Project Title:	**Sensing and predictive treatment of frailty and associated co-morbidities using advanced personalized models and advanced interventions**
Contract No:	690140
Instrument:	Collaborative Project
Call identifier:	H2020-PHC-2014-2015
Topic:	PHC-21-2015: Advancing active and healthy ageing with ICT: Early risk detection and intervention
Start of project:	1 January 2016
Duration:	36 months

D1.3

FrailSafe technical specifications and end-to-end architecture

Due date of deliverable:	M12 (31th December 2016)
Actual submission date:	31th December 2016
Version:	1
Date:	17th November, 2016

Lead Author:	Luca Bianconi (SIGLA)
Lead partners:	Luca Bianconi, Cristiana Degano, Fabio Podda, Matteo Toma (SIGLA), Vasilis Megalooikonomou, Dimitrios Vlachakis, Ioannis Ellul (UoP), Marina Kotsani, Athanasios Benetos (INSERM), Javier Montesa, Luisa Perez Devesa (BRAINSTORM), Roberto Orselli (SMARTEX), Kosmas Petridis, Hariklia Zografou, Panagiotis Sabatakos (HYPERTECH)

Horizon 2020
European Union funding
for Research & Innovation

Change History

Ver.	Date	Status	Author (Beneficiary)	Description
0.1	17/11/16	Draft	SIGLA	Definition of deliverable Table of Contents and partners' tasks
0.10	09/12/16	Draft	SIGLA	First integrated version release, to be reviewed by all the contributors
1.0	19/12/16	Release	SIGLA, UoP	Second integrated version release to be published
1.1	20/12/16	Release	SIGLA, BRAINSTORM, SMARTEX	Third integrated version
1.2	21/12/16	Release	UoP	Revisions to the integrated version
1.3	22/12/16	Release	SIGLA	Forth integrated version
1.4	22/12/16	Release	UoP, CERTH, INSERM	Fifth integrated version
1.5	23/12/16	Release	SIGLA	Final version

EXECUTIVE SUMMARY

On the basis of the outcomes of T1.4 activities, the document hereby has been prepared. Its main objective is to describe, in good details, the high-level overall system architecture as a whole and all the components, both hardware and software, composing it.

This is the first official release of the system architecture and it is planned, by the project work plan, to present it revised in a document (see D1.4) later on (M24) in the life of the project to include all the corrective actions or modifications that will be applied in order to accomplish the objectives of the project at best.

DOCUMENT INFORMATION

Contract Number:	H2020-PHC–690140	**Acronym:**	FRAILSAFE
Full title	Sensing and predictive treatment of frailty and associated co-morbidities using advanced personalized models and advanced interventions		
Project URL	http://FrailSafe -project.eu/		
EU Project officer	Mr. Jan Komarek		

Deliverable number:	1.3	**Title:**	FrailSafe technical specifications and end-to-end architecture
Work package number:	1	**Title:**	Requirements, Use Cases, Architecture and Specifications

Date of delivery	**Contractual**	**31/12/2016 (M12)**	**Actual**	27/12/2016
Status	Draft ☒		Final □	
Nature	Report ☒	Demonstrator □	Other □	
Dissemination Level	Public □	Consortium ☒		
Abstract (for dissemination)	This deliverable reports on the choices done in the design of the FrailSafe system – and sub-systems – technical specifications and architecture (Task 1.4). Firstly, is given an overall introduction to the system concepts, modules and processes; secondly a more detailed presentation of different layers composing the system architecture - devices, front-end interfaces, server back-end infrastructure - is presented in all its parts; finally the integration and deployment approaches are depicted, followed by the system requirements estimated compulsory for guaranteeing a suitable quality of service. This Deliverable benefits of and refers to elements coming from D1.1, D1.2, D3.1.			
Keywords	frailty, frailty classification, wearable solutions, wireless communication, physiological monitoring, cognitive monitoring, older adults' social inclusion, sarcopenia			

Contributing authors (beneficiaries)	Luca Bianconi, Cristiana Degano, Fabio Podda, Matteo Toma (SIGLA), Vasilis Megalooikonomou, Dimitrios Vlachakis, Ioannis Ellul (UoP), Marina Kotsani, Athanasios Benetos (INSERM), Javier Montesa, Luisa Perez Devesa (BRAINSTORM), Roberto Orselli (SMARTEX), Kosmas Petridis, Hariklia Zografou, Panagiotis Sabatakos (HYPERTECH)			
Responsible author(s)	Luca Bianconi, Cristiana Degano, Fabio Podda, Matteo Toma		**Email**	FrailSafe @grupposigla.it
	Beneficiary	SIGLA	**Phone**	+39 010 589635

TABLE OF CONTENTS

FIGURES

1. INTRODUCTION

The FrailSafe project, along and in accordance with its medical research objectives, aims at implementing a platform and a system able to better understand frailty and develop quantitative and qualitative measures to define frailty and its relation to co-morbidities. In this context, the goal is the development of an ICT-based solution through the usage of a modular sensors network that will deliver rehabilitation, and ultimately lead to prediction, prevention and self-management of frailty symptoms.

The resulting product of this work is a platform to monitor and – if possible – delay and reduce the arising of frailty among the population using the FrailSafe system, based on the extensive usage of software, with the aim of: a) collecting and processing the data, b) putting in place a sensors network and c) creating a mobile ecosystem of applications and games for smartphones and tablets.

In this direction, this document aims at describing in detail the architecture of the FrailSafe system in order to define and explain all the technical specifications on both the implementation criteria and the requirements.

The Chapters of the deliverable will describe the platform from different points of view:

- The system description in terms of functionalities that each part will provide;
- The architecture of the system and its overall system requirements, giving a detailed description of each part that compose the architecture itself;
- The interactions and communication between the different modules and/or the overall system.

The technical requirements, also known as non-functional requirements, will be also described. Non-functional[1] requirements that we will address are:

- Security;
- Privacy;
- Reliability;
- Scalability;
- Usability.

The document is organized as follows:

Chapter 2 describes the overall FrailSafe ecosystem and identifies the functionalities of each part composing the ecosystem itself;

Chapter 3 depicts in short the high-level architecture of the system. The explanation of the contents follows the logical organization of the architecture. Thus it crosses all the different layers composing the platform, detailing all the single sub-modules for each tier. A general description and the interactions among modules are underlined;

Chapter 4 illustrates the general approach to the integration of the different sub-systems and which means are adopted to make them securely and efficiently exchange information;

Chapter 5 addresses the security and privacy problems, describing how to cope with them, but also additional non-functional requirements.

[1] A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system rather than specific behaviours, since they correspond to functional requirements. Therefore, it refers to all requirements and keeps information described or functions to be performed.

2. FRAILSAFE SYSTEM DESCRIPTION

In this chapter, the higher-level parts of the system will be presented and described. The following image (Figure 1) shows an overview of the FrailSafe system architecture with all the modules that are described in this section.

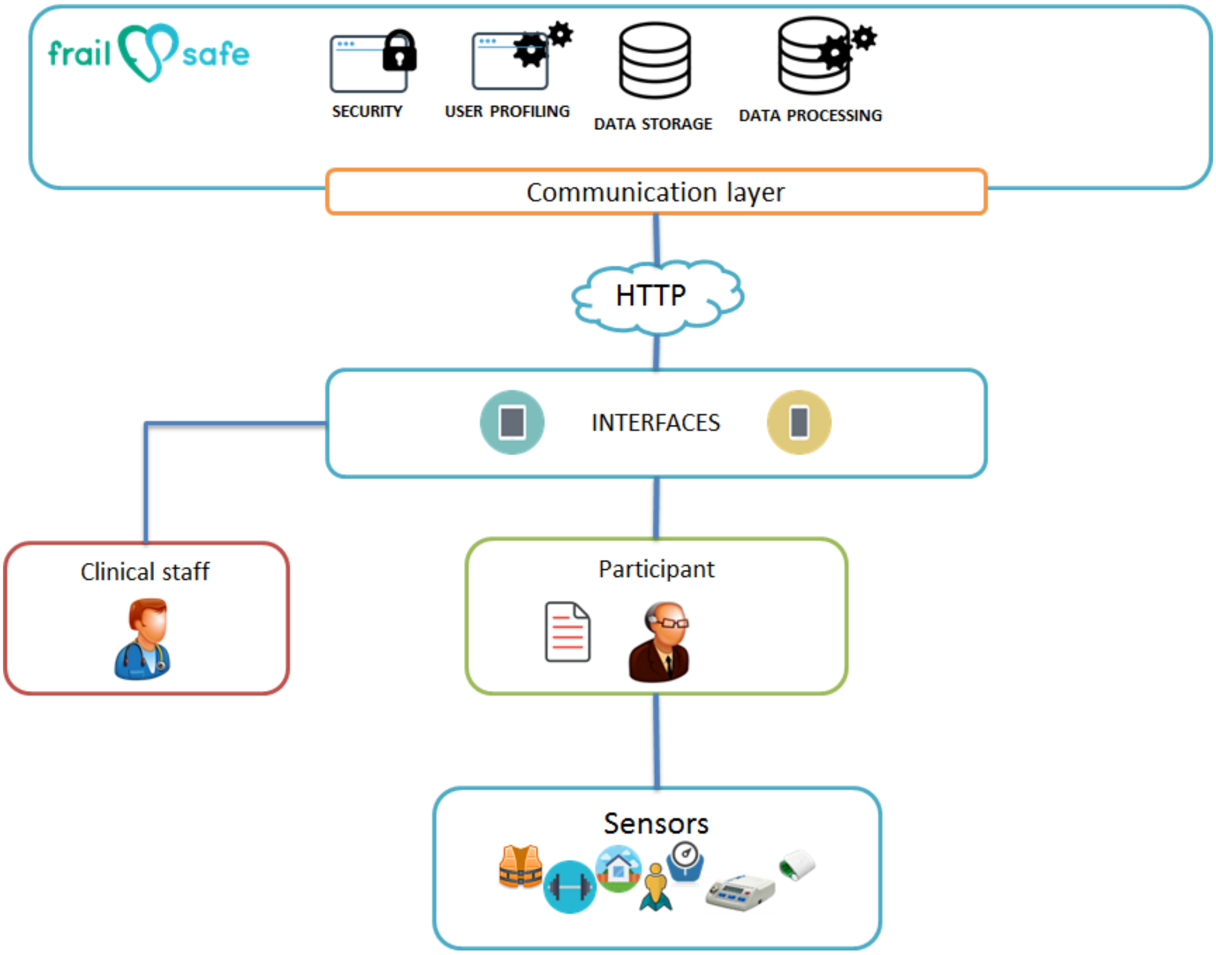


Figure 1 - FrailSafe architecture overview

As can be seen from the picture above, the principal components of the FrailSafe system are essentially five:

- the **Participant,** that is the person participating to the medical study and the one monitored and treated by the platform. These people are the main actors of the medical research;
- the **Sensors layer**, including all the FrailSafe sensors and mobile devices that can be used by the FrailSafe users (participants),. These devices can be mobile (smartphone and tablet), wearables (WWBS) as well as additional medical sensors (blood pressure monitor, digital scale and others) to assist the indoor and outdoor monitoring of the users;
- the **Clinical staff**, that collects some of the data while visiting the participants, visualizes some of the produced data, gives support to participants and performs the medical research on top of all this data. Even if the main actor of the system must be considered the *participant*, the importance of the clinical staff has not to be neglected

and they can be easily recognised as co-participants to the role of the end-user of the system;

- the **FrailSafe interfaces,** that are composed by a number of devices, sensors and software used by the two categories of users above;
- the **FrailSafe Communication Layer including also the cloud-based** back office **components (data processing, data storage, security and privacy, profiling)**, that are needed for the FrailSafe system and are hosted in remote servers. Thus, this *FrailSafe layer* is in charge of retrieving, elaborating and storing the data coming from the **FrailSafe interfaces**.

Even if in this document the system is put in the middle of FrailSafe panorama for better explaining its features and solutions, it is important to underline since the beginning of this document that the USERs – and in particular the so-called PARTICIPANTS – are the real centre around whom the system and the services are conceived, designed and developed. Every part of FrailSafe, from sensors to mobile applications, from web portals to web services is designed according to User Centred Design approaches, as described in details in Deliverable D1.2.

2.1. FrailSafe Sub-Systems description

At a very high-level, FrailSafe can be described as made of three separated, but at the same time tightly connected, parts:

1. firstly, the sensors used by the participants to produce data;
2. secondly, a number of main front-end interfaces having the goal of creating the communication link between the users and the whole platform infrastructure;
3. thirdly, the back-office systems hosted remotely for providing the secure and reliable set of functionalities through which FrailSafe delivers its services to the users.

The picture shown in Figure 2 depicts the high-level schema of the system as it has just been described.

Each single component that is part of FrailSafe will be described in the following Chapters but - before getting into the details of such single elements - it is worthy to give in this Chapter's sub-sections (see 2.1.1, 2.1.2, 2.1.3) a short introduction to each individual part of the system.

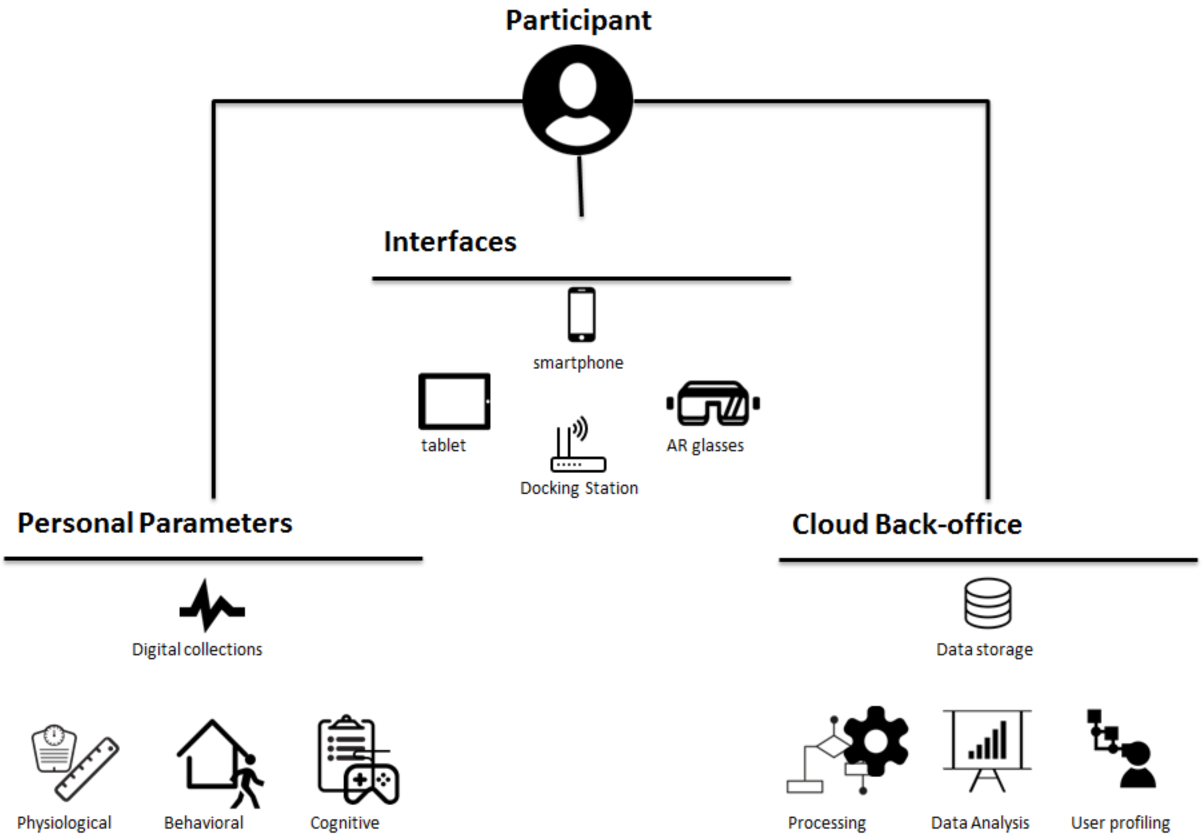


Figure 2 - System parts

2.1.1. Sensors

A set of sensors will be used with the aim of producing the adequate types and amounts of data on each participant as it is needed to reach the objectives of the project. They collect most of the *personal parameters* of a participant.

