business analytics over traditional sources and big data systems built on context history, also offers an interactive Visual interface for depicting (real-time) results of the conducted analyses.

TECHNICAL ANALYSIS				
FI-WARE				
DEVICE MANAGEMENT				
Connectivity of heterogeneous IoT devices	Remote access to IoT devices	Remote control of IoT devices	Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Autodiagnostic features
Yes	Yes	Yes	Yes	Yes
INTEGRATION/INTEROPERABILITY				
Interoperability is implemented using a semantic approach	Interoperability is implemented using a syntactic approach	Implements mechanisms able to make available platform services to third parties outside the Platform	Implements Restful / SOA web services mechanisms to access interoperability feature	Implements interoperability between services and functions defined inside the Platform
Yes	Yes	Yes	Partial	Yes
INFORMATION SECURITY				
No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Data is encrypted on the network	Communications between Platform to the Internet are secured	Protocols and cryptographic schemes ensure end-to-end data integrity	Only authorized devices can be connected to the Platform
Partial	Yes	Yes	Yes	Yes
DATA ANALYTICS				
Implements real-time data analytics	Implements predictive data analytics	Implements data analytics for anomaly detection	Data analytics are included in a Marketplace	Data analytics are accessible using REST/SOA API calls
Yes	Yes	Yes	Yes	Yes
VISUAL INTERFACES				
Visualization of data	Manipulation of data	Creation of analytics	Dashboard available for each user group (caregiver, patient, etc.)	Web application or standalone
No	No	Yes	No	WEB APPLICATION

Figure 11: The technical analysis of FiWare

3.2.8. Onesait

The Onesait Platform aims to provide flexibility so that developers can build their own solutions. Based on Open Source components, Onesait Platform covers the entire life cycle of information (from ingest to visualization through its process and analysis).

From a general point of view the platform can be seen as consisting of three main layers:

- Interaction layer: layer that allows for communication and collaborative interaction between people and machines, based on a unified representation of information. It interacts with different channels, keeping a common semantic language.

- Action Layer: it includes business rules and execution of delocalised, scalable and flexible processes. It executes parametrized commercial actions and processes to agilely adapt to business requirements.

- Intelligence Layer: It extracts analytical information and stores it according to its type, latency, size and security level.

With reference to the purely technical aspect, the platform is in turn divided into five layers: Acquisition Layer, Knowledge Layer, Publication Layer, Management Layer and Support Layer.

The *Physical layer* links to the Acquisition Layer and is composed of five main elements:

- IoT Broker: this Broker allows devices, systems, applications, websites and mobile applications to communicate with the platform through one of the supported protocols. It also offers APIs in different languages.

- Kafka Server: the platform integrates a Kafka cluster that allows for communication with systems using this exchange protocol, usually because they handle a large volume of information and need a low latency.

- Dataflow: this component allows you to configure streaming data streams from a web interface. These flows are made up of one origin (which can be files, databases, TCP services, HTTP, queues, ... or the IoT Broker platform), one or more transformations (processors in Python, Groovy, JavaScript, ...) and one or more destinations (the same options as the origin).

- Digital Twin Broker: this Broker allows communication between the Digital Twins and the platform, and with each other. It supports REST and Web Sockets as protocols

- Video Broker: allows to connect to cameras via WebRTC protocol, and to process the video stream associating it to an algorithm (people detection, OCR, etc.).

The core of the *Service layer* is the Data-Centric approach, which refers to an architecture where data is the main, permanent asset, and applications come and go. In the Data-centric architecture, the data model precedes any given application's implementation and will be valid long after these are replaced: there is a data model, a semantic data model and each application functionality reads and writes through the shared model. Data-Centric architecture is supported through the Ontology, and all the functionality of the Platform looks around this concept. The Ontologies are the Entities that the system manages.

The solution's information flows solution will navigate through the Platform, (ingestion and processing, storage, analytics and publication) from the data producers to the information consumers, following the paradigm of "listen, analyse, act". This allows for the ingestion of information from real-time data sources of virtually any nature type, from devices to management systems. This real-time information from devices and systems accesses the platform through the most appropriate gateways (multiprotocol interfaces) for each system, is then processed, reacting in real time to the configured rules, and finally remains persisted in the storage module's Real Time Database (RealTimeDB).

On the other hand, the rest of the information coming from more generic sources ("Batch Flow") that is obtained by means of processes of extraction, transformation and loading in batch mode (not real time) accesses the solution through the mass loading module of information (ETL).

The *Application layer* links to Publication, Management and Support layers that are described as follows.

Publication Layer:

- API Manager: this module allows you to visually create APIs on the ontologies managed by the platform. It also offers an API Portal for the consumption of the APIs, and a Gateway API to invoke the APIS.
- Dashboard Engine: this engine allows to create, visually and without programming, complete dashboards on the information (ontologies stored in the platform), and then make them available for consumption outside or inside the platform.

Management Layer:

- Control Panel: the platform offers a complete web console allowing for a visual management of the platform's elements using a web-based interface. This entire configuration is stored in a configuration database. It offers a REST API to manage all these concepts and a monitoring console to show each module's status.
- Access Manager: allows to define how to authenticate and authorize users defining Realms with roles, users directory (LDAP,), protocols (OAuth2, ...)
- Caas Console: allows to manage from a web console all the modules deployed (as Docker containers orchestrated by Kubernetes), including version updates and rollback, number of containers, scalability rules,...

Support Layer:

- Market Place: allows to define assets generated into the platform (APIs, dashboards, algorithms, models, rules, ...) and publish in order other users can use it, free or paid)
- GIS Viewers: from the console you can create GIS layers (from ontologies, WMS services, KML, images) and GIS viewers (currently under Cesium technology) from these layers managed by the platform
- Open Data Portal: platform includes a CKAN Portal connected with the platform so that the ontologies can be published as datasets or datasets can be exported to ontologies to be processed with the rest of the pieces of the platform
- Files Manager: this utility allows you to upload and manage files from web console or REST API. These files are then managed with the platform's security.
- WebApps Server: the platform allows you to serve Web applications (HTML + JS) uploaded through the web console of the platform.
- Config Management: this utility allows you to manage configurations (in YAML format) of the applications of the platform by environments

The management can be done through the Platform Control Panel, a complete web console that allows a visual management of the elements of the platform through a web-based interface. All this configuration is stored in a configuration database (ConfigDB). Within its functionality it adds:

- Development control panel: integrates all the tools in the platform that the developer will use when creating applications, including creation ontologies, rules, panels, assigning security, etc.
- DevOps & Deploy: this console allows you to configure the tools for the continuous integration of the platform, and also to implement the platform instances and the additional components that a solution may require.
- Security: allows you to configure all security aspects of the solution, such as the user repository (LDAP, the platform itself), and define and manage users and roles, etc.
- Device management: allows managing and operating the devices of the IoT solutions.
- Monitoring: helps monitoring the platform and solutions through KPIs, alerts, etc.

The *Semantic layer* is represented by the Knowledge Layer and these are its main components:

- Semantic Information Broker: once the information is acquired, it reaches this module, that: validates whether the Broker client has permissions to perform that operation (insert, query, ...) or not, and gives semantic content to the received information, validating whether the sent information corresponds with this semantics (ontology) or not.
- Semantic Data Hub: this module acts as a persistence hub. Through the Query Engine, it allows to persist and consult on the underlying database where the ontology is stored, where this component supports MongoDB, Elasticsearch, relational databases, Graph databases,...
- Streaming Engines: supported by:
  - Flow Engine: this engine allows creating process flows, both visually and easily. It is built on Node-network. A separate instance is created for each user.
  - Digital Twin Orchestrator: the platform allows for the communication between Digital Twins to be orchestrated visually through the same FlowEngine engine. This orchestration creates a bidirectional communication with the Digital Twins.
  - Rules Engine: allows you to define business rules from a web interface that can be applied to data entry or scheduled
  - SQL Streaming Engine: allows to define complex streams as data arrives in a SQL-like language
- Data Grid: this internal component acts as a distributed cache as well as an internal communication queue between modules.
- Notebooks: this module offers a multi-language web interface so that the Data Scientist team can easily create models and algorithms with their favourite languages (Spark, Python, R, SQL, Tensorflow ...).

The *Interoperability layer* is closely related to the *semantic layer,* as previously described the Platform uses a Data-Centric Architecture and the Ontology represents this data model that can be shared between applications and systems: an Ontology is the definition of an Entity in the simplest case or of a Domain Model in the most complex case, that is finally represented as a JSON-Schema. Besides, the platform offers a set of Templates (or Data Models) that allow ontologies to be created following the best recommendations and standards in this regard.

A good example of an attempt to standardize in the Smart Cities field is the FIWARE Data Model[5] (supported by the platform).

Of the many advantages related to the use of ontologies the following can be mentioned:

- Storage independence: the platform uses Mongo by default to store the ontologies but, depending on the use case, it allows to store the ontologies on other repositories. The platform will manage the creation on the DB (indexes, partitions, ...)
- SQL query engine: whichever be the underlying database of the ontology, the platform allows you to query the ontologies in SQL. This, coupled with the independence of the database, allows, where appropriate, to migrate from the chosen database technology should the scenario change. It will be simple:
- Security associated with the ontology: working with ontologies has other advantages. For example, the platform automatically manages security. By default, the user who creates the ontology is the only one who can access it, and she can give read, write or complete management permissions to the users that she considers. Besides, if you use the project concept, you can create shared entities for the users that make up the project
- Syntactic validation: as said, the ontology is represented as a JSON-Schema, and ontology instances (in a relational model, that is the records) are JSON's. Whichever the repository where the ontology is stored, its interface is a JSON (you can see it by doing a query). Well, the platform automatically validates that the information sent to the platform complies with the defined JSON-Schema. In a JSON-Schema, you can define simple (mandatory), numerical (> 10) and complex semantic validations.
- Rules and visual flows associated with ontologies: the platform offers several engines that can be executed upon arrival of an ontology instance. It is very common to have a control ontology that triggers an action in another system (invocation to a REST Service) or simply sends an e-mail. The platform allows you to define this without any programming:
- Transparent protocol management: the platform allows you to manage the ontologies in different protocols, including REST, MQTT, Kafka, WebSockets, ... and the client can choose the protocol according to the use case.
- Multilanguage APIS: in addition to the independence of the protocols, the platform offers APIs in several languages.

[5] https://www.fiware.org/developers/data-models/.

TECHNICAL ANALYSIS

onesait platform

DEVICE MANAGEMENT

Connectivity of heterogeneous IoT devices	Remote access to IoT devices	Remote control of IoT devices	Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Autodiagnostic features
Yes	Yes	Yes	Partial	Partial

INTEGRATION/INTEROPERABILITY

Interoperability is implemented using a semantic approach	Interoperability is implemented using a syntactic approach	Implements mechanisms able to make available platform services to third parties outside the Platform	Implements Restful / SOA web services mechanisms to access interoperability feature	Implements interoperability between services and functions defined inside the Platform
Yes	No	Partial	Yes	Yes

INFORMATION SECURITY

No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Data is encrypted on the network	Communications between Platform to the Internet are secured	Protocols and cryptographic schemes ensure end-to-end data integrity	Only authorized devices can be connected to the Platform
Yes	Yes	Yes	Yes	Yes

DATA ANALYTICS

Implements real-time data analytics	Implements predictive data analytics	Implements data analytics for anomaly detection	Data analytics are included in a Marketplace	Data analytics are accessible using REST/SOA API calls
Yes	No	Yes	No	Yes

VISUAL INTERFACES

Visualization of data	Manipulation of data	Creation of analytics	Dashboard available for each user group (caregiver, patient, etc.)	Web application or standalone
Yes	Yes	Yes	Partial	WEB APPLICATION

Figure 12: The technical analysis of Onesait

3.3. Comparative analysis

In this paragraph the main characteristics emerging from the analysis of the features of each platform are compared. Each feature corresponds to a comparative table that allows an immediate glance at the similarities and differences between the solutions proposed by each platform analysed.

DATA ANALYTICS								
Platform	ACTIVAGE PROJECT	EKOSMART	FI-WARE	REACH 2020	sensiNact	onesait platform	UNCAP	universAAL
Implements real-time data analytics	Yes	Partial	Yes	Yes	Yes	Yes	No	No
Implements predictive data analytics	Yes	No	Yes	Yes	Yes	No	No	No
Implements data analytics for anomaly detection	Yes	No	Yes	Yes	Yes	Yes	Yes	No
Data analytics are included in a Marketplace	Yes	No	Yes	No	No	No	No	No
Data analytics are accessible using REST/SOA API calls	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No

Figure 13: Comparative table for Data Analytics

DEVICE MANAGEMENT								
Platform	ACTIVAGE PROJECT	EKOSMART	FI-WARE	REACH 2020	sensiNact	onesait platform	UNCAP	universAAL
Connectivity of heterogeneous IoT devices	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Remote access to IoT devices	Yes	Partial	Yes	Yes	Yes	Yes	Yes	Yes
Remote control of IoT devices	Yes	Partial	Yes	No	Yes	Yes	No	Yes
Devices lifetime management (software updates, remove bugs, fix security vulnerabilities)	Yes	Partial	Yes	Yes	No	Partial	No	No
Autodiagnostic features	Yes	No	Yes	No	No	Partial	No	Yes

Figure 14: Comparative table for Device Management

INTEGRATION/INTEROPERABILITY								
Platform	ACTIVAGE PROJECT	EKOSMART	FI-WARE	REACH 2020	sensiNact	onesait platform	UNCAP	universAAL
Interoperability is implemented using a semantic approach	Yes	No	Yes	No	No	Yes	Yes	Yes
Interoperability is implemented using a syntactic approach	Yes	Yes	Yes	No	Yes	No	Yes	No
Implements mechanisms able to make available platform services to third parties outside the Platform	Yes	Yes	Yes	Yes	Yes	Partial	Yes	Yes
Implements Restful / SOA web services mechanisms to access interoperability feature	No	Yes	Partial	Yes	No	Yes	Partial	Yes
Implements interoperability between services and functions defined inside the Platform	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes

Figure 15: Comparative table for Integration and Interoperability

INFORMATION SECURITY								
Platform	ACTIVAGE PROJECT	EKOSMART	FI-WARE	REACH 2020	sensiNact	onesait platform	UNCAP	universAAL
No sensitive data is shared with third parties unless it is a necessary part of the architecture, and has explicit consent from the data owner/user	Yes	Partial	Partial	Yes	No	Yes	Yes	Yes
Data is encrypted on the network	Yes	Partial	Yes	Yes	Yes	Yes	Yes	Yes
Communications between Platform to the Internet are secured	Yes	Partial	Yes	Yes	Yes	Yes	Partial	Partial
Protocols and cryptographic schemes ensure end-to-end data integrity	Partial	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Only authorized devices can be connected to the Platform	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Figure 16: Comparative table for Information Security

VISUAL INTERFACES								
Platform	ACTIVAGE PROJECT	EKOSMART	FI-WARE	REACH 2020	sensiNact	onesait platform	UNCAP	universAAL
Visualization of data	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Manipulation of data	Yes	Yes	No	No	Yes	Yes	Yes	Yes
Creation of analytics	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Dashboard available for each user group (caregiver, patient, etc.)	Yes	Yes	No	Yes	Partial	Partial	Yes	Yes
Web application or standalone	BOTH	WEB APPLICATION	WEB APPLICATION	BOTH	WEB APPLICATION	WEB APPLICATION	WEB APPLICATION	STANDALONE

Figure 17: Comparative table for Visual Interfaces

4. Analysis of contextual aspects

In this chapter, some details are provided, concerning each of the platforms analysed, concerning the aspects related to the legal framework, data governance, Ethics and IPR.

4.1. Definition of the contextual framework and areas

Digital Platforms can be understood as a method of organising digital markets by connecting two groups of users (suppliers and customers) facilitate services development and provision and providing adequate solutions to the needs of data sharing[6].

Platforms, such as the ones analysed in PlatformUptake.eu, are usually marketplaces – this means they have a triangular structure where users and service providers first have to establish contractual relationships with the platform owner, in order to be able to have contracts between themselves.

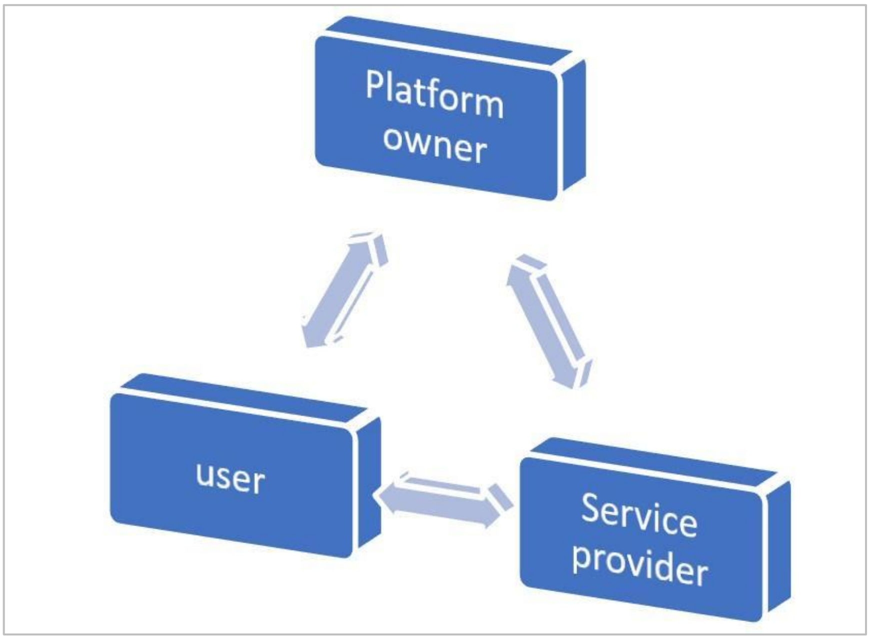

Figure 18: Platform relationships between stakeholders

One another angle, the platforms' activities are very dynamic, adapting themselves to the novelties and to consumer needs as they appear, which often changes the type of activities and therefore also its regulatory framing.

Despite the overarching framework of the Digital Single Market (DSM),[7] briefly detailed in the following section, there is still not a completely uniform landscape in what refers to legislative measures for the platform economy at the EU level, as there are Member States that already introduced national legislation, while others oppose to the idea of a single EU regulation - which would benefit from having a clear beneficial role, at least by clarifying the platform's status, clarifying the users' status and regulating reputational systems.

[6] https://www.europarl.europa.eu/RegData/etudes/BRIE/2017/607323/IPOL_BRI(2017)607323_EN.pdf
[7] https://ec.europa.eu/digital-single-market/en

Regulatory Framework

The Digital Single Market is the strategy of the European Commission that intends to enable the best possible access to the online world for individuals and businesses, through the free movement of persons, services and capital, allowing for engagement in online activities under conditions of fair competition, as well as consumer and personal data protection, irrespective of their nationality or place of residence.

The DSM Strategy was built on three pillars:

- Access: better access for consumers and businesses to digital goods and services across Europe;
- Environment: creating the right conditions and a level playing field for digital networks and innovative services to flourish;
- Economy & Society: maximising the growth potential of the digital economy.

While political measures to be taken amongst Member States are still being discussed towards digital platforms European legislation, online platforms are already subject to existing EU rules in areas such as competition, consumer protection, protection of personal data and single market freedoms. Compliance with these rules is essential to create a level playing field, and therefore the effective enforcement is crucial.

As EU directives related to consumer contracts and market regulation were not created focused on specific platform structures, they often are only partially adjusted or applicable to the different existing services. However, some can be detailed:

1. as European regulatory framework in the cases where consumers established contracts with the platforms:

- the Unfair Terms Directive

  https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:31993L0013&from=EN

- the Unfair Commercial Practices Directive

  https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32005L0029&from=EN

- the Consumer Rights Directive

  https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32011L0083&from=EN

- the future Digital Content Directive.

  https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32019L0770&from=EN

2. as market regulation directives applicable to platforms (both B2C and B2B):

- E-commerce Directive

  https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32000L0031&from=EN

- the Services Directive

  https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32006L0123&from=EN

- Misleading and Comparative Advertisement Directive

  https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32006L0114&from=EN

Nevertheless, they do not address platform-specific issues, and this is, thus, still a grey area in terms of legal framework, that is scattered and often inconsistent.