Even if they are analysed individually in depth within the Deliverable D3.1 – *"Sensor components and communication strategy"*, it is however useful to have a short reference list of them in this document for easing the general understanding of the system processes and components. They are:

- A *sensorized garment*, worn by the participant;
- A *dynamometer,* used by the clinical staff during the visiting process and by the participant during the autonomous game playing;
- A network of Bluetooth Low Energy transmitters, so-called *Beacons*, installed at participant's home to monitor its indoor behavioural patterns;
- A digital *blood pressure monitor*, meant to be used by the participant autonomously as well as with the help of the clinical staff;
- A digital impedance *scale*, meant to be used by the clinical staff;
- The *GPS*, used for monitoring - when needed - the outdoor habits;
- A *pulse waves monitor* that is used by the clinical staff;

The details about how each of these devices participates to the system composition and how they interact with the user and the system is analysed later in this document.

2.1.2. Interfaces

The users – i.e. *the participant* and the *clinical staff* – interact with the system through the mean of three interfaces:

- The ***smartphone***, that plays a major role for the platform since it is probably the most frequently used interface among the ones adopted. It is conceived to be the central connection point among the user, the sensors and the services provided by the system;
- The ***tablet***, that mostly provides the interaction between the user and the system relying on the usage of the games installed on it;
- The ***AR glasses***, having a similar role to the one of the tablet but providing a very particular user experience;
- The **docking station**, that is foreseen to be part of the system after the end of the project for letting the *participants* upload data and charge the *sensorized garment* at the same time.

These devices are used to provide different services of the platform and to face different tasks. They are described in good detail with reference to their context and mode of usage in the following chapters.

2.1.3. Cloud

The third part of the system is represented by what can be called the *FrailSafe Cloud*, that is where all the server-side sub-systems of the platform are deployed and where they are working.

This is the core part of the system and it is where most of the data are securely stored, processed and where the new data triggering actions towards the users are generated.

The *FrailSafe Cloud* is composed of several modules and sub-systems documented - in respect to their role in the overall architecture and of the interactions they have - in the next pages.

3. SYSTEM ARCHITECTURE

In this chapter, the FrailSafe system architecture will be detailed and discussed in order to give a more precise description of the platform and to give an overview of the technologies to be adopted for each block. The system requirements will be also discussed.

In the definition of the FrailSafe system architecture, the focus is to maintain it as modular as possible, both horizontally and vertically. This is a strategic choice, since the system is composed by very different pieces that have to communicate with each other. Another benefit of maintaining a modular architecture is the scalability, which is a very important factor in this project since the number of users could grow rapidly and so, having a system that can easily scale, represents a key property that the system must have.

The following picture (Figure 3) shows a more refined structure of the FrailSafe system architecture, focusing on users and on the main modules that compose the FrailSafe cloud.

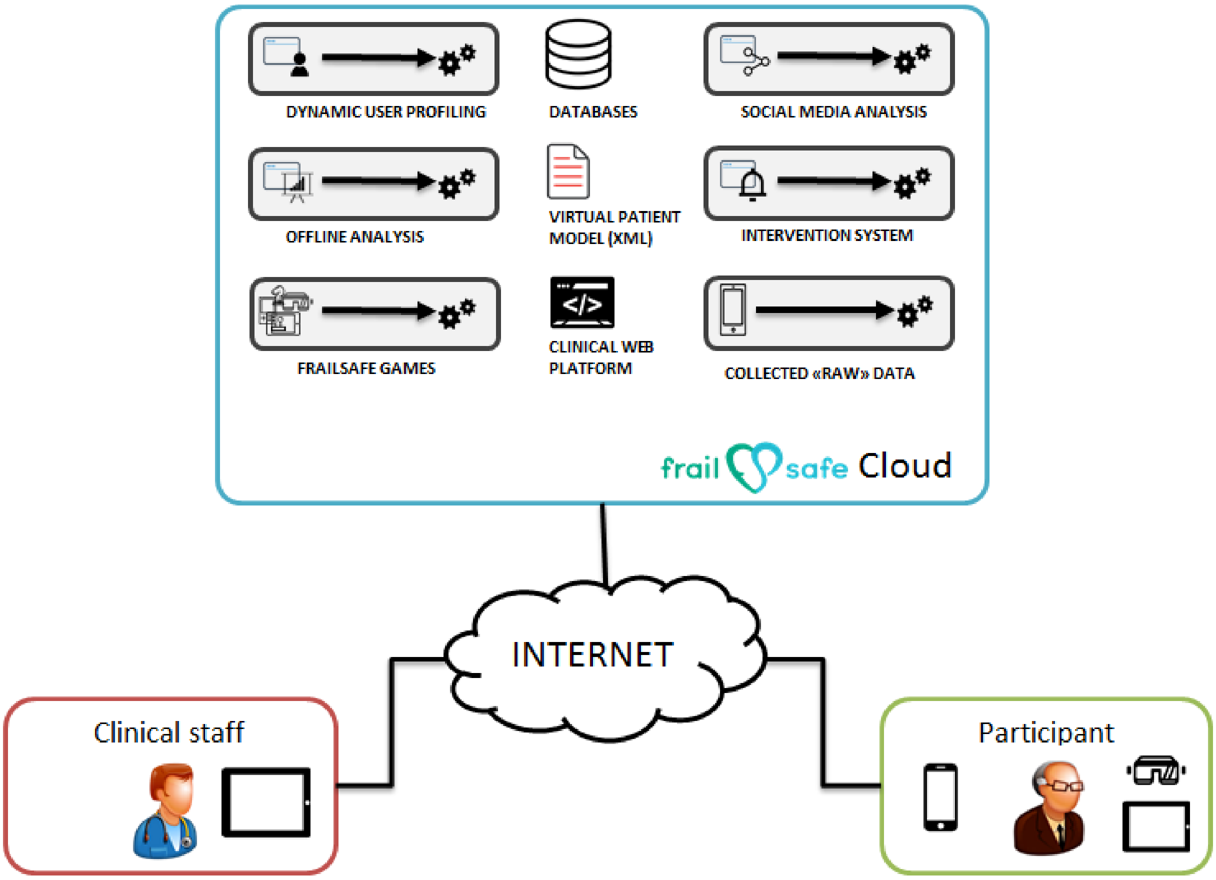


Figure 3 - FrailSafe System Cloud modules overview

As more synthetically anticipated previously in this document – see FRAILSAFE system description - in the picture above it is important to notice the main interactions occurring in the system:

- the *Participant*, who is the main system end-user, has different sensors and software collecting its physical, physiological and cognitive data; all this information is collected by the so-called *FrailSafe Gateway* and sent to the back-office services working remotely. *Participants* can also interact with the system using some applications installed on their mobile devices (monitoring applications on smartphone, cognitive games on tablet);

- the *Clinical Staff*, that is in charge of collecting further data and parameters regarding various aspects of participants' life during a number of scheduled visits, interacts with the platform by accessing a web application through a web browser (see section *3.3.2.3*).

The structure of the system is articulated to work as a series of modular services interacting among each other with the aim of giving the end-users (participant and clinical staff) access to the system functionalities.

It is also important to consider that the architecture has various degrees of approximation in respect to its parts; in fact, participant's sensors data could be retrieved in different ways and with different level of accuracy.

In this respect, we can define different layers of accuracy:

- The highest level of accuracy is given by a so-called *perfect user*, that is that one wearing and owning all the sensors supported and that can provide all the kinds of raw data into the platform and that can interact with all its virtual interfaces (i.e. the possibility of using the given sensors in autonomy, the smartphone applications and the tablet games), performing the full process of medical visit with the clinical staff;

- The lower level of accuracy is given by the user that has a smartphone and can use a variable subset of the whole platform sensors and interfaces;

It needs to be specified that the shifting between the first and the second category is dependent on practical reasons – the degree of evolution of the frailty syndrome - much more than on any technical ones – related by the eventual unavailability of some parts of the system.

From an architectural point of view, this layering can be handled by the modular structure of the architecture itself, but these different types of accuracy must be considered in the modules that use participant's data as well.

This set of services will be implemented separately and will interact to each other, and will form a cloud platform where End Users will interact with mobile applications and web browsers, as shown in Figure 3.

It is worth noting that taking all these services in the same physical place is important to maintain all the hardware devoted to host them, and to make communications between modules more secure, affordable and low-latency. This kind of matters are deeply described in Sections 4 and 5.

3.1. Overall picture

In this part of the document, the architecture of the FrailSafe system will be described. This chapter will explore in more details the parts that compose the platform and each part will be detailed from an architectural point of view. Moreover, each part will describe the integration with the entire FrailSafe system, i.e. the modules to interact with and the input/output exchanges.

The description of the system architecture is organized according to its separation in the three layers, each one grouping a number of modular components - hardware or software or both- as shown in Figure 4.

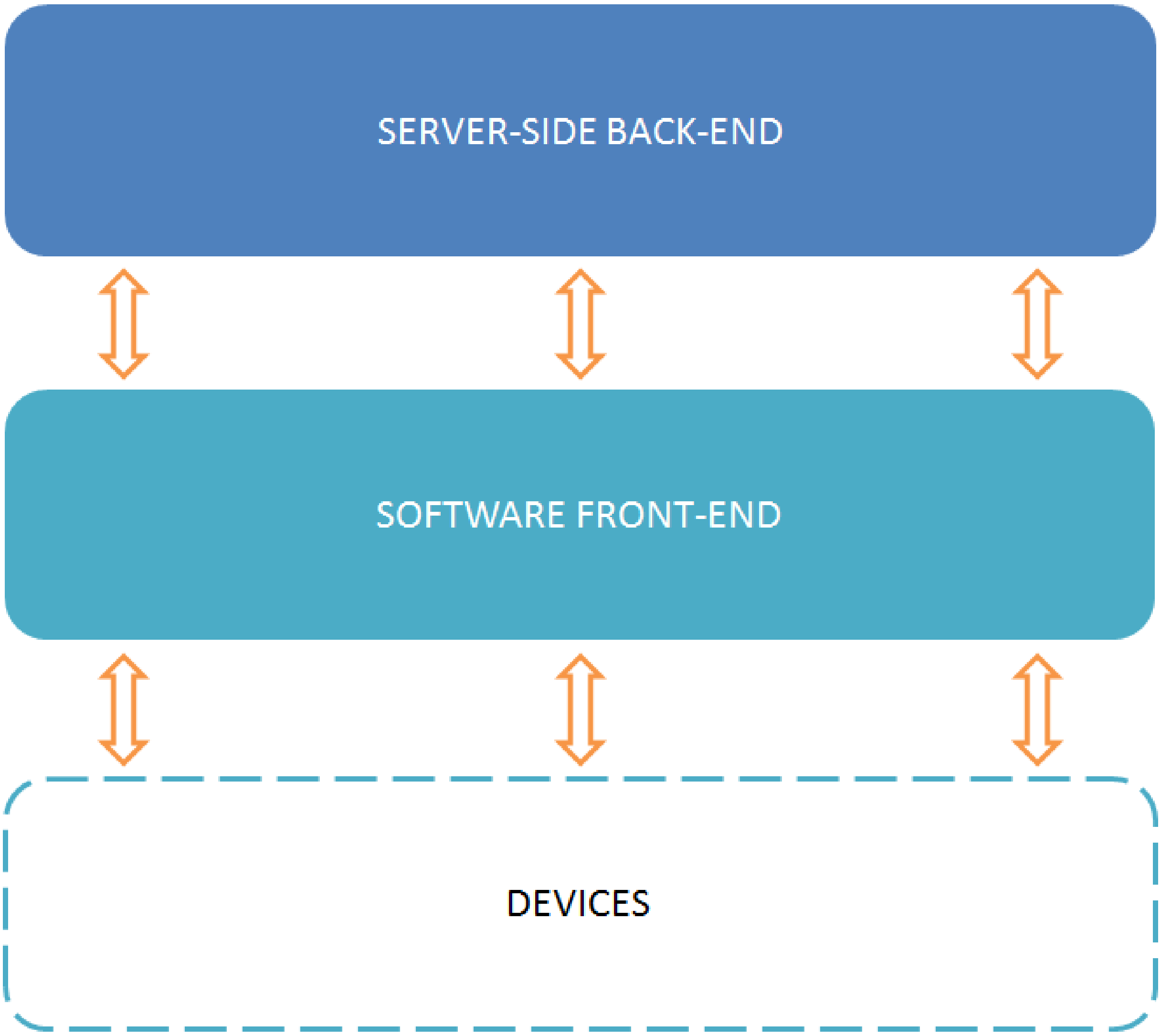


Figure 4 - Architecture's layering

As anticipated, the system can be considered as organised in three layers:

- the Devices, including interfaces and sensors, being part of the FrailSafe platform;
- the Software Front-end that is made available to the users (mobile and web applications) divided further between the ones for the participant and those for the clinical staff;
- the Server-side Back-end, wrapping up all the modules making available all the storing, processing, elaborating and analysing features core of the system;

Each layer is detailed in the following sections.

It is important to add that the conceptual effort of layering the architecture is made with the aim of making it vertically modular.

Towards the direction of increasing the modularity of the platform, an important step is made on its software side by making FrailSafe respect a Service Oriented Architecture design.

To quickly introduce this approach (see also Section 4), each software module of the system has to be seen as a discrete and atomic unit of features that can be deployed, consumed and updated in full independence. It is like a black-box for the rest of the system and its facilities can be accessed only through the set of services (e.g. APIs) it exposes.

Implementing the FrailSafe platform according to a Service Oriented Architecture design is expected to have very interesting benefits for the management of the integration processes of the project. Particularly:

- the delivery and deployment of each module can be performed individually;
- the testing phase is naturally organised in individual units;
- the maintainability of the whole platform should benefit from an integration based on a limited number of touching interfaces among "black-box" – i.e. each module's APIs – making easier to understand and isolate problem sources or identify blocks and bottlenecks.

It is not to forget that also the storage system is intended to be modularized having at least one independent persistence structure for each module of the system.

The final goal of this design is to move towards a micro-service architectural pattern. As a first step a less extreme practice has been decided to be adopted in order to stay flexible and adapt to each partner's internal practices and work organizations.

3.2. Devices

Most of the devices listed within this paragraph have been already fully described in other deliverables already produced for the first Project Review, done at M9. More specifically, D3.1 describes the Wearable WBAN System (WWBS), dynamometers, beacons, scales, blood pressure and arterial stiffness (Mobil-O-graph) monitors, plus the use of mobile devices (tablets and smartphones) as user input devices, while the selection of the mobile devices and of the augmented reality (AR) glasses have been described in Deliverable D5.1.

To avoid repetition, this paragraph will not describe those devices again, but it will depict their role in the context of the general architecture.

3.2.1. WWBS

The Wearable WBAN System has been fully described in Deliverable D3.1, Paragraph 2.1. As a short summary, is composed by:

- a sensorised vest/garment/strap (the final design will be decided after results will be obtained from its use in M15-24) with 2 fabric electrodes, a fabric piezoresistive sensor and 1-3 inertial monitoring units (IMUs), maybe doubled by another garment (with only IMUs on board) for lower limbs monitoring;
- an electronic device for data collection, pre-processing, storage and transmission;
- a software suite for data visualisation and download and for electronic device management.

Use cases and protocols of this system within the FrailSafe project are defined and detailed in the Deliverable D1.2. As a quick reference, it can be said that a WWBS, together with a kit of electronic devices, will be given to a *participant* every Monday and that it will be used autonomously for a monitoring period of 5 days in the first stages of the project. During this period the vest is worn and tested on the user. The electronic device can be coupled with capable devices for Bluetooth data transmission. A short data visualisation/recording, using the software tool provided together with the WWBS (see 3.3.2.1), should be performed to check proper system activity.

WWBS will be used in the frame of this project to save the following data from end-user:

- heart rate (HR);
- breathing rate (BR);
- data on posture, activity and movement.

All raw and processed data will be saved on a micro-SD present in the WWBS electronic device. During the phase of the medical research, when the WWBS is returned to the clinical centre at the end of the monitoring period, the electronic device will be plugged to a PC station and data downloaded to the PC through the provided software (see 3.3.2.1), then uploaded to the FrailSafe cloud through the Clinical Web Portal (see 3.3.2.3). After the end of the project, when all the processes and modules will be validated, the usage of a dedicated Docking Station (see 3.2.8) is foreseen, to automatically charging, downloading data from the WWBS and uploading to the FrailSafe cloud.

Part of the raw and/or the pre-processed data will be also transmitted via Bluetooth to the FrailSafe Gateway, according to the different Use Cases and scenarios identified in the Deliverable D1.2: as an example, data from IMUs might be used for gaming interaction with tablets.

3.2.2. IMUs

The Next Generation IMU2 (NGIMU) from ‘‘x-io Technologies’’ have been purchased (17 in total) after an amendment to DoA that has been requested to FrailSafe Project Officer: reasons to justify this change and selection of devices have been fully explained in D3.1, Annex I.