With this background, the European Law Institute (ELI) has drawn up a set of Model Rules that is meant as a contribution to the ongoing debate and provides a 'visualisation' of how a balanced approach could look, if regulatory action is considered necessary.

"With regard to these questions, the ELI Model Rules not only aim to consolidate existing European and national legislation, but also provide some innovative solutions for issues that could be addressed in forthcoming regulatory initiatives, in particular platform liability and reputation systems. These rules draw inspiration from European and national legislation, recent case law as well as other regulatory instruments such as international standards." They "are based on the premise that the existing rules of competition law are necessary, but not sufficient for ensuring fairness in the digital economy. These rules, which are applicable independently of any threshold regarding market power, are meant to complement antitrust rules."

However, despite the legal framework, many other features are relevant to better understand the contextual aspects of the platforms. Benchmarking from existing models and methods, including those detailed in the Methodology section, and also the OECD indicators, the following extended list was considered:

1. Legal and administrative context

   1.1. Administrative Burdens for Entry and Growth

   1.2. Safety, Health and Environmental Regulations

   1.3. Product Regulation

   1.4. Labour Market regulation

   1.5. Court & Legal Framework

   1.6. Procurement

   1.7. Reimbursement

2. Ethics and privacy

   2.1. Type of data collected

   2.2. Information provided

3. Data sharing and governance

3.1. Models: e.g. Economic; Citizenship; Trusted 3rd party; Collective

3.2. Data management

4. Intellectual Property Register (IPR)

4.1. Types: patents, trademarks, copyrights, and trade secrets

4.2. Open access, open source, close access

From this list, only some of the aspects were possible to consult from the available information (cf. figure in section 3.3) but all aspects may be of interest to be further explored in the subsequent tasks under WP2.

4.2. Contextual analysis of the platforms

4.2.1. UniversAAL IoT

UniversAAL IoT is a mature open platform for the integration of open distributed systems of systems. It was developed over 15 years, from general conceptual work in German research projects EMBASSI and DynAmITE, over proof of concept with unique problem solving approaches in FP6 PERSONA, consolidation and first tooling in FP7 universAAL, and stress testing in real life in CIP ReAAL, which has led to the creation of the initial universAAL IoT ecosystem.

UniversAAL IoT coalition (UIC) is a worldwide non-profit association based in Belgium since April 2018. However, it is not very active in present days.

UniversAAL is an Open platform (Open API, Open Scope, Open Source Licence, Open Provision, Open Operation, Open Adaptation). It operates under the Apache 2 Licence. It collects sensor data, data from services and applications and can work with data stored locally, and therefore does not need access to cloud solutions.

Due to the characteristic of being an enabler, the universAAL platform does not offer the specific service/application of an informed consent form. However, an application with the features would be supported by the capabilities of the platform. Similarly, there is no data sharing model implemented, however, such functionality would be supported if developed in an end-user application.

4.2.2. Activage_AIOTES

ACTIVAGE (ACTivating InnoVative IoT smart living environments for AGEing well) brings together 48 partners from 9 European countries with the objectives to build the first European IoT ecosystem across 9 Deployment Sites (DS) in seven European countries, reusing and scaling up underlying open and proprietary IoT platforms, technologies and standards, and integrating new interfaces needed to provide interoperability across these heterogeneous platforms.

The project delivers the ACTIVAGE IoT Ecosystem Suite (AIOTES), a set of Techniques, Tools and Methodologies for interoperability at different layers between heterogeneous IoT Platforms and an Open Framework for providing Semantic Interoperability of IoT Platforms for AHA, addressing trustworthiness, privacy, data protection and security.

ACTIVAGE ultimate goal is to create the evidence and to be the reference driver of this virtuous circle of the Active and Healthy Ageing market growth that will increase the demand by payers, providers and users, and will intensify the offer of solutions by the industry, SMEs and financial services.

It is establishing a European Multi Centric Large Scale Pilot distributed across nine interconnected Deployment Sites of seven European countries constituting the whole operational and evaluation space, in order to build local IoT ecosystems on top of legacy open or proprietary IoT platforms.

It is funded by H2020, currently CE marked and with no trademarks registered.

4.2.3. EkoSmart

EkoSmart is a platform, developed in Slovenia, to help develop a smart city ecosystem with all the supported mechanisms for efficient, optimized and gradual integration of individual areas into a unified and coherent system of value chains.

The development of the platform and accompanying modules and services was co-funded through the European Regional Development Found, between 2016 and 2019.

The EkoSmart program consisted of six projects contributing in their respective ways to the realization of the program vision:

- Research and development project No. 1 (RDP1) – Design of a smart city ecosystem
- Research and development project No. 2 (RDP2) – Smart mobility.
- Research and development project No. 3 (RDP3) – Active living and wellbeing.
- Research and development project No. 4 (RDP4) – E-health and mobile health.
- Research and development project No. 5 (RDP5) – Integrated health services.
- Research and development project No. 6 (RDP6) – Solution prototypes.

One of the important features of the EkoSmart program is the integration of the solutions in different areas into a common ecosystem, delivering a platform with the same name (EkoSmart platform) which allows easy integration of sector-specific solutions into a common ecosystem (featured in the program, as well as others) facilitating, as such, the identification and support of inter-sectoral value chains. This platform is compatible with global solutions and includes Internet of Things (IoT) features.

The main purpose of the platform is to easily connect and integrate multiple external services and provide users with unified UI experience. Therefore, the data collected and retained depends on the services that users use. By using a specific service, the user has to agree to the terms and conditions specified by the service. Some data about the users is however stored on the platform in order to make it easier for the user to use for example username, user information, billing information and such.

Serves provided on the platform are proprietary and subject to the service provider licences.

4.2.4. Reach2020

Reach2020 is a platform developed in Europe, co-funded under the H2020 EU project with the same name.

The REACH project aimed to develop a service system that turn clinical and care environments into personalisable modular sensing, prevention, and intervention systems that encourage older adults to become healthy via activity (physical, cognitive, mobility, personalized food, etc.).

The project abbreviation stands for Responsive Engagement of the Elderly Promoting Activity and Customized Healthcare. The proposal for this project was developed in 2015 and submitted under pillar 3 of H2020 in societal challenge 1 Personalized Healthcare (PHC).

In the European consortium with 17 partners from higher education institutions and industry, the four EuroTech Universities along with the industry partners (including leading European health care technology, rehabilitation, and care and hospital firms) build the core of this project, with a total grant amounts around 6 Million Euros.

Although it collects Personal Health Data & History, Personal Feedback Data, there are no open datasets coming from the platform. All data is kept in internal repositories. In particular, when data sets shall be shared beyond the Consortium or published, each Consortium member will, as per Consortium Agreement, have the right to double check whether the stated set considered for publishing/sharing may lead to any conflict in the context of planned patent filing.

The Consortium is now seeking for patent protection. The Reach2020 platform is partially based on Open Source Platforms, with some elements closed and proprietary services such as those provided through the Health Suite Digital Platform (HSDP) by Philips.

4.2.5. Sensinact

Sensinact is one of the open source projects established under the Eclipse Foundation. The Foundation is home to the Eclipse IDE, Jakarta EE, and over 350 open source projects, including runtimes, tools, and frameworks for a wide range of technology domains, such as the Internet of Things, automotive, geospatial, systems engineering, and many others.

The Eclipse Foundation was created in January 2004 as an independent not-for-profit organization supported by over 275 members who value the Foundation's governance model, open innovation processes, and community-building events. Members include industry leaders who have embraced open source as a key enabler for business strategy.

The Eclipse Project was originally created by IBM in November 2001 and supported by a consortium of software vendors. The Eclipse Project has been used by millions of developers.

The Eclipse community consists of individual developers and organizations spanning many industries. The Foundation employs a full-time professional staff to provide services to the community. The Eclipse Foundation is funded by annual dues from members and governed by a Board of Directors. Strategic Developers and Strategic Consumers hold seats on this Board, as do representatives elected by Add-in Providers and Open Source committers. Eclipse committers are typically employed by organizations or are independent developers that volunteer their time to work on the Eclipse projects.

The Eclipse Foundation provides four key services to the Eclipse community: 1) IP Management, 2) Ecosystem Development, 3) Development Process, and 4) IT Infrastructure.

An important aspect of the Eclipse Foundation is the focus on enabling the use of open source technology in commercial software products and services. This is made possible by the fact that all Eclipse projects are licensed under the Eclipse Public License (EPL), a commercial friendly OSI approved license.

The Eclipse Foundation also undertakes a number of steps to attempt to ensure the pedigree of the intellectual property contained within Eclipse projects. The first step in the due diligence process is trying to ensure that all contributions are made by the rightful copyright holder and under the Eclipse Public License (EPL). All committers are required to sign a committer agreement that stipulates all of their contributions are their original work and are being contributed under the EPL. If a committer is sponsored to work on an Eclipse project by a Member organization, then that organization is asked to sign a Member Committer Agreement to ensure the intellectual property rights of the organization are contributed under the EPL.

The second step is that the source code related to all contributions which are developed outside of the Eclipse development process is processed through the Eclipse Foundation IP approval process. This process includes analysing selected code contributions to try to ascertain the provenance of the code, and license compatibility with the EPL. Contributions that contain code licensed under licenses not compatible with the EPL are intended to be screened out through this approval process and thus not added to an Eclipse project. The end result is a level of confidence that Eclipse open source projects release technology that can be safely distributed in commercial products.

4.2.6. UNCAP

UNCAP certification suite was developed in the project UNCAP, co-financed by the EU through Horizon 2020.

The UNCAP ("Ubiquitous iNteroperable Care for Ageing People ") project was co-financed by the Horizon 2020 programme, involving 23 partners (including several pilot user partners) from 9 European countries (IT, UK, SI, RO, EL, DE, SE, ES, MK), between 2016-2018.

UNCAP made use of solutions and technologies developed in previous research projects to develop an open, scalable and privacy-savvy ICT infrastructure designed to help aging people live independently, while maintaining and improving their lifestyle. The final solution consist of real products that will be made available on the market.

UNCAP currently delivers a platform based on open industrial standards, able to create new care & assistance paradigms, through an open source, scalable and privacy-savvy ecosystem ready to help aging people live independently

Being based in Italy and France, UNCAP is compliant to European regulations. Based on the devices UNCAP connects to and the services offered, it stores data of different nature such as: sensor data, medical devices, scores, metrics, schedules, personal health information, login, policies, operational, atl@ante and location.

UNCAP is categorized as open specification and open software components, with parts which are licenced from project partners. UNCAP BOX, the UNCAP CLOUD and the UNCAP APP are released as open source. However, elements of the so called UNCAP core system as well as enabling technologies provided by different technology partners are so called background IP, thus constituting pre-emptive rights, which use and application has to be licensed.

4.2.7. FiWare

The FIWARE platform FIWARE initially emerged as a European project[8], but nowadays it is being maintained by the FIWARE Foundation and supported by an open source community.

The FIWARE Technical Steering Committee (part of the FIWARE Foundation) governs the technical direction of the FIWARE platform and activities of the FIWARE Open Source Community. The FIWARE Foundation members are: Atos (Spain), Engineering (Italy), Orange (France), and Telefónica (Spain). The FIWARE webpage[9] provides bylaws about FIWARE software, the FIWARE Foundation and the ecosystem. There is also a Code of Conduct webpage[10] with information on compliance, confidentiality, integrity, commitment, among others, of the FIWARE Foundation.

The types of data collected by a FIWARE platform instance can vary greatly depending on the specific solution enabled by the platform. In the AHA domain, the data might include the details of end users, the measurements or other data stemming from IoT devices, etc. The same is the case regarding the Information provided to the user on data collection, storage, processing and transfer, or any requirements regarding informed consent. These all depend on the use case implemented on top of the FIWARE platform. If needed, use cases or applications requiring such features can be supported by FIWARE, but some of these features may need to be built by the solution's developers.

FIWARE also offers a marketplace, which is based on the FundingBox Platform. Analytical information on data collection, storage, processing and transfer of data as well as the terms and conditions of use can be found in the marketplace legal and privacy webpage[11].

The "FIWARE" name is a registered trademark[12] of the FIWARE FOUNDATION, E.V. All FIWARE components (GEs) are Open Source and provided royalty-free. In specific, it seems that most (if not all) GEs are made available under the GNU Affero General Public License v3.0.

4.2.8. Onesait

The Onesait platform is developed based on the outcomes of previous projects:

- January 2009 - March 2012: Sofia Artemis Project (https://artemis-ia.eu/project/4-sofia.html)
- March 2012 - June 2019: Sofia2 by Indra (discontinued, replaced by Onesait Platform)
- June 2019 – currently: Onesait Platform (https://onesaitplatform.atlassian.net/)

In the Platform, data is the main and permanent asset. The Data-Centric approach turns data into its core. There is a data model, a semantic data model, and each application functionality reads and writes through the shared model.

The platform has integration with different Cloud services such as Azure, GCP, Amazon and allows these services to be used transparently (eg Kubernetes service in Azure or AWS, Google BigQuery, Azure Intelligence Services, ...).

---

[8] https://cordis.europa.eu/project/id/285248
[9] https://www.fiware.org/foundation/bylaws/
[10] https://www.fiware.org/foundation/code-of-conduct/
[11] https://marketplace.fiware.org/legal/privacy
[12] https://trademarks.justia.com/863/86/fiware-86386998.html

Regarding external communications, everything is done through TLS and SSL (both incoming and outgoing). All communication between platform's microservices can also be configured to be done via HTTPS, so that the containers specify the Sidecar pattern that is responsible for establishing security.

The platform supports various authentication and authorization mechanisms, using OAuth2 by default. The platform also offers credential lifecycle management.

The platform is built on a large number of widely tested and standard open-source components in its field. The commitment to Open Source is such that the platform is also completely open-source and is released on Github.

The Data Governance Model of Onesait rests on the following principles:

- Completeness of the information: Ensuring that the standardized data as a whole has logic for its exploitation.
- Global business / infrastructure vision: Establishment of the business definition for the data, allowing them to be identified through an end-to-end vision, as well as with an infrastructure vision that allows knowing the physical location of the data.
- Responsibility for data: Assignment of responsibilities and roles regarding information management, reliability, integrity, provisioning and exploitation of data, through a governance model that facilitates the monitoring of its evolution.
- Efficient data: Compliance with the principles of non-duplication, integrity and consistency of data.

KEY ELEMENTS IN QUALITY ASSURANCE

- Definition of standards: Generation of minimum information requirements to consider the data as correct (verification of lengths, type of data, formats).
- Data validation: Establishment of validation mechanisms that allow the integration of the data in the storage infrastructure according to established standards, minimizing the occurrence of incidents.
- Cross view of the data life cycle: Identify the complete process of the data life cycle, allowing its global management through the traceability and mapping of information.

4.3. Comparative analysis

The following figure answers to the questions drawn for the contextual analysis, as depicted in Figure 3 and allows a comparative analysis of the platforms.

COMPARATIVE CONTEXTUAL ANALYSIS								
Platform	ACTIVAGE PROJECT	EKOSMART	FI-WARE	REACH 2020	sensiNact	onesait platform	UNCAP	universAAL
REGULATORY FRAMEWORK								
In which country is it based?	Spain	Slovenia	Germany	Germany	France	Spain	France & Italy	Belgium
Is there information available on the regulations it complies with?	Yes	No	Yes	Yes	Yes	No	Yes	Yes
Is it a medical device? Is it certified? CE mark? Other?	No | CE mark	No	No	No	No	No	No	No
Is there information available on how it was funded?	H2020	ERDF	FP7 Now Fiware Foundation	H2020	Eclipse Foundation	mindsait Indra company	H2020	FP6 FP7 National Funding
ETHICS AND PRIVACY								
Type of data collected	Personal and sensitive	Personal and sensitive	Personal and sensitive	Personal and sensitive	Personal and sensitive	Personal and sensitive	Personal and sensitive	Personal and sensitive
Information provided to the user on data collection, storage, processing and transfer	Yes	Yes	Yes	Yes	Yes (depends)	Yes	N/A	Yes (depends)
Is there an informed consent?	Yes	Yes	Yes	Yes	N/A	No	N/A	No
IPR								
Is the platform registered?	No	No	Yes	No	No	No	No	No
What is the access model?	Not yet defined	Closed access	Open Source and provided royalty-free. Main license scheme: GNU Affero General Public License v3.0	Partially based on Open Source Platforms, with some elements closed	Open source, Open access	Open source, Open access	Open specifications and open software components, with parts which are licenced form project partners IPR	Open platform (Open API, Oopen Scope, Open Source Licence, Open Provision, Open Operation, Open Adaptation)

Figure 19: Comparative analysis of the contextual aspects

5. Analysis of financial aspects and business models

Due to the lack of consistency of available methods implemented to analyse the business models of existing open source platforms in the AHA and AAL domains, the next part of this document seeks to address this challenge by applying the Business Model Canvas in the case of the 8 platforms selected, based on the approach outlined in section 1.2.

5.1. Definition of the financial aspects and business models

The results of the analysis in this chapter will complement the description of the platforms' features and characteristics by depicting a more comprehensive picture of their financial and business aspects.

The choice of the Business Model Canvas as a tool is based upon the assumption that it allows to visualise, in a more focused way, the interdependent relations between the different characteristics of a business concept and considers those which matter most and have the greatest impact on driving growth. In particular, it helps one identify the competitive advantages as well as the weaknesses of the platforms' business models under consideration and provide in the course of their analysis useful insights on possible factors that have supported or contributed to their current state.

Following the assessment logic outlined by the Business Model Canvas (BMC) (Figure 21), the document seeks to present information covering the four main areas of the platforms' business models: infrastructure, offering, customers and finances. Each of the areas consists of building blocks, accounting for a total of nine, describing different components, players and functional correlation among these. At the core of the BMC is the value proposition. The right side shows how values are created for customers and the company. The left side of the BMC, on the other hand, documents what resources a company needs to successfully implement the business model [1].

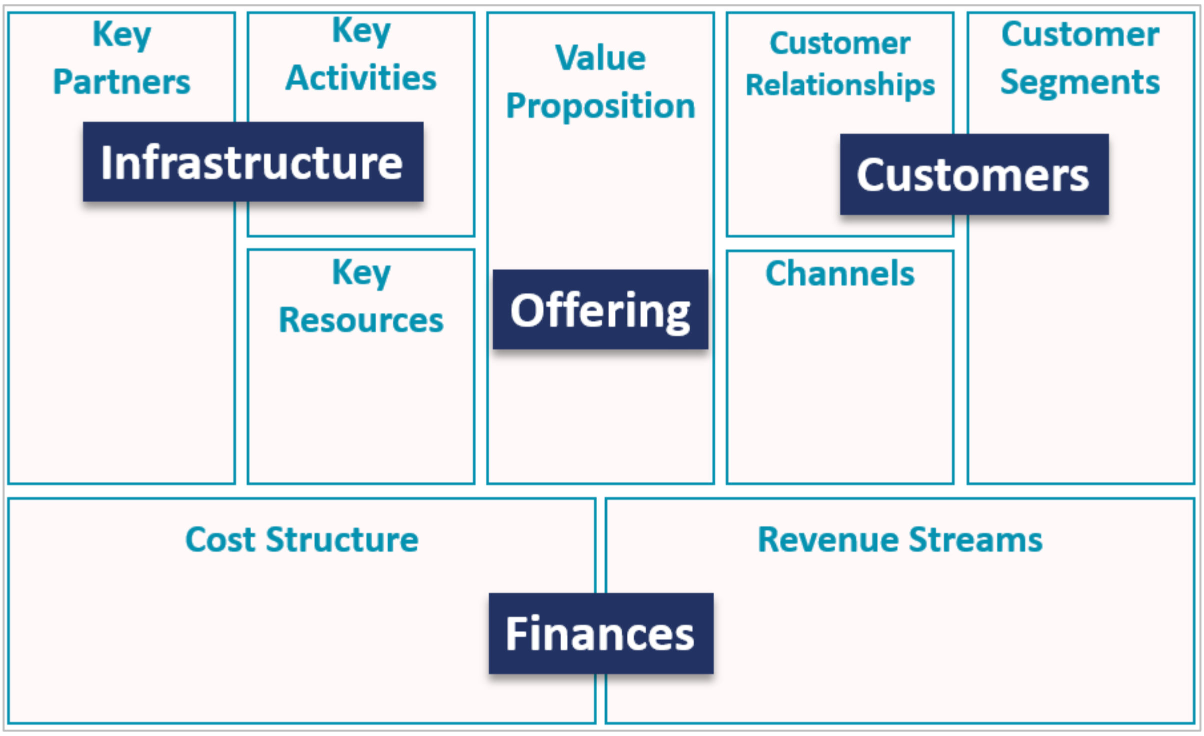

Figure 20: Business Model Canvas

Taking into account the theoretical basis presented above, the following chapter will seek to study the business models of the platforms under consideration based on the available information acquired through existing social and professional networks as well as a desk research targeting data openly accessible on the internet.