The NGIMU is a compact IMU and data acquisition platform that combines on-board sensors and data processing algorithms with a broad range of communication interfaces to create a versatile platform well suited to both real-time and data-logging applications.

On-board sensors include a triple-axis gyroscope, accelerometer, and magnetometer, as well as a barometric pressure sensor and humidity sensor. An on-board AHRS sensor fusion algorithm combines inertial and magnetic measurements to provide a drift-free measurement of orientation relative to the Earth. Each device is individually calibrated using robotic equipment to achieve the specified accuracy. Additional, external sensors, such as force or bend sensors, can be connected to the 8-channel analogue input interface.

Real-time communication is achieved via USB, Wi-Fi, or serial/RS-232. Data may also be logged to an on-board micro SD card. The NGIMU uses the popular OSC communication protocol and so is immediately compatible with many software applications and straight forward to integrate with custom applications with libraries available for most programming languages.

While WWBS will be used for long term monitoring of several physiological parameters, IMUs measure physical activity via monitoring the trunk as well as the upper and lower limbs for performing movement and posture analysis (i.e. classifications and parameterizations of user's movements). In order to simplify the design of the garments, the quality of the recordings and the comfort of the user, the system will be split in two or more components, a vest/T-shirt that will monitor trunk and upper limbs and a pair of trousers or other solutions for lower limbs monitoring.

[2] *http://x-io.co.uk/ngimu/*

3.2.3. Dynamometer

Dynamometers have been purchased after an amendment to the DoA of FrailSafe that has been requested to the Project Officer: the reasons to justify this change and selection of devices have been fully explained in Deliverable D3.1, Paragraph 2.2 and Annex II.

6 dynamometers per clinical partner have been delivered, so that 5 can be used by end-users at home and 1 remains at disposal of each centre.

These special hand dynamometers have been studied to monitor hand strength and it can be used for longitudinal studies (i.e. verifying the changes in hand strength in older adults in the long term) and for gaming, testing the ability to perform not only a prolonged effort but also to do that with proper timing and calibration.

All devices communicate via Bluetooth to the mobile devices, so that data can be:

- recorded and uploaded to the FrailSafe cloud;
- used to play games on tablets.

3.2.4. Beacons

The indoor localization module using beacons has been fully described in D3.1, Paragraph 2.6.

As a short summary, beacons are tiny, low power transmitters attached to walls or objects in the physical world. Each beacon has a Bluetooth Low Energy transmitter. It broadcasts tiny radio signals over the air containing unique, location-specific data. Modern smartphones constantly scan for these signals. If they enter their range, an associated monitoring app responds with the desired action (monitoring indoor activities, i.e. detecting how much time the participant spends in each room).

The Estimote™ beacons (Figure 5) have been selected for use in the FrailSafe project.

They have been selected primarily for their usability and ease of development and installation (see Deliverable D3.1, Paragraph 2.6.4).

Figure 5 - Estimote beacons

3.2.5. AR Glasses

The Augmented Reality (AR) glasses are devices meant to be worn by the participant as regular glasses, and able to project on the person’s field of view various virtual objects. They are used as interaction devices for AR games developed for the participants, involving both physical and cognitive effort, such as trying to move or catch a virtual object with the head or body movement.

The hardware requirements of the AR glasses to be used in FrailSafe have been detailed in deliverable D5.1. Due to safety reasons, AR glasses with optical see-through displays were considered as the most appropriate solution, since they allow a direct natural view of the world, with the virtual objects being displayed on top of it through a holographic optical element or similar technology.

The AR glasses selected for the purposes of the FrailSafe system are the Meta 2 glasses from Meta vision (see Figure 6). The glasses have a large field of view (90°) which allows for a natural view of the surroundings. They are connected to a PC via a cable, which is a disadvantage if they are meant for room-level movements. However, for safety reasons, the FrailSafe applications using the AR glasses are going to be limited to small movements with the participant standing or sitting, thus making the cabled connection an insignificant issue. The main characteristics of the Meta 2 glasses are the following:

- 90° field of view
- 2560x1440 pixels display
- 720p front-facing RGB camera
- Hand interactions
- Positional tracking
- 6-axis IMU
- Tethered connection to PC

Figure 6: The Meta 2 AR glasses.

3.2.6. Tablet

As mentioned in Deliverable D3.1, in FrailSafe ecosystem, the tablet represents a mobile device which the user employs to interact with the different applications of the system and to be connected to the network.

This device can be considered a perfect tool for its main characteristics (portability, connectivity, small weight, etc.) and in FrailSafe architecture it is mainly used by the participants for playing the Games and by the clinical staff during the trials to collect data from the participants using the Clinical Web Portal.

Analysing the minimum specifications, exposed in Deliverable D3.1, the choice of the model has been for the “Google Pixel C” (Figure 7).

Figure 7 - Google Pixel C

Thereafter, we see the technical specifications:

- Dimensions: 242 x 179 x 7 mm;
- Weight: 517 g;
- Display size: 10.2 inches;
- OS: Android OS, version 7.1.1 (codename “Nougat”);
- Connectivity: Wi-Fi 802.11 a/b/g/n, Bluetooth v4.1, GPS;
- Sensors: Accelerometer, gyroscope, proximity sensor, compass.

The use of mobile devices (tablets and smartphones) in the FrailSafe project has been fully described in D3.1, Paragraph 3, while their selection in Deliverable D5.1.

As a short summary, they will be used as:

- sensorial components;
- input devices;
- logging components;
- communication gateway; and
- data processor.

The tablet is one of the elements composing the so-called *FrailSafe Gateway* (see 3.3.1.7).

3.2.7. Smartphone

In the FrailSafe ecosystem, the smartphone has been exploited to its fullest potential for the great variety of sensor hardware integrated and for the possibility to transmit and receive data along the network hardware.

For the different employments, the choice has been for the “LG Nexus 5X” (Figure 8).

Figure 8 - LG Nexus 5X

Thereafter, we see the technical specifications:

- Network: GSM / CDMA / HSPA / LTE (Minimum GSM / CDMA);
- Display size: 5.2 inches;
- OS: Android OS, version 7.1.1 (codename “Nougat”);
- Connectivity: Wi-Fi 802.11 a/b/g/n/ac, Bluetooth v4.2, GPS, NFC;
- Sensors: Fingerprint, accelerometer, gyro, proximity, compass, barometer;

As described in Deliverable D3.1, the smartphone participates in different context in the communication scheme:

- Sensory component;
- Input device;
- Logging component;
- Communication component;
- Processing data.

The smartphone is one of the elements composing the so-called *FrailSafe Gateway* (see 3.3.1.7).

3.2.8. Docking Station

The use of this device is not foreseen during the collection of data within the timespan of the *FrailSafe* project, but potentially in the phase(s) following project conclusion, as a tool for the marketable product.

This device will be composed by a mini-PC, a touch-screen and a case with a slot/port for connection to the WWBS electronic device. This port will be used to charge the device battery and, at the same time, download recorded data using an automatic procedure. The mini-PC will be also programmed so that it is connected to an available WIFI network or cabled to a router, according to case, so that downloaded data can be made available to cloud services, including data storage on a remote server.

3.2.9. FORA Scale

The FORA scale has been described in some detail in Deliverable D3.1, Paragraph 2.3. It is meant to be used for measuring Body Weight, Body Fat and Body Mass Index of participants. In particular, during the process of visiting the older people, as required by the clinical assessments protocol, the collection of the values of the participants is performed through the usage of the FORA weightscale W310[3].

The protocol foresees that the clinical staff assists the participant in using the scale collecting the specified above data and that once the device shows the results of its evaluation these values are recorded into a remote database through a specific form provided by the Clinical Web Portal - i.e. the eCRF – (see 3.3.2.3). Alternatively, the data can be gathered thanks to the usage of the mobile application described in this document at section 3.3.2.2 as *Trial Center Auxiliary Devices*.

3.2.10. FORA BP Monitor

The FORA BP monitor has been described in Deliverable D3.1, Paragraph 2.4. It is meant to be used for measuring the blood pressure (arterial and systolic) of participants. This is done both during the process of visiting the older people, as required by the clinical assessments protocol; in addition a number of measurements can be performed in autonomy by the participant while at home. As described in Deliverable D3.1, the device for collecting the values of the participants is the FORA Active plus P30 Plus BT[4] blood pressure monitor.

The protocol foresees that the clinical staff assists the participant in using the BP monitor collecting the specified above data and that once the device shows the results of its evaluation these values are recorded into a remote database through a specific form provided by the Clinical Web Portal - i.e. the eCRF – (see 3.3.2.3).

In addition, the device is given to the participant to be used autonomously at home and the data produced can be gathered thanks to the usage of the mobile application described in this document at section 3.3.1.6 as *Home Auxiliary Devices*.

3.2.11. Mobil-O-Graph

The Mobil-O-Graph Agedio B900 has been described in Deliverable D3.1, Paragraph 2.5. It is meant to be used for measuring the arterial stiffness of participants. This is done during the process of assessing the participants at the trial centre, as required by the clinical assessments protocol. As described in the D3.1 document, the device for collecting the values of the participants is the Mobil-O-Graph Agedio B900[5].

The protocol foresees that the clinical staff takes measurements of the participant's arterial stiffness using the Mobil-O-Graph, collecting the specified above data and that, once the device shows the results of its evaluation, these values are recorded into a remote database through a specific form provided by the Clinical Web Portal - i.e. the eCRF – (see 3.3.2.3).

Alternatively, the data can be gathered thanks to the usage of the mobile application installed on the iPad associated to the Mobil-O-Graph as described in this document at section 3.3.2.2 as *Trial Center Auxiliary Devices*.

[3] http://www.foracare.com/weightscale-W310.html

[4] http://www.foracare.ch/Meter-P30plusBT.html

[5] http://www.iem.de/en/products/mobil-o-graph.html

3.3. Software Front-End

If we exclude the very basic actions performed directly on sensors and devices, the end-users mainly interact with the *FrailSafe* system through several software components and do it mainly through the mobile devices provided by the project (the smartphone and the tablet).

These software front-ends provide some very important features of the system. In particular:

- They make possible the retrieval of the data collected through the devices and their facilities;
- They are responsible for the exchange of the data - produced by all the users' interactions – with the server-side back-ends;
- They provide the User Interfaces to the platform services (mobile and web applications);

It is also worthy to add that these features are customized to the physical interface they are executed on.

As depicted in Figure 9, the front-end components are separated logically into two groups, according to the category of end-users they refer to. The first group is of those used by the *participant* and the second is of those used by the *clinical staff*.

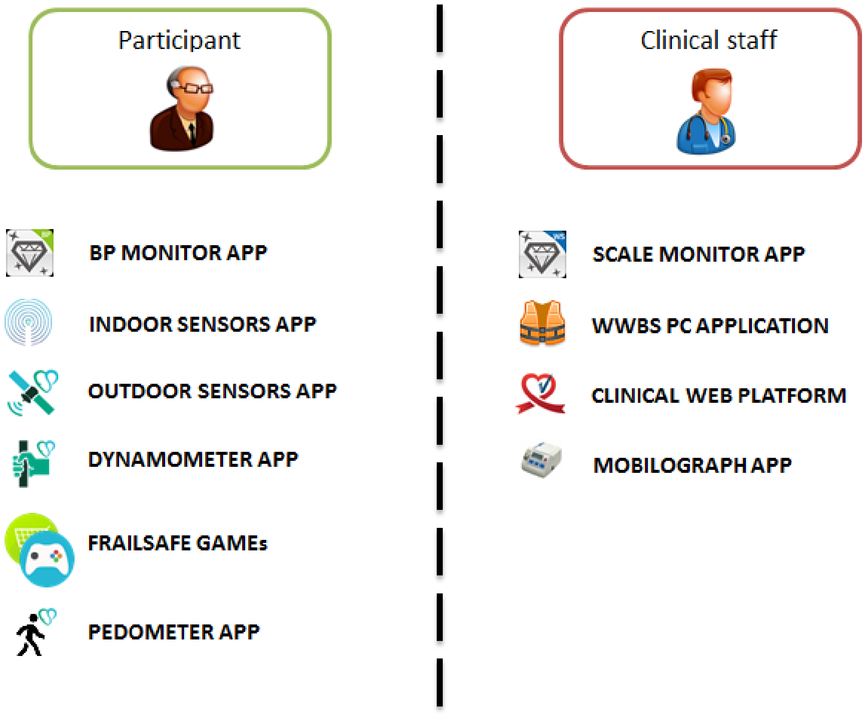


Figure 9 - Scheme of grouped front-ends

Considered at high-level, the previously described features provided through the usage of the front-end, it is important to shortly detail here how they are meant to work.

As depicted in Figure 10, it is possible to see that starting from the data collection and including the data exchange feature, a number of sensors (see 3.2) is used to produce and collect data about the *participant* end-user. All the data are retrieved from these sensors by what we call the *FrailSafe* Gateway (see 3.3.1.7) in several different ways, stored locally and finally sent to a remote server. An internet connection must be available to make this last step possible.

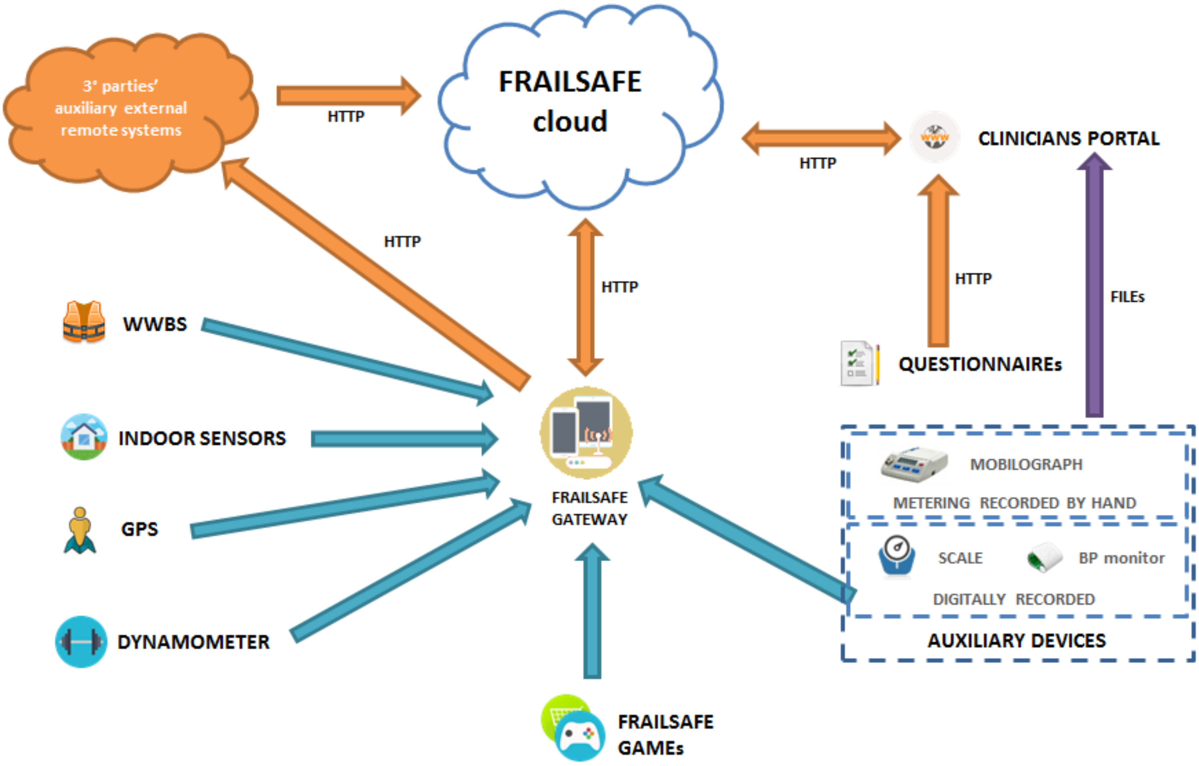


Figure 10 - Overall front-end data flows

The other important feature made available by the front-end, that is worthy to quickly introduce, is the fact of being the portable user interface to the platform services for all the categories of users.

A short explanation of how this is organised follows below:

- for the *participant,* the main interactions are those with the sensors used autonomously – e.g. the WWBS or the BP monitor – and with some mobile apps they need to use for both performing the recording of sensors data, for playing with the games on the tablet and both for receiving eventual messages coming from the platform itself (generated locally or remotely by the different system modules eventually in charge of it);

- for the *clinical staff,* the main interactions are in some specific areas:
    - with the sensors - and the related mobile apps - for controlling the devices, performing the measurements of the participants' parameters and recording those data;
    - with the web applications used for performing the participants' visits (filling up forms or questionnaires and uploading the data exported from sensors);
    - with the web applications used for configuring the games;

The single components of the software front-end group are detailed in the following sections.