5.2. Business analysis of the platforms

5.2.1. UniversAAL IoT

The universAAL IoT platform resulted from multiple projects co-funded by the FP6, FP7 and CIS EU programs. The consortium was composed of various partners from academia, industry, civil society and research involved in development and piloting of the platform. The development of the platform built upon the outcomes from projects such as PERSONA. During the implementation of the project ReAAL, 30 applications were ported to be compatible with universAAL. These were used by 13 pilot sites across Europe where universAAL was applied as standalone in combination with a cloud solution. As an outcome of the project the non-profit association UniversAAL IoT coalition (UIC)' was founded to bridge the open source community of universAAL to the market and thus ensure sustainable use of the created services beyond the project.

universAAL is an open source platform that enables ICT industry in Europe to rapidly develop and deploy innovative AAL solutions. The platform provides the end users with a fully customized experience, accelerating and growing the potential for interconnectivity by ensuring that new features and applications are regularly added to integrated systems.

The unique value proposition evolves around the communal actions facilitated by universAAL where all compatible products, services and devices can connect instantly and collaborate to the benefit of the end user that could otherwise have not been achieved in isolation. As a living IoT system, the platform grows with each application being developed and thus contributing to a vast library of resources available automatically to all the end users.

Master repositories for the source code, as well as wikis and issue management are all currently hosted in the GitHub service. Further on, the universAAL platform offers a variety of security functionalities as part of the various layers of the platform, partly relying on existing security features provided by the underlying systems. Grouped authorized universAAL nodes exchange messages encrypted by the group's shared key. Permissions can be defined for each message type. Communication with the world outside the group can be secured using the standard public key infrastructure. The platform also offers support to all of its users providing a free release history and version upgrade instructions including Javadoc for application developers and documentation on reported suggestions and issues resolved by the community. In the course of its implementation universAAL platform showcased its capabilities on pilot sites across Europe.

The total costs of the project ReAAL amounted to 10.775.000 € whereas 4.994.000 € from the budget were spent for 534 PM on developing applications and rolling out the platform at 13 different pilot sites. Future revenue streams are to be generated through service sale as a part of the ReAAL project goals.

5.2.2. Activage AIOTES

ACTIVAGE is a European multi centric large-scale pilot on smart living environments building the first European IoT ecosystem across 9 Deployment Sites (DS) in seven European countries. The project has received funding from the European Union's Horizon 2020 research and innovation programme in the amount of 25.129.823,71€ and has a consortium consisting of 49 partners from seven countries, 10 of which are industrial, 14 leading research institutes or universities, 11 major corporations plus 15 highly innovative SMEs, coordinated by Medtronic Iberica. Activage is also part of the IoT European Large-Scale Pilots Programme collaborating to foster the deployment of IoT solutions in Europe in a usage context, as close as possible to operational conditions.

The ACTIVAGE value proposition as an exploitation and commercialization hub is a service supporting the deployment of solutions towards products for the AHA domain. The platform creates scenarios of transference near to real conditions for the future commercial exploitation of the project results. To achieve this goal the project foresees the involvement of third parties engaged in "open calls" to provide solutions to specific issues. The platform allows SMEs to design, develop and test solutions as well as cities to deploy full ACTIVAGE technology and services. Engagement is strategic action facilitated by the platform seeking to enable the consolidation of the ecosystem and its sustainability.

The key assets of the platform are ACTIVEAGE offers AIOTES, Tools, marketplace, AHA services and knowledge that are available in a professionalized central repository with its appropriate governance policies and operation. An "Exploitation Vehicle" takes care of the ownership of ACTIVAGE legacy and impulses the evolution of the ecosystem after the end of the project for the next five years. This exploitation vehicle is called "ACTIVAGE-ORG" and highlights the difference between ACTIVAGE-Project and the organization that will take care of the future exploitation of the project legacy.

The customer segment of the platform consists mainly of researchers, service providers as public administration, health and social care agencies and payers, application and services developers.

5.2.3. Ekosmart

The project was initially funded by European Regional Development Fund and aimed at Slovenian market, but is currently being developed and extended to provide the ability to be used internationally, starting with Italy and Austria.

The purpose of the EkoSmart program is to develop a smart city ecosystem with all the support mechanisms necessary for efficient, optimized and gradual integration of individual areas into a unified and coherent system of value chains. The program focuses on three key domains of smart cities: health, active living and mobility; and forms strategic relationships with municipalities and other areas of smart cities, such as energy, smart buildings, involvement of citizens, smart communities, etc. Ekosmart introduces the universal architecture of a smart city, based on the combination of self-learning and self-optimizing agents enabling the realization of all the concepts of smart cities, such as interoperability, self-adaptivity and self-configurability, open data, semantic interoperability, and integration of social capital.

The EkoSmart is founded on high quality and varied consortium of the most advanced partners in these areas, with strategic links to smart home and health programs. The consortium consists of 25 partners from various fields: two research institutions (Anton Trstenjak Institute of gerontology and intergenerational relations and Jožef Stefan Institute), five faculties at two universities (UL – Faculty of Electrical Engineering, UL – Faculty of Computer and Information Science, UL – Faculty of Sport, UL –

Faculty of Medicine, and UM – Faculty of Electrical Engineering and Computer Science), five other public institutes (University Clinic Golnik, National Institute of Public Health, University Medical Center Ljubljana, URI-Soča, Adolf Drolc Healh Care Centre), ten small and medium sized enterprises (Cosylab, Alpineon d.o.o., Elgoline d.o.o., Inova IT d.o.o., Marand d.o.o., Nela razvojni center d.o.o., RC IKTS d.o.o., Robotina d.o.o., SRC sistemske integracije d.o.o., Špica International d.o.o.), and three large companies (Iskra d.d., Medis d.o.o., Telekom Slovenije d.d.).

The value proposition of EkoSmart lies in the ambition to integrate solutions in different areas into a common ecosystem. This provides users with better user experience and choice of which services and products he or she wants to use and subscribe to. Even bigger benefits exist for the service providers as the platform not only represents a place where they can provide their products but also allows them to easily integrate other services into their own product making it better and increasing its value. Too often the practice of introduction of smart cities shows limited focus on certain areas and lacks connection with others. One of the important objectives of the EkoSmart program is therefore the development of the platform with the same name which allows easy integration of sector-specific solutions into a common ecosystem and will, as such, facilitate the identification and support of inter-sectoral value chains. This platform will be compatible with global solutions and will include concepts such as the Internet of Things (IoT).

Compared to other similar solutions in the field, EkoSmart is placed on electronic and mobile health offering self-configurable, self-integrating, self-optimizing, flexible and adaptable universal architecture with simple addition of modules. The platform allows intensive development and implementation of new ICT methods and concepts, such as IoT and methods of artificial intelligence as a backbone of technology and human society development. Emphasis is put on the concept of smart specialization – the introduction of interconnected comprehensive chains, and markets.

In terms of economy, the vision of the EkoSmart program is to launch Slovenian solutions in the field of smart cities on the world market. The realization of this vision is based on several major approaches namely the concentration of knowledge and experience, focus on the user, evolutionary development and flexible architecture.

5.2.4. Reach2020

The Reach2020 platform is co-funded by the H2020 EU program. The project total cost was € 6. 078.657,50 (with an EU funding contribution of € 4.588.315). The consortium is composed of 18 partners that collaborated in the development of the platform and its services. One of the partners of the consortium, among others, is Philips, which integrated its HealthSuite Digital Platform (HSDP) with the Reach2020 platform planned for patent protection. The rest of the consortium is represented by: Technical University of Munich, Germany; Technical University of Eindhoven, The Netherlands; École Polytechnique Fédérale of Lausanne, Switzerland; Technical University of Denmark; University of Copenhagen, Denmark; Fraunhofer, Germany; Lyngby Taarbæk Kommune, Copenhagen, Denmark; Schön Klinik, Bad Aibling, Germany; HUG, Switzerland, Geneva; Zuidzorg, Eindhoven, The Netherlands; Biozoon, GmbH, Food Innovations, Bremerhaven, Germany; Sturrm, Business Modeling & Strategic Planning, Eindhoven, The Netherlands; Smart Cardia, Wearables, Software & Technology, Switzerland; Alreh Medical, Rehabilitation Equipment, Poland; Arjo, Rehabilitation Equipment, Sweden and DIN, Standardization, Germany.

The Reach2020 business model is developed based on four use case settings (SK, HUG, ZZ, and Lyngby), with the attempt of defining an initial business strategy and vision of deploying the platform and making use of in each of the four use case countries, from the testbed level to the specific market conditions on the national level, and evolving later to the overall Reach2020 business model at the EU level. The Reach2020 platform value proposition advances around five main touchpoints: personal mobility device, active environment, socializing and nutritional monitoring, gaming and training, and finally wearables.