3.3.1. Participant front-end

As already mentioned in the previous paragraphs, it is foreseen that the *participant* has a limited number of interactions available with the platform. The main reason leading to that is to increase the usability of these interfaces and of the platform in more general, trying to reduce the cognitive overloading caused by the necessity of learning and remembering how to use the platform.

In details, the participant is meant to interact with:

- The sensorized garment, i.e. the WWBS, not needing any participant software front-end to be used for performing the data collection (no actual interaction needed);

- The Indoor and Outdoor monitoring apps, needing just to be installed and setup on the participant’s smartphone to run in background (no actual interaction needed);

- The mobile app collecting data from the dynamometer. The participant can eventually use in autonomy the dynamometer – given by the clinical staff – for performing measurements;

- The *FrailSafe* Games, running as app on the tablet given to the participant, can be used in autonomy – eventually in combination with some of the other sensors available;

- The blood pressure monitoring device that can be used autonomously to perform measurements: a specific mobile app is needed to perform these measurements, save them and sent these recordings to the remote infrastructure in charge of using them;

- Any message coming from the platform – both from local and remote processing – communicated to the participant via a specific app, running on the smartphone.

In the following pages a detailed presentation of each one of the previously listed software components can be found.

3.3.1.1. Indoor Sensors (indoor monitoring) Android app

Description

The indoor monitoring module is responsible for the monitoring of the indoor position of the participant, within his/her home environment, and for the interaction with the FrailSafe gateway. The indoor monitoring module receives signals from indoor sensors, i.e. Bluetooth Low Energy (BLE) beacons (see 3.2.4), processes them, in order to extract location information, and transmits the collected data to the mobile gateway.

The indoor monitoring module is implemented as a mobile application (see Figure 11). The clinical staff can configure the module’s parameters through a mobile graphical user interface.

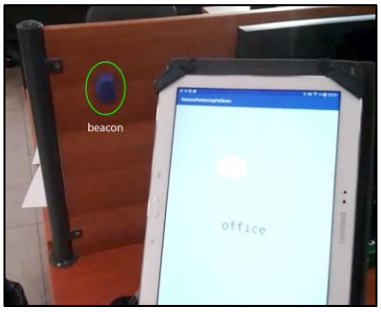


Figure 11 - Indoor Monitoring App

Features

The main functionalities of the indoor monitoring module can be summarized in the following:

- Collection of distance information from BLE beacons positioned in the participant's premise;

- Computation of the room ID in which the person is at any time;

- Computation of the x and y coordinates of the person within the premise, in real time;

- Transmission of the collected data (room ID, coordinates, timestamp) to the FrailSafe gateway;

- Configuration of the indoor monitoring parameters through a graphical user interface. These parameters include:
    - The room ID assigned to each beacon;
    - The coordinates of the beacons within the facility;
    - The frequency of distance measurements considered;

- Training of the indoor monitoring module in a specific facility, using supervised information supplied by the clinical personnel, through the graphical user interface, in case the room IDs assigned to the beacons or the exact beacon coordinates are not available.

Data I/Os

In this section the dependencies with respect to other modules – in matter of I/O data flows – are described.

Figure 12 depicts an overview of the interactions between the indoor monitoring module and other FrailSafe components. The specific inputs and outputs of the indoor monitoring module are presented in Table 1 and Table 2, respectively.

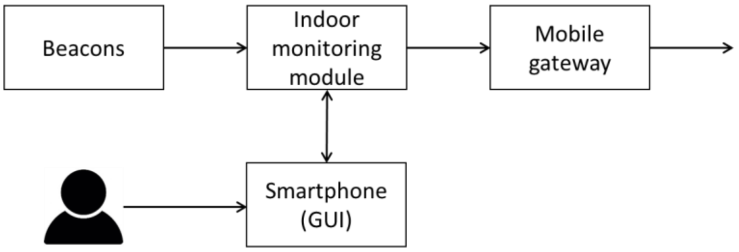


Figure 12: Communication of the indoor monitoring module with other FrailSafe components.

Input name	Description	Source component
Beacon signal strengths	*The signal strength of each beacon positioned in the facility, as measured by the mobile device hosting the indoor monitoring module.*	BLE beacons
Configuration parameters	*The configuration parameters for the monitoring procedure, including the room ID assigned to each beacon, the coordinates of the beacons within the facility and the desired measurement frequency.*	Indoor monitoring GUI
Training data	*Supervision data provided by an expert (e.g. clinical personnel), such as the room ID or exact coordinates assigned manually when the hosting device is at a specific location. These data are used to allow the module to learn the target facility space, without having explicit knowledge of the facility area and the exact beacon locations.*	Indoor monitoring GUI

Table 1: Input parameters of the indoor monitoring module.

Output name	Description	Destination component
Room ID	*The ID of the room in which the participant is at a specific time, as computed by the indoor monitoring module.*	FrailSafe gateway
x and y coordinates	*The x and y coordinates at which the participant is at a specific time, as computed by the indoor monitoring module.*	FrailSafe gateway
Measurement timestamp	*The timestamp of the specific time instance at which the above room ID and coordinate measurements have been computed.*	FrailSafe gateway
Configuration parameters	*The current values of the configuration parameters for the monitoring procedure, including the room ID assigned to each beacon, the coordinates of the beacons within the facility and the measurement frequency.*	Indoor monitoring GUI

Table 2: Output parameters of the indoor monitoring module.

3.3.1.2. GPS (outdoor monitoring) Android app

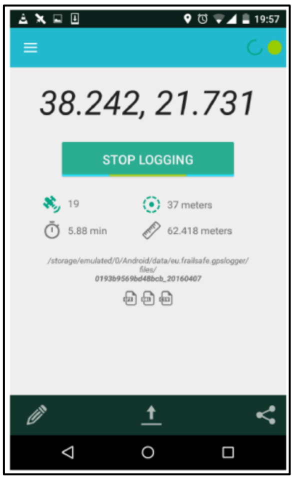
Description

FrailSafe will use information capture, analysis and modelling to make an overall assessment of physical activity patterns through embedded smart phone sensors and tracking systems.

The outdoor monitoring module is responsible for (a) the gathering of the tracked position information of the participant using it (namely, latitude, longitude and other location-specific measurements) and (b) the transmission of the collected information to the FrailSafe *Gateway*.

The mobile phone acts as a Global Positioning System (GPS) receiver, making it capable of obtaining information from GPS satellites and then to accurately calculate its geographical location.

The users can configure the module's parameters through a graphical user interface that is implemented as a mobile application (GPS Tracker App). The main screen of this application is illustrated in the figure below. More specifically, the application starts logging by clicking the main "Start Logging" button and continues to log, even when the application is closed, until the button is clicked again ("Stop Logging"). It preserves a notification open, informing the user that it is still logging.



38.242, 21.731

19

37 meters

5.88 min

62.418 meters

/storage/emulated/0/Android/data/eu.frailsafe.gpslogger/files/
0193b9569bd48bcb_20160407

Figure 13 - Outdoor Monitoring App

Features

The main functionalities of this module can be summarized in the following key components:

- Logs day's session to a file with the current date in ''yyyymmdd.<type>'' format;

- Provides activity identification
  - Walking
  - Running
  - Driving

- Auto-stores (locally to the mobile phone) and auto-sends (through the Mobile Gateway to the FrailSafe Cloud Server) log sessions in specified time intervals;

- Provides configurable settings for the FrailSafe GPS Tracker application regarding:
  - General
    - Enable/Disable GPS;
    - Enable/Disable logging session when application starts;
    - Show/hide notification buttons ("Annotate" and "Stop").

  - Logging
    - Change log file type:

        - GPS Exchange format (.GPX): Most common format used to show locations and routes (Default selection);
        - Keyhole Markup Language (.KML): Used by Google Earth[6];
        - Comma-separated values (.CSV): Plain text-file logging;
    - Change local folder: Where to save the log files locally on the phone.

- Performance
    - Location providers: Choose to receive location data from GPS, Network and Passive (All 3 are selected by default);
    - Logging time interval: Frequency of position logging in seconds (Default is 30);
    - Keep GPS between fixes: Whether to keep GPS open after acquiring a position. Results in higher accuracy but lower battery life;
    - Distance & Accuracy filter: Minimum distance travelled and position accuracy required to log a location;
    - Retry and Absolute timeouts: Number of seconds to retry getting a (better) position before giving up;
    - Activity Recognition: Whether to keep logging when the user is detected to be still.

- Uploading
    - Auto-sending: Allow sending the files automatically;
    - Auto-sending frequency: How often to send files (Default: 60 minutes);
    - Send when Stop Logging is pressed;
    - Send .zip if applicable;
    - Send over Wi-Fi only.

Data I/Os

In this section the dependencies with respect to other modules – in matter of I/O data flows – are described.

The Outdoor monitoring module is one-way connected to the FrailSafe Gateway by providing it the tracked logs data for each participants (see Figure 14). The FrailSafe Gateway is then responsible for streaming the received information to the FrailSafe Cloud server for further processing and analysis. The specific inputs and outputs of the outdoor monitoring module are presented in Table 3 and Table 4.

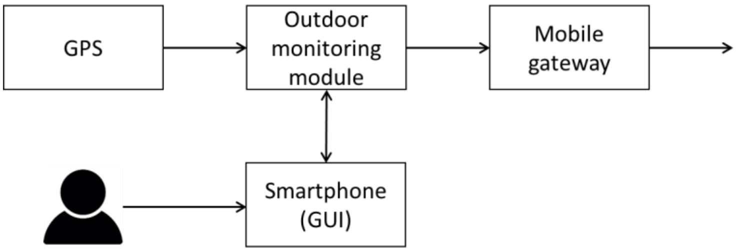


Figure 14. Communication of the outdoor monitoring module with other FrailSafe components.

[6] https://www.google.com/earth/

Input name	Description	Source component
GPS signal measurements	*The GPS signal location-specific measurements as gathered by the mobile device hosting the outdoor monitoring module.*	GPS Receiver
Configuration parameters	*The configuration parameters for the outdooring monitoring procedure, including logging, performance and uploading settings.*	Smartphone GUI

Table 3: Input parameters of the outdoor monitoring module.

Output name	Description	Destination components
Location-specific measurements **(.GPX,.KML,.CSV)**	*The current values of the ongoing user tracking procedure to be visualized by the mobile application and be sent to the* FrailSafe *gateway that includes the measurement of the latitude and longitude GPS coordinates, distance & duration travelled and overall position accuracy.*	Smartphone GUI FrailSafe gateway

Table 4: Output parameters of the outdoor monitoring module.

3.3.1.3. Pedometer Android App

Description

In order to measure indoor and outdoor step movement, FrailSafe will provide an efficient tracker of participant daily movement. The idea of this application is simple; it records the number of steps one has walked for any activity (walking, running, stairs-climbing) by using the low-power movement sensors included in modern smartphones. The Pedometer module is responsible for (a) the gathering of the tracked step activity information of the participant and (b) the transmission of the collected information to the FrailSafe Gateway.

Features

The main functionalities of this module can be summarized in the following key components:

- Number of times a walking activity is initiated;
- Automatic steps counter & pace (steps per minute) detection;
- Automatic tracking of stairs climbed.

The Pedometer module is one-way connected to the FrailSafe Gateway Platform by providing it the tracked mobility log data for each participant (see Figure 15). The Mobile Gateway Platform is then responsible for streaming the received information to the FrailSafe Cloud Server for further processing and analysis. The specific inputs and outputs of the step monitoring module are presented in Table 5 and Table 6, respectively.

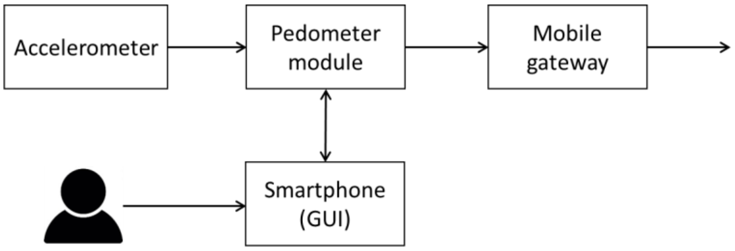


Figure 15. Communication of the pedomenter monitoring module with other FrailSafe components.

Input name	Description	Source component
Accelerometer measurements	*The accelerometer signal measurements as gathered by the mobile device hosting the step monitoring module.*	Smartphone Accelerometer
Configuration parameters	*The configuration parameters for the step monitoring procedure, including logging, performance and uploading settings.*	Smartphone GUI

Table 5. Input parameters of the step monitoring module.

Output name	Description	Destination components
Step-specific measurements	*The current values of the ongoing user tracking procedure to be visualized by the mobile application and be sent to the* FrailSafe *Gateway, that includes the measurement of step activity such as the walking initiations/ duration, steps accumulation and stairs climbed.*	Smartphone GUI FrailSafe Gateway

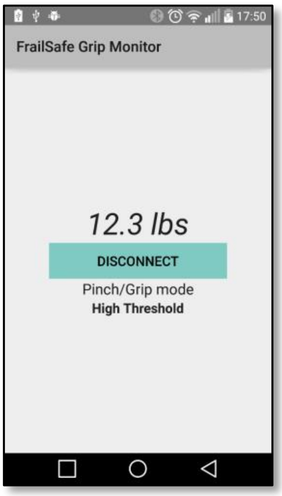
Table 6. Output parameters of the step monitoring module.

3.3.1.4. Dynamometer Android app

Description

FrailSafe will explore digital dynamometer hardware technology to execute force evaluation and testing (FET) as well as to measure overall hand grip strength of the participant allowing the clinician to document, reinforce, and support their treatment plan. The *Hoggan MicroFET* Grip digital handgrip has been chosen (see section 3.2.3) for routine screening of grip and pinch strength assessments.

The strength evaluation module is responsible for (a) the measurement of the handgrip strength forces evaluating consistency of effort in terms of peak force and elapsed time and (b) the transmission of the collected information to the FrailSafe Gateway and Games Platform Frontend. The users can configure the module’s parameters through a graphical user interface that is implemented as a mobile application (Grip Monitor App). The main screen of this application is illustrated at the figure below.



FrailSafe Grip Monitor

12.3 lbs

Pinch/Grip mode
High Threshold

Figure 16 - Grip Monitoring App

Features

The main functionalities of this module can be summarized in the following key components:

- Offline transmission of logged strength force and time measurements for daily screening (through the Mobile Gateway to the FrailSafe Cloud Server);

- Real-time streaming of hand strength and time assessments as guidance tools in game-oriented intervention strategies;

- Provision of configurable settings for the FrailSafe Grip Monitor application regarding:
  - General
    - Connect/Disconnect with a paired dynamometer device;
    - Enable/Disable logging session when application starts;
  - Logging
    - Change strength units of measure type:
      - Pounds (lbs.), Newtons (N), or Kilogram-force (kgf);
    - Change local folder: Where to save the log files locally on the phone;
  - Real-time Display
    - Current mode of operation
      - Pinch or Grip;
    - Threshold setting
      - Low (0.8 lb. to 200 lbs. in 0.1 lb. increments);
      - High (3 lbs. to 200 lbs. in 0.1 lb. increments);
    - Current force measurement;
  - Performance
    - Activity Recognition
      - Whether to keep logging when the user is detected to not using the device;
  - Uploading
    - Auto-sending: Allow sending the files automatically;
    - Auto-sending frequency: How often to send files (Default: 60 minutes);
    - Send when Stop Logging is pressed.

Data I/Os

In this section the dependencies with respect to other modules – in matter of I/O data flows – are described. The Strength evaluation module is one-way connected to the FrailSafe Gateway and Games Platform by offline & real-time streaming the tracked strength data (received from the dynamometer device via Bluetooth technology) for each participant (see Figure 17). The FrailSafe Gateway is then responsible for streaming the received information to the FrailSafe Cloud Server for further processing and analysis. On the other hand, digital grip strength stream data will also be exploited as a navigation controller from the developed FrailSafe games. The specific inputs and outputs of the strength evaluation module are presented in Table 7 and Table 8, respectively.

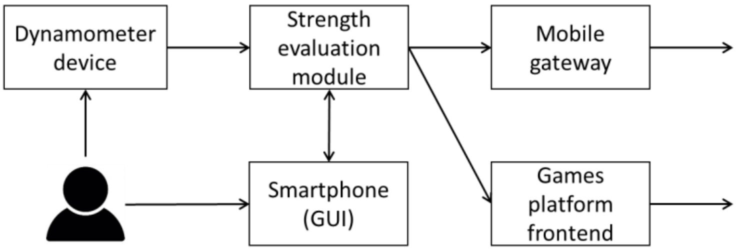


Figure 17. Communication of the strength evaluation module with other FrailSafe components.

Input name	Description	Source component
Strength evaluation measurements	*The strength force measurements as gathered by the handgrip dynamometer hardware via Bluetooth protocol.*	*Dynamometer device*
Configuration parameters	*The configuration parameters for the strength evaluation procedure, including logging and performance settings as well as evaluation options.*	*Smartphone GUI*

Table 7: Input parameters of the strength evaluation module.

Output name	Description	Destination components
Offline strength evaluation measurements	*The aggregated values of the time-specified (daily) strength tracking procedure (force and elapsed time evaluations) to be sent to the* FrailSafe Gateway *for clinical screening and data analysis.*	FrailSafe Gateway
Real-time strength evaluation measurements	*The current force and elapsed time measurement values of the ongoing strength evaluation process to be visualized by the mobile application as well as to be streamed at the games platforms front-end.*	Smartphone GUI Games platform

Table 8: Output parameters of the strength evaluation module.

3.3.1.5. FrailSafe GAMEs

Description

The games component of FrailSafe is a rather large component, consisting of a variety of games having two main purposes:

- monitoring the physiological and cognitive parameters of the participants;
- providing rehabilitation exercises.

The games require the *participant* to perform some tasks using FrailSafe equipment, e.g. tablet, Augmented Reality (AR) glasses, dynamometer, Inertial Measurement Unit (IMU) sensors on a vest, etc.

During the game play, the user performance is monitored by the game application. The collected data are transmitted to the game's back-end module, which is responsible for recording them.

Features

The individual games will have their own purpose (train cognitive, physical or a combination of both functions) and gameplay characteristics (2D or 3D scenarios), but there will be some common features shared among all games:

- The purpose of the game will be to perform a cognitive and/or physical activity employing the provided devices, e.g. memorizing objects or applying physical strength to move items on the screen;

- The graphical interface of a game could be displayed on multiple display devices, including the following:
  - tablet screen;
  - AR glasses;

- The games will be mainly designed to be played on a tablet device, but there will be versions available for using the games on computer platforms;

- There will be games designed specifically to be played with the AR glasses;

- A number of input devices could be used to register the participant's activity. These include:
  - the tablet touch screen;
  - the head tracking system of the AR glasses;
  - IMU sensors placed on a chest and arms vest;
  - Dynamometers;
  - GPS sensors;
  - indoor localization sensors (beacons);
  - Heart rate sensor;
  - Breath rate sensor.

- The display devices will be combined with the sensory hardware above (see table below):

Sensor / Display	Mouse & key-board	Touch screen	GPS	Head tracking	Dynamo-meter	Vest IMU sensors	Beacons	Heart & breath rate sensors
PC	🟩	🟥	🟥	🟥	🟩	🟩	🟨	🟨
Tablet	🟥	🟩	🟨	🟥	🟩	🟩	🟨	🟨
AR glasses	🟥	🟥	🟥	🟩	🟩	🟩	🟨	🟨

- To prevent FrailSafe games from causing accidents and falls, the games will not use the GPS sensors, the heart rate sensor, the breath rate sensor and the beacons to interact with the participant. These sensors will exclusively be used to monitor the participants while they play or do other daily tasks;

- The input from the devices and sensors will pass through a synchronization module, in order to filter and align the relevant data;

- There will also be a collection of games specifically designed for Augmented Reality as the screen resolution is different, the design is more complex and the user is not able to interact with the screen;

- The games will be designed to use one display device at a time;

- The games will be implemented with appropriate graphical design, 2D or 3D, based on the needs and objectives of the game. The 3D games will also be implemented with different game control perspectives, First Person View (FPV) or top-down. Every participant playing a game is assigned a user account, which is linked to the data collected by the game;

- During a playing session, the game application will record the user behaviour, performance, score and results for later processing and analysis done by the offline Analysis (see section 3.3.3.3). The user device will automatically upload the data recorded during the session to the FrailSafe cloud;

- The games will have multiple difficulty levels, e.g. easy, advanced and pro, independently configurable for each cognitive/physical function addressed in a game;

- The participant will be able to view the list of assigned exercises and difficulty levels and to open the game that he/she decides to play within a specific session;

- The functional domains that will be handled by the FrailSafe games are:
    - Physiological domains:
        - Resistance;
        - Balance;
        - Endurance;
        - Muscular strength;
        - Coordination;
    - Cognitive domains:
        - Long term Memory (episodic memory);
        - Short term Memory (working memory);

- Orientation;
- Verbal fluency;
- Executive Function;
- Visual Spatial ability;
- Processing speed;
- Attention.

The FrailSafe games features are distributed between the two separate levels in which their internal structure is layered into and that can be considered as sub-modules:

- The first, described here as *Front-End part*, devoted to manage all the interactions with the users;

- The second, called here *Back-End part*, dedicated to manage the local data management and all the communications with the server-side APIs provided by the FrailSafe Cloud. In particular, the back-end of the FrailSafe games system will handle the game-related information of user accounts registered on the FrailSafe cloud. This module is responsible for sending the recorded user performance data to the FrailSafe cloud.

Data I/Os of Front-End part

Figure 18 presents an overview of the communication between the game module and the other FrailSafe components, while Table 9 and Table 10 present the specific inputs and outputs of the game module.

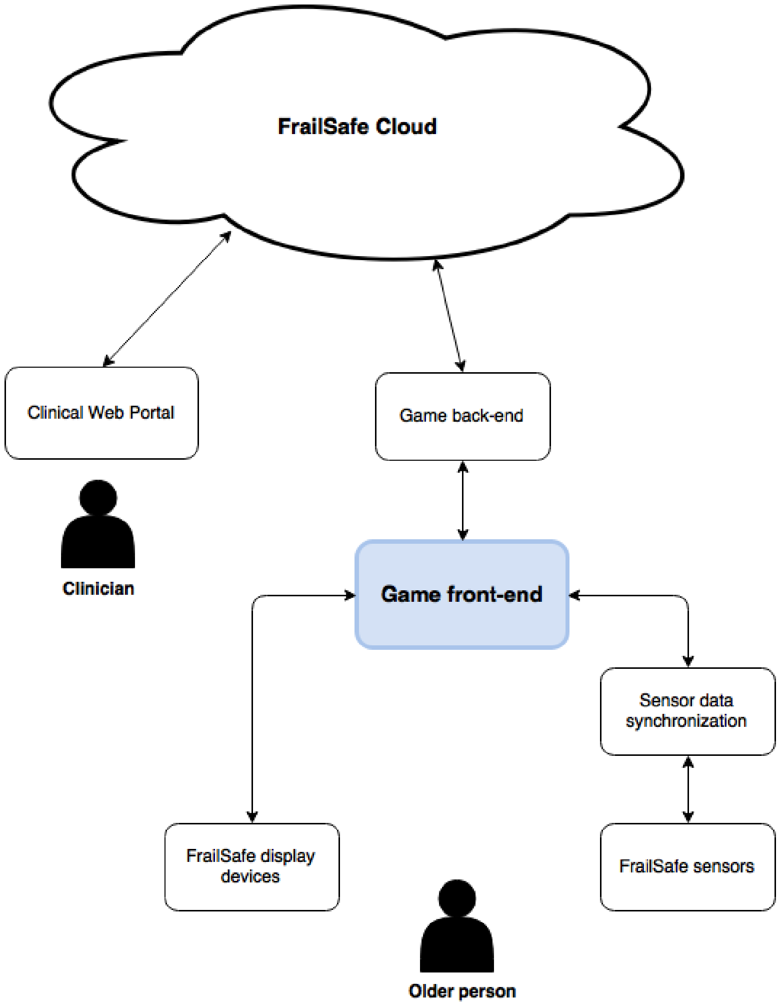


Figure 18: Communication of the game front-end module with other FrailSafe components.

Input name	Description	Source component
User interaction	*Data from the* FrailSafe *display devices and sensors involved in the game. The data are synchronized after passing through a data synchronization module.*	FrailSafe display devices and sensors
Game application configuration	*Parameters that configure the game application e.g. sensor peer configuration, user credentials, etc.*	FrailSafe display devices
User personal data	*User information and history used to personalize the front-end for the specific user, e.g. completed games in previous sessions, score, etc.*	FrailSafe cloud
Game schedule and difficulty level	*Data of exercises, game schedule and difficulty levels that have been assigned to the participant by the clinical staff.*	FrailSafe cloud

Table 9: Input parameters of the game front-end module.

Output name	Description	Destination component
User performance	*Summary of the user performance in the game, along with every event timestamps.*	FrailSafe cloud
Raw data of the session	*Full record of the session, i.e. every action performed by the user for each game played in the active session.*	FrailSafe cloud
User credentials	*User login and password of the participant requesting access to the game application.*	FrailSafe cloud

Table 10: Output parameters of the game front-end module.

Data I/Os of Back-End part

As described in the Features section of this Chapter, the games back-end module will upload the user, game and session information to the FrailSafe cloud if the connection to the cloud is established. The back-end shall store the data locally until they can be sent.

This module will also download the user information that is relevant for the active session, e.g. list of games that have to be played, assigned schedule and difficulty of those games and current status of assigned games.

Figure 19 depicts the communication of the games back-end module with the other FrailSafe components. Table 11 and Table 12 summarize the inputs and outputs, respectively, of the game back-end module.

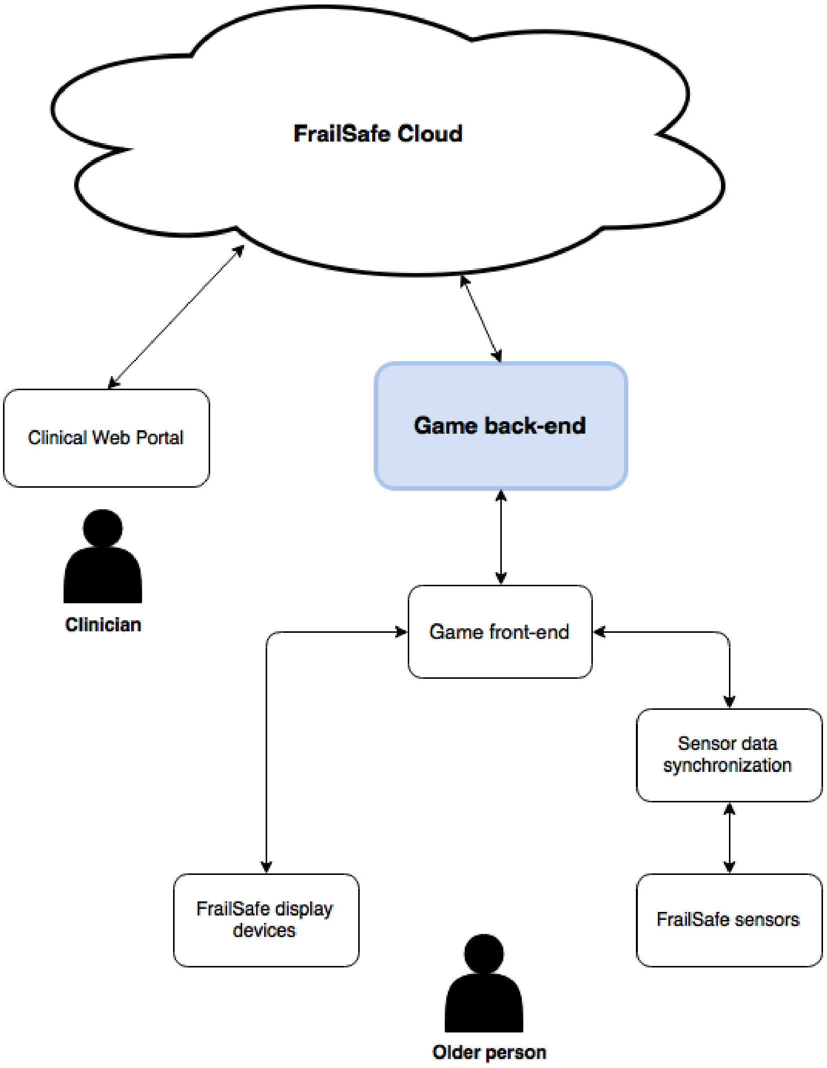


Figure 19: Communication of the game back-end module with other FrailSafe components.

Input name	Description	Source component
User personal data	*User information and history used to personalize the front-end for the specific user, e.g. completed games in previous sessions, score, etc.*	FrailSafe cloud
Game schedule and difficulty level	*Data of exercises, game schedule and difficulty levels that have been assigned to the participant user by a clinician.*	FrailSafe cloud

Table 11: Input parameters of the game back-end module.

Output name	Description	Destination component
User performance	*Summary of the user performance in the game, along with every event timestamps.*	FrailSafe cloud
Raw data of the session	*Full record of the session, i.e. every action performed by the user for each game played in the active session.*	FrailSafe cloud
User credentials	*User login and password of the participant requesting access to the game application.*	FrailSafe cloud

Table 12: Output parameters of the game back-end module.

3.3.1.6. Home Auxiliary Devices

Description

The Home Auxiliary devices module is responsible for the collection of data from those third party auxiliary devices used in FrailSafe that can be used autonomously by the participant while at home - i.e. the FORA blood pressure (BP) monitor[7].

This module can be considered functionally and operatively uniform to the Trial Center Auxiliary Devices module as long as the work and data flow from data collection to remote data storage are almost the same.

For the BP device, there is a dedicated mobile application available from FORA, which is used for monitoring the measurements (see Figure 20).

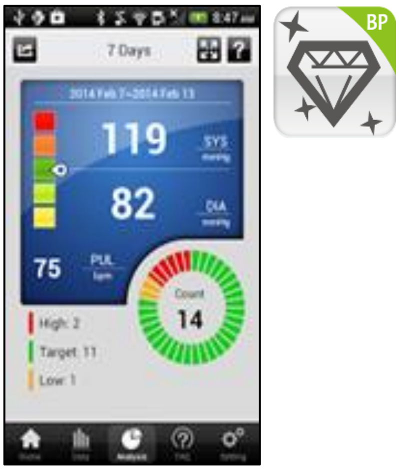


Figure 20 - Fora App

Features

The main functionalities of the auxiliary devices module are the following:

- Collecting the measurements from the corresponding devices;
- Displaying the measurements to the user. Apart from the display embedded in the devices themselves, the devices are wirelessly connected to dedicated mobile

[7] http://www.foracare.ch/Meter-P30plusBT.html

applications, which can visualize the measurements using more sophisticated means (charts, plots, etc.), as well as present statistical information from previous usages;

- Third party synchronization. For the BP monitor, the data collected on the app are synchronized to the FORA cloud system automatically through the dedicated mobile app. Later on relying on the FORA TeleHealth Web API, the data are collected and imported into the FRAILSAFE system and collected there;
- Alternatively, the measurements can be downloaded as CSV file from the FORA TeleHealth web portal and uploaded manually into the FrailSafe platform through a specific form on the Clinical Web Portal.

Data I/Os

In this section the dependencies with respect to other modules – in matter of I/O data flows – are described. The dependencies of the auxiliary devices module to the other FrailSafe components are illustrated in Figure 21. The auxiliary devices module accepts input from the auxiliary devices themselves and transmits output to the FORA cloud and finally to the FrailSafe cloud. Table 13 and Table 14 describe the specific inputs and outputs, respectively, of the auxiliary devices module.

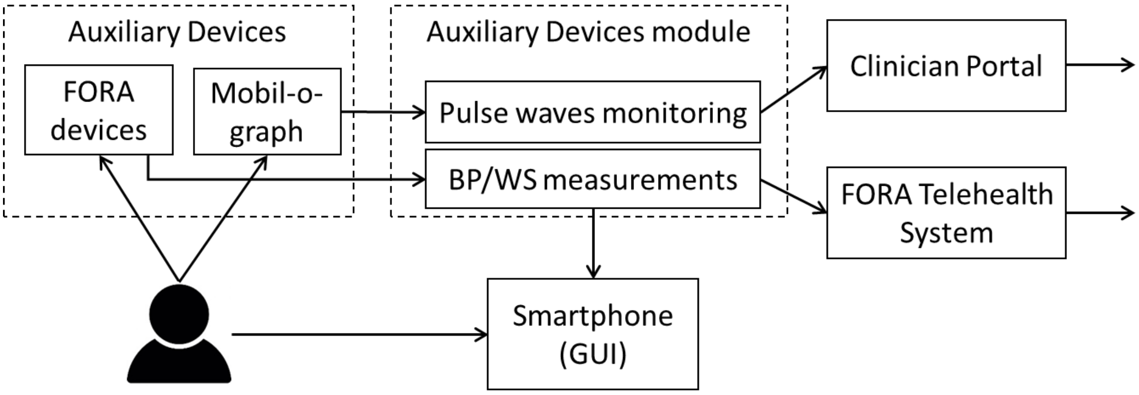


Figure 21: Communication of the auxiliary devices module with other FrailSafe components.