The stakeholder model concentrates on the older people as end users. Other stakeholders considered are caregivers and clinics or institutions they work for. Distinctively different is the role of the app/platform in the contexts. Where in the rehab clinic the app is the gateway for cooks, caregivers and nutrition experts towards the elderly user and groups of users to ask for help preparing food and monitor food intake, the app is an additional component in the home context for users to contact one another and share recipes or inspiration on food.

Reach2020 is currently preparing the formation of a “REACH Active Ageing GmbH” which will serve beyond the project as an integrator of REACH partner’s individual products and services and a solution provider to above named market segments. Key assets of its business model at national levels are the collected data through the sensing system and developed data analytics algorithms aimed for the analysis of the aggregated information.

The project total costs amounted for € 6.078.657,50 (with an EU funding contribution of € 4.588. 315). The generation of revenue is based upon a fee paid per user or by the (local) government, insurance companies or clinics and institutions. Due to the health improvements the service guarantees back money that would otherwise be spent in the health sector. Furthermore, the platform will generate revenue by distilling insights from data collected and monetizing this data to third party developers in the health, food and nutrition industries. The platform is therefore open to further development and plugins by third parties to enhance the overall PSS. However, since the needs for using the different functionalities of the platform touchpoint and engine concept can grow in the process of aging, the Reach2020 business model supports personalized on-demand usage instead of offering all functions at the same time. Therefore, both the system architecture and the business model need to be modularized according to the different stakeholder network configurations and reimbursement mechanisms in the national markets.

5.2.5. Sensinact

The project lead of Sensinact is CEA-Leti: Laboratoire d'électronique des technologies de l'information, a French institute creating innovation with the goal to pass these on to the industry aiming to improve the quality of life worldwide. Parties which have expressed interest in collaborating with the project are Osaka University, Keio University, Engineering Ingegneria Informatica spa, ACUTUS Software, Inc., JRISS, Kyoto Sangyoo University and STMicroelectronics. Eclipse has been created to bring the community of developers and other smart city stakeholders together with the aim to tackle challenges such as, among others, economic crisis, security threats, natural disasters and ageing society with an open approach seeking to foster a smart city business ecosystem.

The core value proposition of the Sensinact project lies in the ambition to enable the collection, processing and redistribution of any data relevant to improving the quality of life of urban citizens and the programming of interfaces allowing different modes of access to this data (on-demand, periodic, historic, etc.). Finally it seeks to facilitate the application development and deployment to easily and rapidly build innovative applications on top of the platform.

Based on comprehensive communication strategy CEA has reached out to end users, developers and project partners transferring technology to the local industry through the various offices established all over France. To the platform ecosystem belong also numerous industrial and academic partners.

At the heart of Sensinact is its service-oriented approach in which IoT devices expose their functionalities in terms of services (temperature service, presence detection service, air quality monitoring service, alarm service, etc.). Each service then exposes one or several resources such as sensor data or actions. Building applications thus become a matter of composing sensing services with actuation services. Four elements build the core service offering of the platform.

With the lack of a de facto standard data model today in the IoT domain, Sensinact adopts a generic and extensible data model to facilitate building adapters for various protocols. Its core model is based on four types of resources: sensor data, action, state variables, and properties. Those resources are accessible by generic and easy to use Application Programming Interfaces (API) providing synchronous (on demand) and asynchronous (periodic or event based) access to data and actions of IoT devices, as well as access to historic data.

5.2.6. UNCAP

The UNCAP ("Ubiquitous iNteroperable Care for Ageing People ") project was funded by the H2020 program and received a total sum of 3.000.000 €, involving 23 partners (including several pilot user partners) from 9 European countries (IT, UK, SI, RO, EL, DE, SE, ES, MK). Out of the 23 consortium partners, the company NIVELY has emerged to continue offering the outcome product. UNCAP delivers a complete system including software, hardware and a cloud-service. It contains the UNCAP Box, UNCAP App, UNCAP Cloud. The company NIVELY was identified and selected to operate UNCAP as a separate business line, commercializing UNCAP project results and licensing the needed technologies from the selected UNCAP consortium members.

The value proposition of UNCAP lies in the development of an open, scalable and privacy-savvy ICT infrastructure making use of solutions and technologies developed in previous projects to help aging people live independently while maintaining and improving their lifestyle. The final solution consists of real products that will be made available on the market.

The services which are offered by UNCAP to the end users and are directly related to the value proposition are the UNCAP Access Electronic Health Record (EHR) and Elderly Monitoring including fall detection and fall prevention, motion detection, device localization, red button activation, dispatching of notifications and interaction with the environment. To take the products and services of the platform to the domestic market the pilot user partners took into account the feasibility for a startup as possible.

Important stakeholders are the Healthcare services providers and care professionals, SMEs (CIO, CTO of companies, with infrastructure capacity, responsible for planning, infrastructure, sales, communication), distributors, hospices and private doctors, nursing homes, municipalities and governmental entities, companies doing home care services, end users and relatives. One of the most important key stakeholders are the relatives, caregivers and persons in care, which belong to the caregiver group and user group respectively. These are the individuals that purchase and/or use the UNCAP products and services, thus playing a major role. Pharmacies and small and medium sized enterprises make up another core group of the key stakeholders, since they provide, and use UNCAP products as well.

Since eCare services are more and more co-financed by care insurances technologies in the field, they are difficult to afford if customers are on their own. To tackle this challenge, the project consortium took under consideration the implementation of a flat rate model. This would allow customers or care institutions for example to use a leasing model, where fixed payments are paid each month. Another possibility being elaborated on is to rent the services provided by the platform which is considered to be a bit more flexible since the payments can vary depending on e.g. whether the service was used or not. Leasing is another option to generate revenue and fits specifically well the business segment, since tax advantages can be utilized, and the general monthly fee would most likely be lower compared to renting. On the other hand renting eCare products and services over the platform would particularly well fit the customers, due to the possibility of terminating the rent agreement at short notice.

5.2.7. FiWare

The presented business information uses text from various parts of the FIWARE website[13].

In terms of key partners, there are more than 325 partners involved with the FIWARE Foundation, which range from Platinum members (i.e., the core FIWARE Foundation members), Gold members, Gold strategic end users and associates. Apart from the above partners, there are also some institutional, community and media partners listed as key partners of the FIWARE Foundation. The FIWARE Community further comprises all individuals and organizations contributing towards achieving the FIWARE Mission. The FIWARE Community is not only formed by contributors to the technology (the Open Source Community working on the FIWARE platform), but also those who contribute in building the FIWARE ecosystem and making it sustainable over time. The FIWARE website provides a full list of members[14].

There are five (5) key activities of the FIWARE Foundation:

- To empower the developers to bring best-of-breed tools to write great code, manage the development process and benefit from quality validation processes, end users to access great software easing the development of new solutions, making the best use of the, being able to provide feedback and benefit from large-scale testing facilities, and companies and other organizations (who are FIWARE Foundation members) to organize all sorts of events, information, discussions and other activities and resources.

- To promote the new technologies integrated in FIWARE, fostering their adoption as de-facto standards, the developers producing such new technologies, the offering built with or around FIWARE Technologies, the users of the FIWARE Technologies who build such offerings, the FIWARE ecosystem as a catalyst for economic opportunities.

- To augment increasing interfaces for connectivity, consolidating or promoting standards, increasing use cases for new application domains

- To protect the FIWARE trademark, the compliance with the FIWARE Code of Conduct, FIWARE Technologies by ensuring they remain available as Open Source, the openness, meritocracy and transparency which guides the decision making

[13] https://www.fiware.org/
[14] https://www.fiware.org/foundation/members/members-list/

- To validate the labelling qualifying the Quality Assurance of the FIWARE Technologies, organizations that own the expertise to validate "Powered by FIWARE" solutions, "FIWARE IoT-ready" devices or people/organizations capable to provide FIWARE development, integration, training and consulting services.

The key resources of the FIWARE Foundation are its more than 325 members from more than 35 countries, the more than 150 open source projects, the more than 160 "Powered by FIWARE" solutions, the 8000 developers using FIWARE technologies, the more than 100 evangelists, the 16 FIWARE iHubs, the 10 FIWARE Lab Nodes, and its 15 Strategic partnerships.

The offered value proposition is an open sustainable ecosystem around public, royalty-free and implementation-driven software platform standards that will ease the creation of Smart Applications in multiple sectors. Using FIWARE technologies, organizations can capture the opportunities that are emerging with the new wave of digitalisation brought by combining the Internet of Things with Context Information Management and Big Data services on the Cloud, and developers can gather context information at large scale from many different sources. FIWARE also helps to easily process, analyse and visualize managed context information, easing the implementation of the smart behaviour and the enhanced user experience required by next-generation Smart Applications.

The main customer segment are public or private organizations and their associated developers who are interested in context data management in order to facilitate the development of smart solutions in various domains such as health, cities, industry, agri-food, energy, etc. End users are also the beneficiaries of the developed smart solutions.

The main channels are the FIWARE Community, Events (such as Hackathons, Conferences, Workshops, Fair, Summits, Info Sessions), the Accelerator programme and the Marketplace. In specific, FIWARE was built thanks to the joint efforts of different actors and now goes a step further in the creation of a community to gather web entrepreneurs, mentors, investors, students, academia, industry and public sector innovators to keep progressing with three goals in mind: 1) Expand the reach of FIWARE at a global level, 2) Set new innovation hubs around the world, 3) Create a European environment of innovative business hubs. Sixteen (16) FIWARE Accelerators exist, where funding is provided for the most talented teams and business proposals building upon FIWARE technology. FIWARE organizes or participates at Hackathons, Conferences, Workshops, Webinars, Fairs and Summits throughout the world, where the audience can meet the FIWARE people and technologies. The FIWARE Marketplace serves the purpose of globally disseminating existing commercial offerings around FIWARE. It is a global one-stop shop that gives visibility to a wide range of solutions/ platforms powered by FIWARE, FIWARE-ready technologies as well as FIWARE related training/coaching or consultancy, integration and support services.