Input name	Description	Source component
Measurement data	*The measurement data, collected from the auxiliary devices. The contents and the format of the data are specific to each auxiliary device.*	Auxiliary devices

Table 13: Input parameters of the auxiliary devices module.

Output name	Description	Destination components
Measurement data	*The measurement data, represented to a format appropriate for the corresponding internal (clinician portal) and external (FORA) streaming portal.*	Clinician Portal FORA Telehealth System

Table 14: Output parameters of the auxiliary devices module.

3.3.1.7. FrailSafe Gateway

Description

The so-called FrailSafe *Gateway* is a logical component of the system. It is presented as a unique element and this is useful for giving a coherent presentation of the system architecture having a unified vision of the set of basic and horizontal features it provides. Having said that, it is important to underline that even though it is presented as one element, its features are shared in practice among three separated elements of the system.

These elements are three of the devices made available by the system: the *Smartphone*, the *Tablet* and the *Docking-Station*. All the functionalities implemented by this FrailSafe *Gateway* (Figure 22) then are distributed and in different amounts provided by each one of these actors.

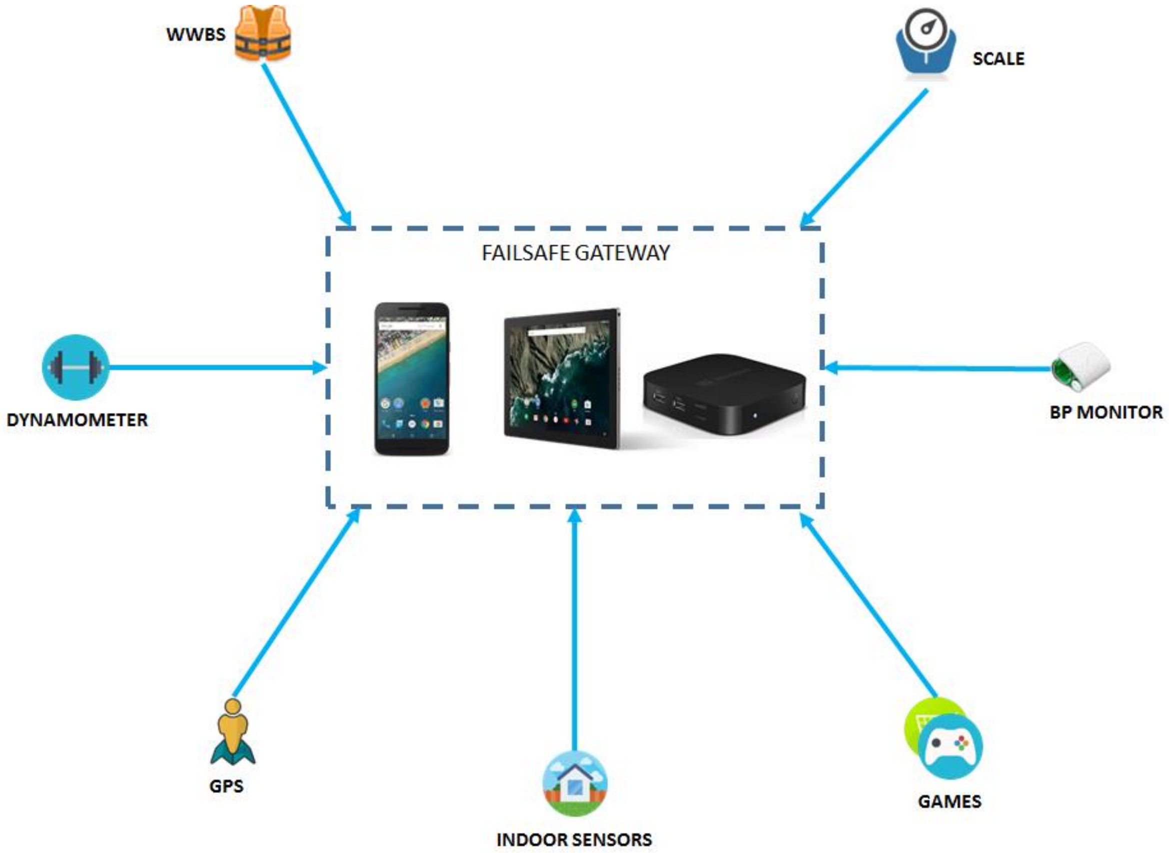


Figure 22 - the FrailSafe "gateway"

It is worthy to quickly depict how each one of these units is contributing to be part of the FrailSafe *Gateway*.

The first of these components to be taken in consideration is the Smartphone. It mainly collects data and sends them and for doing so it relies on mobile apps.

They are described in more details in this document at the *Front-End Chapters* but they can be summarized as follows:

- App for Indoor Monitoring
- App for Outdoor Monitoring
- App for Dynamometer
- Apps for Home Auxiliary Devices
- App for step counting

All these applications deal with their proper cloud endpoints separately and each one of them is expected to exchange the required data with the remote servers independently – in any case in accordance with the identity and security protocol established by FrailSafe.

The second element being part of the Gateway is the Tablet. It mainly collects data through the usage of the provided FrailSafe *Games*.

The third actor that is composing the FrailSafe *Gateway* is the so-called Docking Station, that is a mini-PC based device, used at the participant's home mostly with the aim of guaranteeing the secure transferring of the WWBS data while it is charging its internal battery. The data transferring from the WWBS to the Docking-Station is performed via USB cable connection and the sending is performed via WIFI connection. Everything is done automatically.

Features

As depicted in Figure 23, the FrailSafe Gateway has two main objectives:

- Collecting data from all the data sources (sensors, apps, etc.);
- Sending this data to the remote cloud systems.

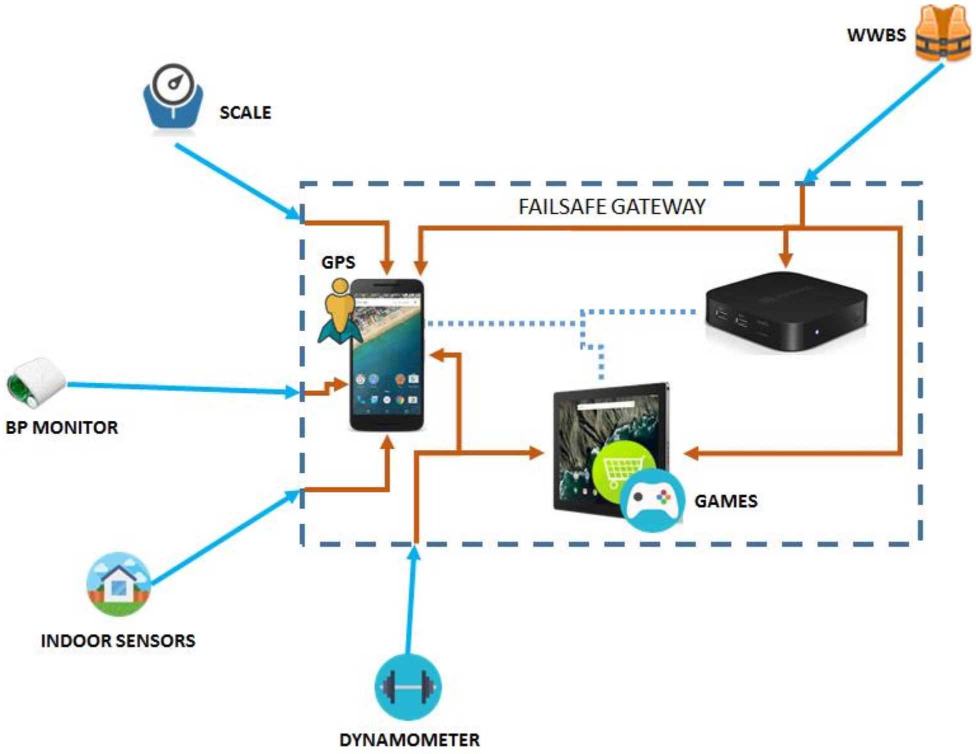


Figure 23 - FrailSafe Gateway data inputs

As represented in Figure 23, this *Gateway* gathers data from a set of data sources directly interacting with them in several means (i.e., Bluetooth, internal APIs, third party's app, etc.). They are:

- The WWBS;
- The Indoor Monitoring System;
- The Outdoor Monitoring System;
- The Dynamometer;
- The FORA Scale;
- The BP Monitor;
- The Games;
- The steps.

Once the data are finally stored into the Gateway, they are sent via secure internet connection to the FrailSafe *Cloud*.

Along these two main features, there is one more of big relevance and it is related to the Online Data Analysis, described in detail in D4.2, devoted to perform real-time analysis for identifying some interesting events (loss of balance, fall, etc.) on the data collected by the FrailSafe *Gateway* and in particular from the WWBS.

3.3.2. Clinicians front-end

The Clinical staff front-ends are meant, above all, to allow this category of users to perform the medical research data collection and to tailor the settings of participant's configurations.

To enhance the data collection needed by the medical research the clinical staff has two main sets of interfaces available:

- The PC and mobile applications, for using the sensors and producing through their usage the data, finally exported in file format or automatically sent to the remote servers;
- The web application, i.e., the Clinical Web Portal, implementing the features needed to collect data about the participants in digital forms and questionnaires formats – via several different processes; This is meant to be used by the clinical staff through the web browser available on the tablet;

Furthermore, the Clinical Web Platform acts as a user interface for the functionalities needed for tailoring and fine-tuning FrailSafe services in order to provide customized individual services.

In details, the clinical-staff is meant to interact with:

- A standalone PC, for downloading the data recorded from the WWBS;
- The clinical web portal, to manage the visits executions and the data collection;
- A mobile application, for visualizing the data monitored by each measurement of the digital scale;
- A mobile application (on the iPad), for recording data and saving them into the Mobil-O-Graph, while visiting the participant;
- Web portals for settings games.

In the following pages a detailed presentation of each one of the previously listed software components can be found.

3.3.2.1. WWBS Software for data downloading

A software suite will be provided to each clinical centre to enable them in using WWBS and to collect data recorded during the week by the end-users. This software is compatible with

Windows 7 and later versions. It is composed by a driver for electronic device recognition and software for data management (SmartScope). All details on how to install them and on driver management are explained in the User Manual that will be provided together with the executable.

Features

SmartScope will enable the clinician to:

- visualise streaming of data from the electronic device;
- save the streaming in a file on the PC;
- visualise recorded data (from files saved on the PC);
- download data recorded on the electronic device;
- save them on the PC;
- convert file format in other standard formats: .cvs or .edf.

Data I/Os

Data that can be monitored using WWBS software are:

- ECG;
- heart rate;
- respiration;
- breathing rate;
- 9 degree of freedom (DoF) from each IMU (or a quaternion as an alternative)
- steps.

During the visualisation of data in streaming also the battery level will be shown.

Input name	Description	Source component
Measurement data	*The measurement data, collected from the WWBS sensors.*	WWBS

Table 15: Input parameters of the WWBS Software

Output name	Description	Destination components
Measurement files	*The measurement data are collected in a file format (csv, edf).*	Clinician Web Portal

Table 16: Output parameters of the WWBS Software.

3.3.2.2. Trial Center Auxiliary Devices

Description

The Trial Center auxiliary devices module is responsible for the collection of data from the third party auxiliary devices used in FrailSafe - i.e. the FORA Weight Scale (WS) and the Mobil-o-graph.

This module can be considered functionally and operatively uniform to the Home Auxiliary Devices module as long as the work and data flow from data collection to remote data storage are almost the same.

For WS devices and the Mobil-O-Graph, there are dedicated mobile applications available respectively from FORA and Agedio, which are used for monitoring the measurements (see Figure 24).

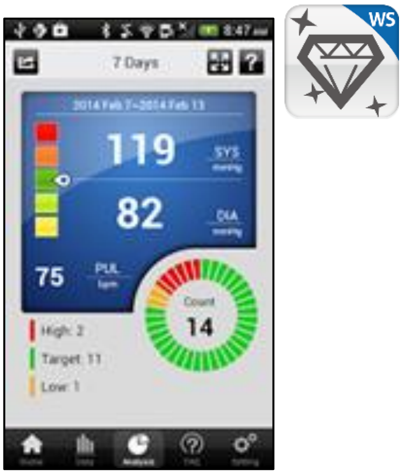


Figure 24

Figure 24 - FORA App

Features

The main functionalities of the auxiliary devices module are the following:

- Collecting the measurements from the corresponding devices;
- Displaying the measurements to the user. Apart from the display embedded in the devices themselves, the devices are wirelessly connected to dedicated mobile applications, which can visualize the measurements using more sophisticated means (charts, plots, etc.), as well as present statistical information from previous usages;
- Third party synchronization. For the WS device, the data collected on the app are synchronized to the FORA cloud system automatically through the dedicated mobile app. Later on relying on the FORA TeleHealth Web API, the data are collected and imported into the FrailSafe system and collected there;
- For the Mobil-O-Graph, “streaming” is performed manually, by submitting the hand-written files to the FrailSafe cloud. Alternatively, it is possible to input through the Clinical Web Portal the measurements received grouped monthly to a specific mail recipient in CSV format. Finally, to perform a more systematic integration, the possibility of receiving such data via API automatically via an explicit secure HTTP call is in discussion with the vendor;
- Alternatively, the measurements can be downloaded as CSV file from the FORA TeleHealth web portal and uploaded manually into the FrailSafe platform through a specific form on the Clinical Web Portal.

Data I/Os

In this section the dependencies in respect to other modules – in matter of I/O data flows – are described.

The dependencies of the auxiliary devices module to the other FrailSafe components are illustrated in Figure 25. The auxiliary devices module accepts input from the auxiliary devices themselves and transmits output to the FORA cloud and finally to the FRAILSAFE cloud. Table 17 and Table 18 describe the specific inputs and outputs, respectively, of the auxiliary devices module.

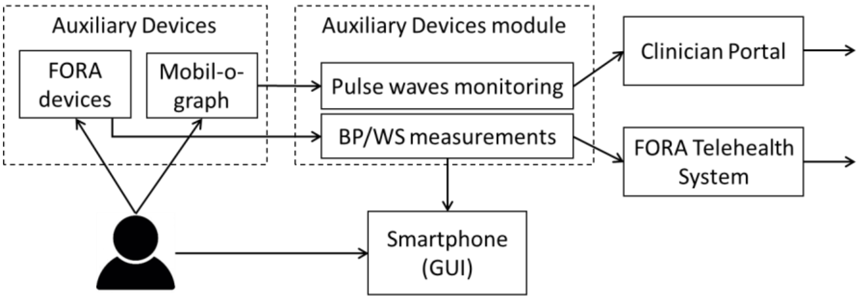


Figure 25: Communication of the auxiliary devices module with other FrailSafe components.

Input name	Description	Source component
Measurement data	*The measurement data, collected from the auxiliary devices. The contents and the format of the data are specific to each auxiliary device.*	Auxiliary devices

Table 17: Input parameters of the auxiliary devices module.

Output name	Description	Destination components
Measurement data	*The measurement data, represented to a format appropriate for the corresponding internal (clinician portal) and external (FORA) streaming portal.*	Clinician Portal FORA Telehealth System

Table 18: Output parameters of the auxiliary devices module.

3.3.2.3. Clinical Web Portal

As described previously in the system architecture overview section, the Clinical Web Portal represents one of the tools used by the clinical staff to collect and generate data inside the flow of the entire FrailSafe system.