Apart from the aforementioned endeavours focusing on getting, keeping and growing their customer base, FIWARE Foundation are also offering the following free services towards improving their customer relationship: a) FIWARE Lab, a non-commercial sandbox environment of the FIWARE Community, which offers the capability to innovate and experiment with the FIWARE Technologies. Entrepreneurs 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. b) FIWARE Academy, a list of video tutorials, slide decks and other training materials available for developers learning about the FIWARE Ecosystem. In addition to the training materials for each individual GE, there are some integrated courses that have been recorded.

Regarding costs, since the platform is provided for free, the cost is mainly associated with the resources needed to a) install, configure and deploy the platform components and build the required functionality on top of it (fixed cost), and b) run the platform components (fixed or monthly per-user cost, dependent on whether the platform will be deployed on local or cloud resources). Hence the cost is highly dependent on the use case requirements in terms of functionality and the user base to be served. The more FIWARE GEs are used the more VMs need to be employed (according to the specification of each GE there might be a recommendation for 1 GE per VM), and the greater the user base is the more resources should be allocated to each VM, hence the greater the cost.

Finally, the main revenue stream seems to be the subscription fees associated with becoming a member of the FIWARE Foundation[15].

5.2.8. Onesait

Onesait Platform is owned by Indra Company and provides the flexibility so that developers can build their own solutions in a solid and agile way using Open Source technologies, a flexible architecture and an innovative approach.

The value proposition of the platform evolves around the open source components covering the entire life cycle of information (from ingest to visualization through its process and analysis). It offers a unified web console for the development and operation profiles of the solutions. Thanks to different assistants and the encapsulation of best practices, it is possible to develop systems with complex architectures in a simple way, reducing cost and time to market

Onesite offers a data centric architecture which is the main and permanent asset. There is a data model, a semantic data model, and each application functionality reads and writes through the shared model. Further on, the platform capabilities such as publishing, viewing and messaging are based on a microservices architecture. This ensures concept's isolation between different parts and offers flexibility for the customization of different capabilities. Moreover, the platform is designed to speed up complex systems development based on unified version, business centric development, software lifecycle management and flexible development. Further, the platform's services can be deployed where preferred, on any public or private cloud or own CPD (Physical or VM). Onesait is built on a large number of widely tested and standard open-source components in its field. This reduces the curve of use of these technologies for new platform users, where it also offers tools that make it easier and more productive to use them. The platform is also completely open-source and is released on GitHub.

Core end-users of the platform are developers who have access to the systems, information, tools and resources of Onesait. More than 100,000 companies in around 90 countries worldwide are connected with the Onesait ecosystem.

Onesait Platform has two editions, one is Community Edition (Open Source) and the other one is Enterprise Edition (commercial support). Onesait Platform Community edition is a free. Onesait Platform Enterprise edition represents the premium paid version of the platform.

[15] https://www.fiware.org/foundation/members/membership-fees/

5.3. Comparative analysis

Considering the descriptions provided above, the following figure highlights the main similarities and differences in terms of the business analysis of the platforms analysed:

COMPARATIVE BUSINESS ANALYSIS								
Platform	ACTIVAGE PROJECT	EKOSMART	FI-WARE	REACH 2020	sensiNact	onesait platform	UNCAP	universAAL
Value Proposition	Exploitation and commercialization hub. Service supporting the deployment of solutions towards products for the AHA domain.	Combination of multiple services in one ecosystem, so the end user doesn't need to register, subscribe or manage each service individually.	An open sustainable ecosystem around public, royalty-free and implementation--driven software platform standards that will ease the creation of Smart Applications in multiple sectors.	A service platform that turns clinical and care environments into personalisable modular sensing, prevention, and intervention systems that encourage the elderly to become healthy via activity (physical, cognitive, mobility, personalized food, etc.).	Enabling the collection, processing and redistribution of any data relevant to improving the quality of life of urban citizens and the programming of interfaces allowing different modes of access to this data (on-demand, periodic, historic, etc.).	Open source components covering the entire life cycle of information (from ingest to visualization through its process and analysis). It offers a unified web console for the development and operation profiles of the solutions.	Open, scalable and privacy-savvy ICT infrastructure making use of solutions and technologies developed in previous projects to help ageing people live independently while maintaining and improving their lifestyle.	Open source platform that enables ICT industry in Europe to rapidly develop and deploy innovative AAL solutions.
Costumer Segment	Researchers, service providers as public administration, health and social care agencies and payers, application and services developers.	Private organisations, public organisations, road users, family members, older and chronically ill people.	Public or private organizations and their associated developers.	Older persons at home still independent and older persons at rehabilitation (patient in rehabilitation with stroke, dysphagia, etc.).	Developers and project partners.	Developers.	Caregivers and persons in care, Pharmacies and small and medium sized enterprises .	Developers, service providers .
Revenue Streams	Yes	Upcoming	Yes	Yes	Yes	Yes	No	No

Figure 21: Comparative analysis of the business aspects

6. Success and hindrance factors

In this chapter we provide an overview of the main success and hindrance factors of online platforms, based on the information collected in the 3 main dimensions of technical, contextual and business factors.

For some of these dimensions there is further information that may be collected and analysed in the subsequent tasks within PlatformUptake.eu. However, with the data available, a potential use of the Critical Success Factors model is approached for further development and measurement in T2.3 and T2.4.

The Harvard Business Review (9,11) displays three factors for the success of a platform strategy:

- Connection: how easily others can plug into the platform to share and transact
- Gravity: how well the platform attracts participants, both producers and consumers
- Flow: how well the platform fosters the exchange and co-creation of value

Harvard Business Review (11) studied in 2019 the reasons why most online platforms fail.

"The average life of the failed platforms is only 4.9 years. Many gig economy platforms collapsed within 2-3 years because they did not have enough users or funding. Given the need for deep pockets, it should not be surprising that standalone firms tended to have shorter lives than those that were acquired or launched as part of a larger firm or consortium of firms. Standalone firms had an average duration of only 3.7 years. Acquired firms, which generally had stronger balance sheets, were capable of fighting longer (averaged 7.4 years), while firms that were part of larger entities were just average in length of survival."

On the side of barriers, literature (2, 11), offers several possibilities:

- Mispricing on one side of the market (which side of the market should be charged, and which should be subsidized is the strategic question)
- Failure to develop trust with users and partners
- Prematurely dismissing the competition by being too arrogant or overconfident
- Entering too late the market when the competition is already far ahead
- Entry of newcomers on the online platform market affects the market share.
- Effective entry does not appear to be less likely in more concentrated digital markets.
- Concentration tends to increase over time in each sector, but competition from other sectors often intensifies.
- Online platform market shares tend to be fragile, limiting the extraction of material rents – most platforms offer their services to users for free and it seems even platforms with a large market share would lose most of their users if they introduced even a modest user fee.
- Innovation seems to persist among online platforms, even in concentrated markets.

- Traditional competition concerns (e.g. contractual arrangements) still can be relevant.
- Bundling and ties to other products and services – the effect of this practice is likely to vary based on the cost of switching to another service.
- Requirements for platforms, for example social networks have an expensive technical component in addition to the variable costs of moderation.
- Requirements for providers can create a barrier.
- Barriers to multi-homing: costs play a role.

These barriers are rather generic and more illustrative for big online platforms such as Uber, Amazon, Microsoft, Apple than platforms in the niche field of the AHA domain. Nevertheless they may learn general lessons on online platforms in the field of AHA and its special target group and aims. For example:

a) the lesson of focusing on users' trust by providing a secured data management platform especially for sensitive AHA data.

b) the lesson of pricing: when and what are users willing to pay for services provided through AHA platforms.

c) the package: providing AHA products and information in combination with other products might increase the success of platforms if costs allow the switch.

The above mentioned success and hindrance factors will be used as the background knowledge to be elaborated while developing the interviews that will take place in T2.3.

Towards Critical Success Factors

For the aggregation of the results of the analysis performed and the presentation of a scheme that can illustrate and serve as framework for the continuation of the works within PlatformUptake.eu, it was decided to use as inspiration the framing of **Critical Success Factors** (CSF's). This is a term originated from data and business analysis and refers to the key factors or activities required for ensuring the success of a business activity. As a definition, critical success factors refer to "*the limited number of areas in which satisfactory results will ensure successful competitive performance for the individual, department, or organisation*"[6]. Identifying CSF's is important as it allows firms to focus their efforts on building their capabilities to meet those aims.

The principle of identifying critical success factors as a basis for determining the information needs of managers was proposed by Daniel Ronald [5] as an interdisciplinary approach with a potential usefulness in the practice of evaluation within library and information units but popularized by John Rockart [3]. There are four basic types of CSF's:

1. **Industry CSF's** resulting from specific industry characteristics;
2. **Strategy CSF's** resulting from the chosen competitive strategy of the business;
3. **Environmental CSF's** resulting from economic or technological changes; and
4. **Temporal CSF's** resulting from internal organisational needs and changes.

Table 5: Types of Critical Success Factors

The Industry	There are some CSF’s common to all companies operating within the same industry. Different industries will have unique, industry-specific CSF’s
Competitive strategy and industry position	The nature of the position in the marketplace or the adopted strategy to gain market share gives rise to CSF’s differing strategies, and positions have different CSF’s
Environmental Factors	Economic, regulatory, political, and demographic changes create CSF’s for an organisation.
Temporal Factors	These relate to short-term situations, often crises. These CSF’s may be critical but are usually short-lived.

Each CSF should be measurable and associated with a target goal. A critical success factor is not a Key Performance Indicator (KPI) but these indicators will quantify the objectives and enable the measurement of strategic performance.

Based on the CSF framework and making a critical analysis of the outcomes of the in-depth analysis previously presented, PlatformUptake.eu identifies four success criteria, namely: **efficiency, effectiveness, fulfilment of the functional requirements, and stakeholder satisfaction**.

These criteria are formulated based on three dimensions: technical, contextual and business. Further, one supplementary dimension is considered as overarching or transversal: the resources. Stakeholders are also considered, and all areas are represented in Figure 22.