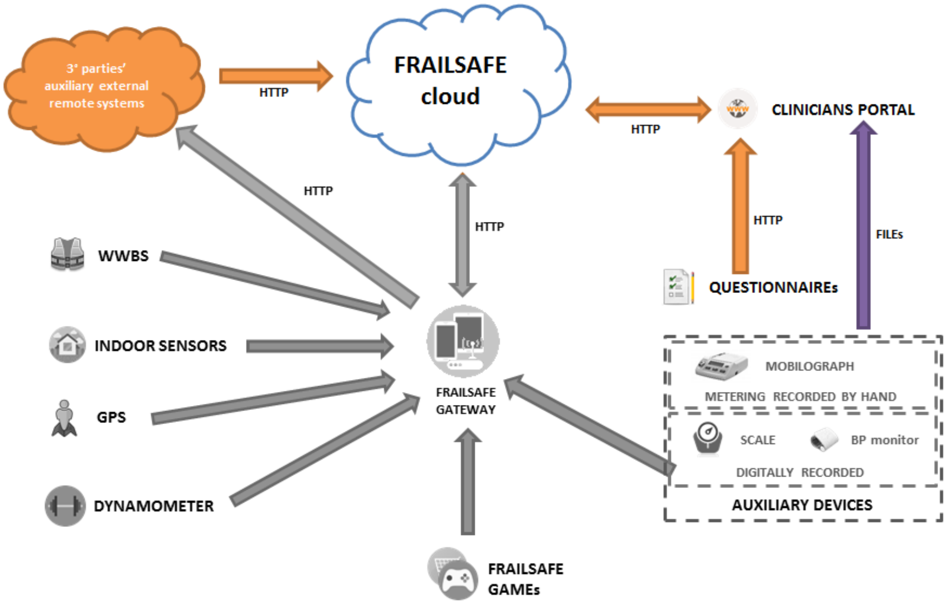


Figure 26 - Clinical Web Portal in FRAILSAFE Architecture

This instrument is a web application where the clinical staff can capture, review, manage, store, analyse and report all the data in various formats relating to the participants of the study. The main purpose of the portal is to help the medical staff in their clinical trials. In this way, it can be considered as what the clinical people call "electronic Case Report Form".

Description

A Case Report Form (CRF) is designed to collect the participant's information, measurements, etc. during a clinical trial; its development represents a significant part of the clinical trial and can affect study success. Site personnel capture the subject's data on the CRF, which is collected during their participation in a clinical trial.

The International Conference on Harmonization Guidelines for Good Clinical Practice define the CRF as:

A printed, optical or electronic document designed to record all of the protocol.

Case report form designing requires enormous planning and attention to minute detail. CRF should be designed for optimal collection of data in accordance with the study protocol compliance, regulatory requirements and shall enable the researcher to test the hypothesis or answer the trial related questions. A well-designed CRF should represent the essential contents of the study protocol and, in an ideal situation, CRF is designed once the study protocol is finalized. It can be prepared either concurrently along with the protocol development, but may result in many versions, and hence needs to be version controlled.

In the current global scenario of the FrailSafe project, the development of an electronic CRF (from now on *eCRF*) has been preferred over a usual paper CRF for the following benefits:

- Eliminate unnecessary duplication of data;
- Reduce the possibility for transcription errors;
- Encourage entering source data during a subject's visit, where appropriate;
- Eliminate transcription of source data prior to entry into an eCRF;
- Facilitate remote monitoring of data;

- Promote real-time access for data review;
- Facilitate the collection of accurate and complete data;
- Possibility to have auto generate data (test result, calculate formulas, etc.);
- Easier data export and data analysis.

As discussed previously, a continuous interaction between technical partners and clinical partner has been necessary to define all the features of the entire process with zero/minimal errors and to fit the web portal with the protocol of the clinical trial and with the entire system architecture.

Features

The eCRF aims to help the clinical staff during the trials with some features that are necessary to complete the participant's study following the usual clinical protocol.

Based on the analysis of individual elements of the user's activities, the following features have been defined and implemented:

- *User Management* This service enables the user, with System Administration privileges, to create and manage login credentials for other users. These users are members of the clinical staff and their job (clinical investigator, technical investigator, secretary, etc.) defines the available features for each one;
- *Participant Management* The medical users register the own subjects of the study to the system adding information about the person (identification code, date of entry in the study, gender, etc.) and the criteria of the study (consent provider, inability to walk, etc.);
- *Trials Management* This feature enables the clinical staff to edit the trial information (data, time, etc.) and view the status (incomplete, complete);
- *Device Management* The medical users have the possibility to assign the devices (BPMonitor, Mobil-O-Graph, etc.), used in FrailSafe ecosystem, to the participants to keep track of the borrowing schedule of that;
- *Collect Data* This feature represents the main objective of the eCRF system. Clinical users collect information of the participant by filling some questionnaires relating to different contents (Clinical History, Cognitive Evaluation, Nutritional Assessment, etc.) or uploading files created by devices or by the participants themselves (written text, sketch, etc.).

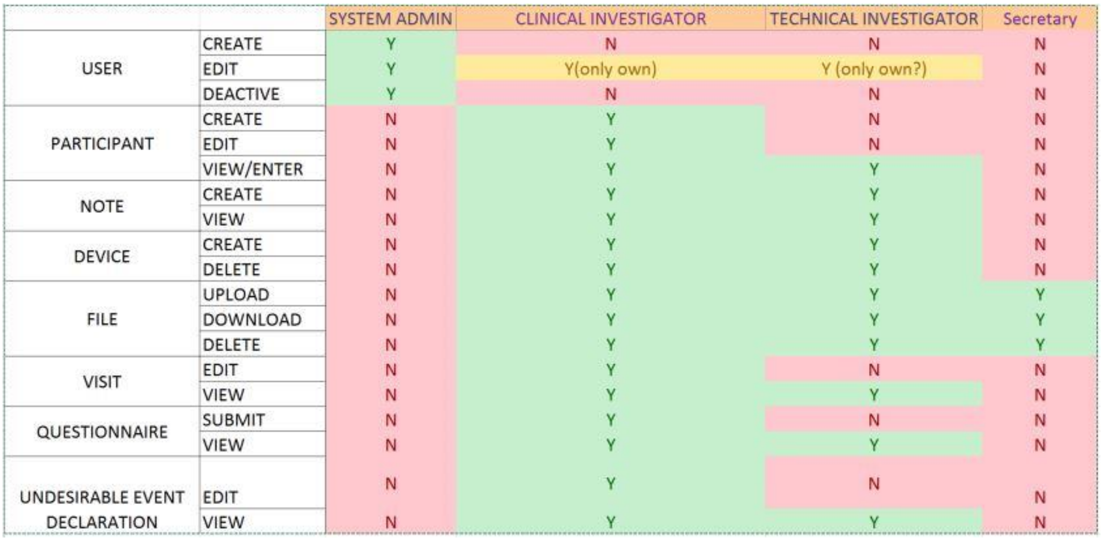
There are some other aspects of the application that are useful to take into consideration. In particular some of the other characteristics of the Clinical Web Portal are:

- *Cross-platform and Multi-device*
  eCRF is a web application, with a client–server service oriented architecture software application that looks like a website written in a standard format such as HTML and JavaScript, which are supported by a variety of web browsers.
  This approach allows the users to use the portal on every device that has a web browser and an internet connection with no specific operative systems (laptop, smartphone, tablet, etc.).
  The main drawback of this approach is that more time, both in design and development phase, is required to consider how and where the different contents

have to been shown in matter of displays - with various resolutions and sometimes with some browsers' features differences - and in test phase to verify the correct usability for the user on every device and browser.

- *Authentication*
  Authentication will be provided according to the Identity and Security modules, described later on in this document – see *Module Identity and Security* .

- *Authorization*
  Different categories of users are defined a priori by the clinical staff settings according to job competency, authority, and responsibility. Users of different categories have different levels of access to the system resources according to the tasks assigned to their role;



		SYSTEM ADMIN	CLINICAL INVESTIGATOR	TECHNICAL INVESTIGATOR	Secretary
USER	CREATE	Y	N	N	N
	EDIT	Y	Y(only own)	Y (only own?)	N
	DEACTIVE	Y	N	N	N
PARTICIPANT	CREATE	N	Y	N	N
	EDIT	N	Y	N	N
	VIEW/ENTER	N	Y	Y	N
NOTE	CREATE	N	Y	Y	N
	VIEW	N	Y	Y	N
DEVICE	CREATE	N	Y	Y	N
	DELETE	N	Y	Y	N
FILE	UPLOAD	N	Y	Y	Y
	DOWNLOAD	N	Y	Y	Y
	DELETE	N	Y	Y	Y
VISIT	EDIT	N	Y	N	N
	VIEW	N	Y	Y	N
QUESTIONNAIRE	SUBMIT	N	Y	N	N
	VIEW	N	Y	Y	N
UNDESIRABLE EVENT DECLARATION	EDIT	N	Y	N	N
	VIEW	N	Y	Y	N

Figure 27 - eCRF role/rules definition

- *Encryption data and connection*
  The data collected in the eCRF are considered as highly sensitive data so the entire system must guarantee the highest level of security.
  The data are not written in plain text in the databases, they are encrypted before to being stored and the decryption requires a secret key. These processes are transparent to the system user during the use of eCRF.
  A protocol for secure communication over a computer network which is widely used on the Internet is the so called HTTPS and it is used in the system as described later on this document – see chapter *System deployment and integration*.

- *Data quality*
  One of the advantages of using an electronic CRF instead of paper is the possibility to check the data by the system before storing them in databases to reduce the possibility of errors.
  In the eCRF the clinical staff collects the participant's information during the trial in the same way used with paper that is with questionnaires, forms, etc. that are displayed on the device screen. During the data addition or before saving, the system checks the coherence of the data and in case something is not correct it helps the user with feedbacks. For example, the participant information cannot be inserted in the system if all the mandatory fields are not filled or it is not allowed to type characters in a field that is a measurement value, etc.

- *Multilanguage*
  The eCRF system in the FrailSafe project is used in three different countries (France, Greece, Cyprus). For this reason, it is necessary that the entire portal supports different languages to help the clinical staff and the participants during the trials to use only one common language to avoid translation errors or misunderstandings.

Data I/Os

With the use of computerized systems for capturing clinical investigation data, it is common to find at least some source data recorded electronically.

An electronic record is any combination of text, graphics, data, audio, pictorial, or other information represented in digital form that is created, modified, maintained, archived, retrieved, or distributed by a computer system. The eCRF is an example of an electronic record.

A data element in any eCRF represents the smallest unit of observation captured for a subject in a clinical investigation. Examples of data elements include weight, systolic blood pressure measurement, written text, or other clinical observations made and documented during a study.

Each data element is associated with an authorized data originator what in computer science is called “Input”.

Examples of data originators:

- Clinical investigators;

- Clinical investigation subjects;

- Medical devices (e.g. Blood Pressure Monitor, Mobil-O-Graph, etc.);

- Records from external remote system.

Many data elements in a clinical investigation can be obtained during a study visit and can be entered directly into the eCRF either manually or electronically.

On one side an example of manually added data is the action of an authorized person who transcribes the data from paper or electronic source documents into the system or uploads a file with the measurements recorded from the medical devices or a photo of the participant written text.

On the other side an example of electronically added data is the automatic process of data transferring from the auxiliary device mobile application to the FrailSafe cloud through the various means adopted by each one of these devices.

The eCRF includes the capability to record who entered or generated the data and when it was entered or generated. Data element identifiers should be attached to each data element as it is entered or transmitted by the originator into the eCRF.

Data element identifiers contain the following:

- Originators of the data element;
- Date and time when the data element was entered into the eCRF;
- Clinical investigation subjects to which the data element belongs.

Input name	Description	Source component
User interaction	*Data from the interaction with the FrailSafe Clinical Web Portal by the clinical staff during the participant trials.*	FrailSafe Clinical Web Portal pages (forms, questionnaires, etc.)
Measurement data	*The measurement data, represented to a format appropriate to upload in the portal (database record, file).*	Auxiliary devices
External records	*The record data from the FORA Telehealth devices. The data are got by the services exposed by FORA Telehealth system*	FORA Telehealth cloud

Table 19: Input parameters of the clinical Web portal.

Output name	Description	Destination component
User credentials	*User login and password of the clinical person requesting access to portal.*	FrailSafe cloud
Participant information and measurements	*Data collected by the clinicians during the trials.*	FrailSafe cloud

Table 20: Output parameters of the clinical Web portal.

3.3.3. Server-side Back-end

The beating heart of the FrailSafe system is represented by its server-side modules. Several processes work remotely in background to provide the services made available by the system to the end-users. These back-end components provide some of the core features of the system.

In particular:

- They are meant to store securely and remotely the data produced by the different client front-ends;
- They are responsible of doing all the different data processing and elaborations that produce additional data and results;
- They make available all the services consumed by the end-users through proper front-ends;
- They provide the secure and private users' account management features to guarantee a proper handling of users' access data;
- They implement the security guaranteeing the correct exchange of data among the different clients and the server infrastructure.

Each module is described in the following sections but, before getting into those details, it is worthy to spend still a few words on the overall picture of the server-side back-end system.

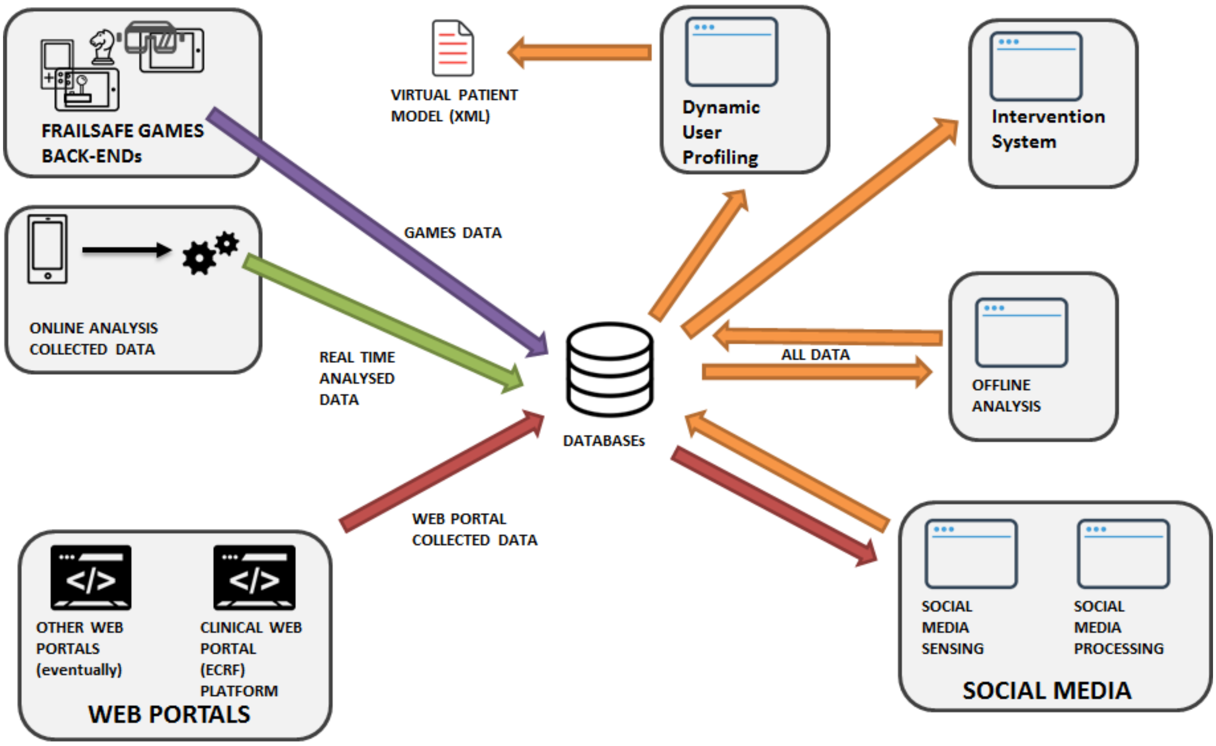


Figure 28 - Server-side back-end processes overview

A number of client front-ends are meant to produce an amount of data going as input into the system. These incoming data are securely transferred - and stored – into the server-side storage module – i.e. databases – thanks to specific modules.

These modules, in charge of this and all the other interactions with the clients, are:

- The FrailSafe GAMEs back-end sub-system, that provides the interfaces to exchange data with the games applications used on the tablet and the infrastructure to tailor and manage the gaming activities;

- The Clinical Web Portal server-side part, that is responsible for providing all the interfaces to its client-side counterpart for making the Clinical Web Portal work.

All these inputs are coming for being stored into a persistence layer, that is composed by several different databases and storage systems (relational databases, non-relational databases and physical file-systems).

On the other side of the FrailSafe cloud sub-system, there are those modules performing all the elaborations and the post-processing tasks on the stored data. The main outputs coming from these components are the so-called Virtual Patient Model (VPM) and a number of analysed and elaborated data made available through the so-called Intervention System module.