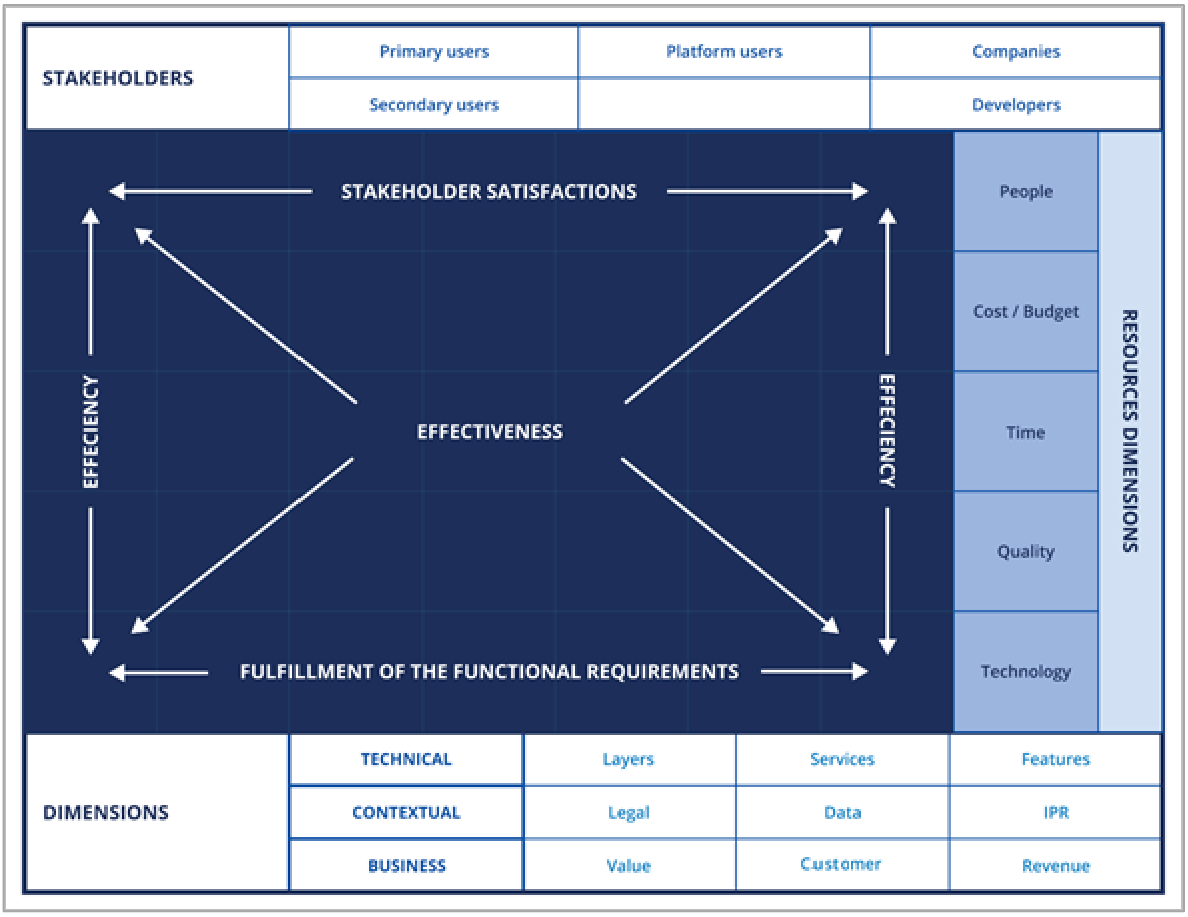

Figure 22: Critical Success Framework for open AHA platforms

7. Conclusion

In this document all the activities related to Task T2.2 "Observe common and differentiating features and characteristics of existing platforms that can act as success or hindrance factors in their uptake" have been described. The main findings of the technical, contextual and business analyses were combined with the Critical Success Factors Model and will be used as the basis for the development of the subsequent analytical tasks within WP2.

8. References

Literature:

[1] Alexander Osterwalder & Yves Pigneur, Business Model Generation, Published by John Wiley & Sons, Inc., Hoboken, New Jersey. P.44

[2] Andrew Lilico and Matthew Sinclair at Europe Economics on behalf of the Department for Business, Energy and Industrial Strategy, Dynamic Competition in Online Platforms. Evidence from five case study markets, March 2017.

[3] Chief executives define their own data needs. (1979) Rockart, John F. Harvard Business Review, Mar/Apr79, Vol. 57 Issue 2, p81-93, 13p

[4] EIP-AHA (2018). Caregivers' role in Information and Communication Technologies development and exploitation for Age-Friendly Environments. Accessed in 2020/05/26 https://en.caritascoimbra.pt/wp-content/uploads/sites/3/2018/10/CaregiversRoleonICTforAFE_WhitePaper.pdf

[5] Management information crisis (1961) Daniel, D. Ronald. Harvard Business Review, Sep/Oct61, Vol. 39 Issue 5, p111-121, 11p

[6] McKee, James (2012) Applying Principles from IT Architecture to Strategic Business Planning https://books.google.pt/books?id=9r6eBQAAQBAJ&printsec=frontcover&hl=pt-PT#v=onepage&q&f=false

[7] OECD (2020). Healthy life years statistics. Accessed in 2020/05/26 https://ec.europa.eu/eurostat/statistics-explained/index.php/Healthy_life_years_statistics

The Healthy Life Years indicator (HLY) is a European structural indicator computed by Eurostat. HLY measures the number of remaining years that a person of a certain age is expected to live without disability. It is actually a disability-free life expectancy.

Websites:

[7] https://hbr.org/2013/01/three-elements-of-a-successful-platform [accessed 27.05.2020]

[8] https://executive.mit.edu/blog/8-ways-to-launch-a-successful-digital-platform [accessed 27.05.2020]

[9] https://hbr.org/2016/03/6-reasons-platforms-fail [accessed 27.05.2020]

[10] https://dspace.mit.edu/handle/1721.1/99112 Platform failures: lessons for strategic management

[11]https://hbr.org/2019/05/a-study-of-more-than-250-platforms-reveals-why-most-fail

[12] https://dspace.mit.edu/handle/1721.1/76685: Failure to launch: Critical Mass in Platform Businesses.

[13] About universAAL: https://github.com/universAAL

[14] About Activage AIOTES: www.activageproject.eu

[15] About Ekosmart: http://ekosmart.net/en/

[16] About Reach2020: https://mediatum.ub.tum.de/doc/1482713/1482713.pdf; https://reach2020.eu/

[17] About Sensinact: https://projects.eclipse.org/proposals/eclipse-sensinact

[18] About UNCAP: http://www.uncap.eu/; D4.5 – Analysis of impact on business and financing models, http://www.uncap.eu/downloads/

[19] About Onesait: Onesait Platform Developer Portal https://onesaitplatform.atlassian.net/wiki/spaces/OP/overview?mode=global

9. Appendix A - Glossary

Record Name	Definition
Platform	It is an operating environment, under which various applications and service programs from the application layer are designed, implemented, tested, released and maintained.
Project	It is defined as an undertaking that is temporary in nature for the purpose of creating a product, service, or result. A project must have a definite beginning and end (not ongoing). It can be over a very short or very long duration but must remain a temporary endeavour.
Framework	It is a software providing high-level functionalities through application-specific software. It refers to a collection of libraries/classes with the common goal of providing a scaffold on which to build software and identifies the whole set of services that are part of the service layer. Frameworks might completely alter how you implement your program, or they might just speed up common tasks.
Ecosystem	It is a collection of different elements related to a platform and the exploitation of its results. The elements, or parts, can include people, hardware, software, facilities, organisations, policies and documents. Their roles and the way they interact with each other arise from a mutually beneficial purpose, such as commercial gain, innovation or common interest.
Stakeholder	Individual or organization having a right, share, claim, or interest in a platform or in its possession of characteristics that meet their needs and expectations; N.B. Stakeholders include, but are not limited to end users, end user organizations, supporters, developers, producers, trainers, maintainers, disposers, acquirers, customers, operators, supplier organizations and regulatory bodies. (ISO/IEC June 2010).
Platform developer	It is the individual or group of individuals, typically hired by *End user customer*, that follow and implement the entire life cycle of the applications, or more generally, of the products, deriving from a given platform.
Platform end user	It is the individual or groups of individuals intended as the main beneficiaries of an application or set of applications provided by the considered platform. It contains *Primary end user (AHA)* group together with non-AHA beneficiaries and platform developers.
Primary end user (AHA)	In the AHA domain it is the single individual intended as the main beneficiary of a service or set of services provided by the considered platform. The primary end users directly benefit of these services with an increase of their quality of life.
Secondary end user (AHA)	The person or organisation, such as formal and informal care persons, family members, care organisations and their representatives, who are in direct contact with a primary end user. This group uses a set of

	applications or services provided by the platform to grant to the primary end users an increase or maintenance of their quality of life.
End user customer	Institutions and private or public organisations that are not directly in contact with products and services, but who somehow contribute in organising, paying or enabling them. This group includes the public sector service organisers, social security systems, insurance companies.
Physical layer	Layer responsible for the recognition and exchange of messages with physical devices and sensors. It implements a high-level communication interface that allows the upper levels easy access to the devices, dealing with raw communication. It is the building block for the service layer.
Service layer	Layer which provides sets of methods that allow access to devices and a first data processing in the context of a specific domain. It is the building block for the Application Layer.
Application layer	Layer in which methods developed in the service layer are combined to create applications.
Semantic layer	Layer in which a knowledge system (e.g. ontology) is defined in order to give a formal representation and provide a natural interface for accessing the functionalities of an underlying layer. It usually is the building block of the interoperability layer.
Interoperability layer	Layer which provides a common interface for accessing and exporting application functionalities from/to different platforms.
Application	It is a software that, by typically combining multiple services, offers to the user a unique, or restricted for a specific purpose, application experience. An application can also include a user interface (graphic, textual, touch). Other important elements of an application are the degree of permissiveness of the license, dependence on the operating system, being standalone or network, to be installed or portable, etc. From a user perspective, elements like accessibility, user-friendliness, look-and-feel, availability, support, maintenance, etc. have to be considered.
Software	Collection of instructions that tell the computer and its related devices how to work.