These modules are:

- The Social Media components, i.e. the Social Media Sensing and the Social Media Processing, that are in charge of elaborating and extracting data about the social behaviours of the participants;

- The Offline Analysis, performing most of the post-processing operations on the data available in all the data storages;

- The Intervention System, that is the sub-system in charge of providing feedbacks in function of the detected events to the participant and the clinical staff;

- The Dynamic User Profiling module, that is responsible for elaborating the data available in all the data storages to produce and update the Virtual Patient Model of each participant (i.e., the abstraction of the data useful to tailor the system's services to each individual).

A detailed description of each server-side back-end module is presented in the following Section.

3.3.3.1. Module Social Media Sensing

This module leads to Deliverable D4.4a, entitled Linguistic Corpus. In this document, a report on the social data collection phase, in which e-mails, Facebook posts and Twitter messages from several participants will be gathered and tagged according to each participant's mental frailty condition, will be made. Moreover, this linguistic corpus will focus in the Greek and French languages.

Description

The main idea of this module is to measure social interaction of ageing people as well as social and behavioural parameters that emotionally characterize their scripts. The dataset will be collected with use of questionnaires and by implementing crawlers for fetching users' social media texts.

The plan is to collect data and to investigate techniques that connect text features and multiple values of the Big Five personality traits with symptoms of frailty. The training classification phase aims at predicting / characterizing frailty based on the written scripts.

The self-filled questionnaires are:

- *Social interaction*, which is measured by the number of incoming/outgoing phone calls/SMS, emails, use of social networks etc.;

- Data collection of *written texts*, where participants are asked for written text. Specifically, they are asked to think of a major life event (prompts for life events are available such as weddings, child's birth, professional achievements, etc.) and in following to describe in written text an attached picture;

- *Big five personality trait*. There are works that connect mental disorders with the Big Five personality traits and works that try to exploit this information employing social media.

Features

The hand-written questionnaires of Big Five are transferred to .csv in format and each trait is calculated by the average of the score (scale 1-5) or reverse score (6 –score) of certain questions in the questionnaire as it is implemented in our code.

The output of the code is stored in the following format:

Subject, Extraversion, Agreeableness, Conscientiousness, Neuroticism, Openness.

Furthermore, we have implemented a methodology derived from the study of F.Celli[8] that use unsupervised learning techniques so as to recognize the Personality Traits for an

[8] Celli, Fabio. "Unsupervised personality recognition for social network sites."*Proc. of Sixth International Conference on Digital Society*. 2012.

individual using their texts. This implementation could be used in case of missing data from the questionnaire.

The implementation is in Python 2.7 with external modules of *numpy* and *os*.

The Facebook crawler can retrieve public data, including wall posts, only from Facebook pages, provided developer tokens have been obtained, and not from Facebook accounts. The latter requires special permission from the account owner. The implementation is in Python.

The Twitter crawler can retrieve tweets and retweets from any Twitter account on the condition that the appropriate developer credentials have been obtained prior to crawler launching. The Twitter crawler is developed in Python and also an alternative implementation in Java has been created.

Another critical issue is the design of an appropriate web interface for the execution of the social crawlers from the clinical partners where only the appropriate information will be stored in an SQL database regarding texts collected by the above social crawlers.

Data I/Os

As mentioned above, as input data we have the questionnaires (all three of them) and as output data, we have:

- Users answers regarding the social interaction questionnaire;
- Written texts from questionnaires;
- Output from Big five personality test;
- Texts obtained from Social Media crawlers.

3.3.3.2. Module Social Media Processing (Offline version)

Description

The offline social media processing module (LingTester) is the FrailSafe language analysis tool that aims to process the user’s typed text and detect abnormal behaviour. At this point, the prototype is in early alpha stage, but still it is able to perform classification according to levels of frailty.

LingTester is able to detect signs of mental frailty and personality trait shifts by linguistic processing of a person’s written (typed) messages. The linguistic analysis is performed in several layers (ranging from word spelling to Part of Speech -POS- analysis) utilizing the state of the art models in order to determine the mental states of the participants the input texts exhibit. The linguistic corpus obtained from D4.7 is used both for the initial training and the final passive mode (off-line) testing of the prototype. The current architecture of LingTester led the need to divide the research and implementation of the module in four main areas.

The first area of research relates to the analysis and the issues of the collected subjects’ data. The structural organization of the local database and development of the methods for the management of the local database is another area of research. A simple but very informative and mobile structure for offline data storage along with its necessary manipulation methods is designed and implemented. In the area of the highly critical classification task, the domains of feature extraction, feature selection text classification and model optimization are deeply studied and exhaustively experimented in order to obtain the first acceptable prediction results.

The implementation of the aiding software for the FrailSafe user is another key area of research. With the deployment of technologies like Python and Java a cross-platform approach was achieved and the first semi-integrated user software package was developed successfully.

Submodules architecture

As shown in Figure 29, written text is submitted to the LingTester tool through a predetermined process and is stored within a secure database for further analysis. In order to create the training model, all participant rows are fetched from the offline database and features are extracted for the next step. Each feature utilises different resources and is based on custom developed or different third-party tools. This step is followed by the training module which extracts the prediction model in a binary class format for testing and evaluation (Figure 30 summarizes the LingTester model statistics). This methodology has been repeated multiple times so as to maximise accuracy while also optimising all parameters of the system. The final model is packaged in a way to be more user-friendly. All the aforementioned tools are described thoroughly in the equivalent report of LingTester, Deliverable D4.10.

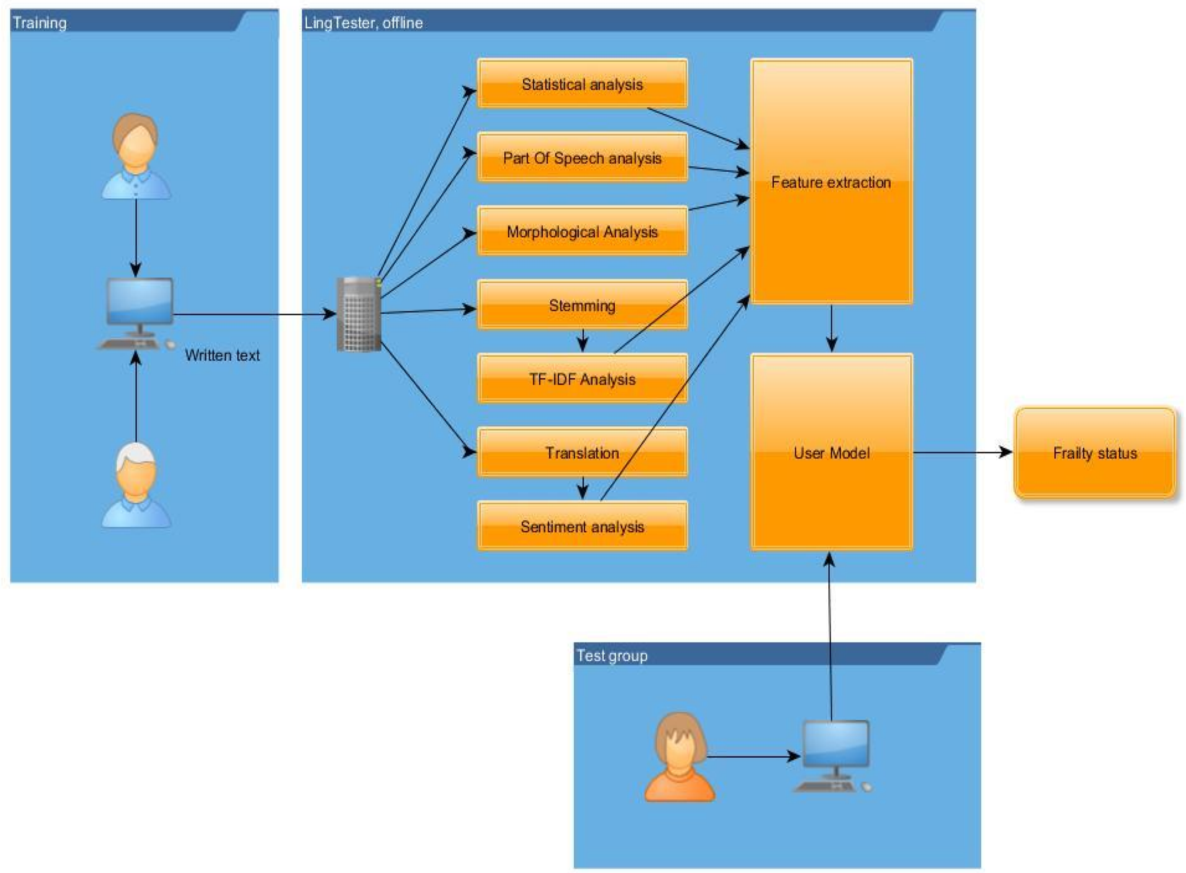
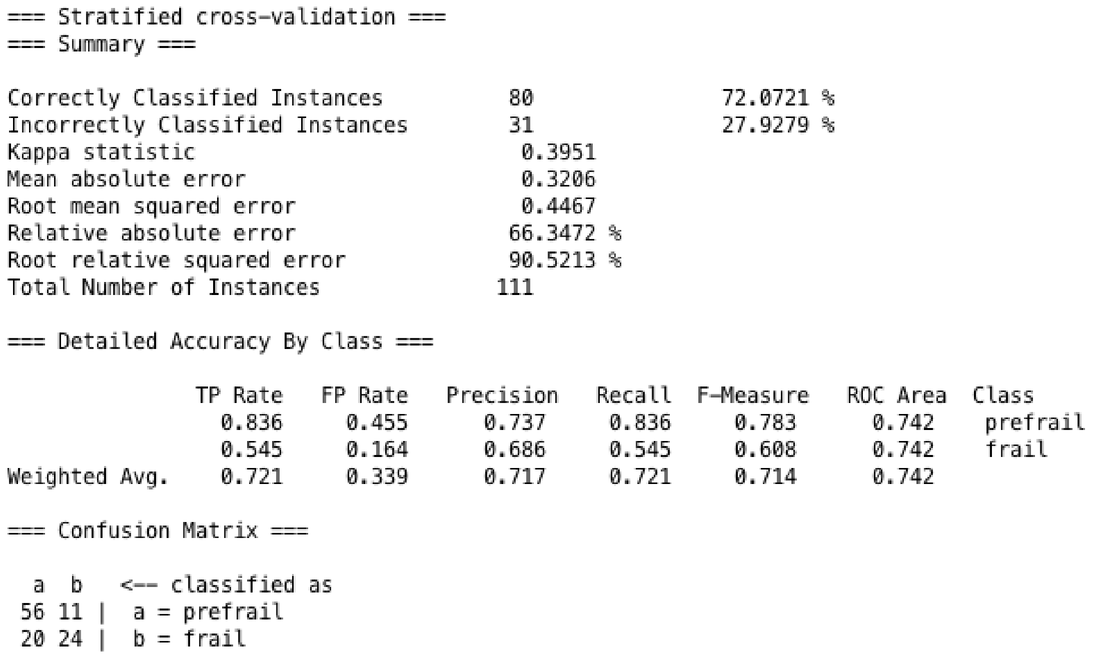


Figure 29 - LingTester workflow



```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          80               72.0721 %
Incorrectly Classified Instances        31               27.9279 %
Kappa statistic                          0.3951
Mean absolute error                      0.3206
Root mean squared error                  0.4467
Relative absolute error                 66.3472 %
Root relative squared error             90.5213 %
Total Number of Instances              111

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.836     0.455     0.737      0.836     0.783      0.742    prefrail
                 0.545     0.164     0.686      0.545     0.608      0.742    frail
Weighted Avg.    0.721     0.339     0.717      0.721     0.714      0.742

=== Confusion Matrix ===

  a  b   <-- classified as
 56 11 |  a = prefrail
 20 24 |  b = frail
```

Figure 30 - LingTester model statistics

3.3.3.3. Module Offline Analysis

Description

The conceptual structure of the offline analysis back-end module is presented in Figure 31. It is directly connected with the FrailSafe Database Management System (DBMS), which stores all data coming from different sources. Several advanced offline analysis algorithms use the data to generate models which will lead to integrative interpretation and better understanding of frailty. The offline analysis process follows, in general, these steps:

- Data Cleaning − In this step, the noise and inconsistent data are removed;
- Data Integration − In this step, multiple data sources are combined;
- Data Selection − In this step, data relevant to the analysis task are retrieved from the database;
- Data Transformation − In this step, data is transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations;
- Data Mining − In this step, intelligent methods are applied in order to extract data patterns. Techniques like clustering and association analysis are among the many different techniques used for data mining;
- Pattern Evaluation − In this step, data patterns are evaluated;
- Knowledge Discovery − In this step, the discovered knowledge is saved in appropriate form to be represented.

The output of the offline data analysis module is saved in the database. Part of the analysis results, such as data aggregation, is fed to the VPM module for representation. Other analysis results might be useful for future analysis purposes.

Features

The offline analysis module is responsible for combining the data stored into the database and provide some useful interpretations. Some of its features include:

- Aggregate data that will be presented in the VPM module;
- Find patterns and associations between external indicators and participant states;
- Detect vital signs of a persons' frailty level changing in a significant manner;
- Identification of motifs (in spatio-temporal signals) and frequently repeated patterns or outliers (corresponding to frailty signs).

Data I/Os

The range of the input data is quite diverse as it includes the raw sensor data generated by the devices, the questionnaires and the medical records of the participants, and the annotations generated by the experts (both clinicians and researchers). Basically, this module is interconnected through the database to almost all the sources of data. The output data of this module is the analysis results that are stored again in the database.

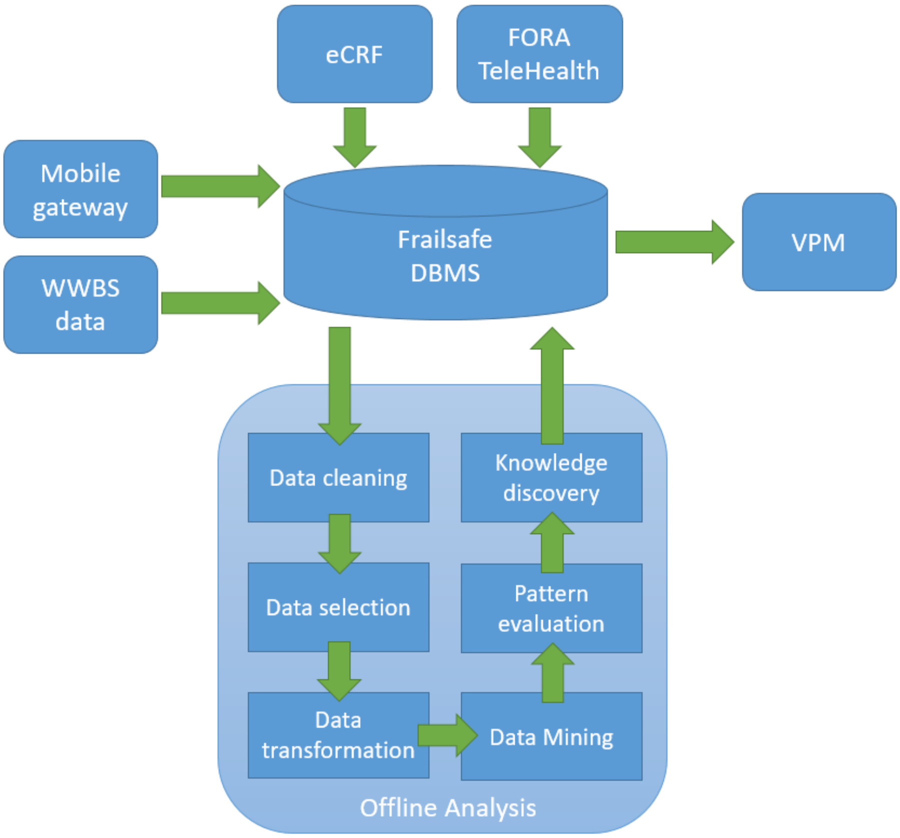


Figure 31 Conceptual structure of the Offline analysis back-end module

3.3.3.4. Module Dynamic User Profiling

Description

A back-end software module is in charge of producing and updating the so-called Virtual Patient Model (VPM). This latter is the formal representation of an individual using the FrailSafe platform in the role of participant and this representation is tailored to this specific subject according to the data collected.

A detailed definition of FrailSafe's personalized Virtual Patient Model (VPM) composed of participant information has been designed and developed.

The corresponding data will be collected by unobtrusively monitoring their everyday life through a variety of embedded and wireless smart indoors and outdoors sensors, social interaction, clinical assessment and self-evaluation tests.

The VPM will be coupled with a monitoring system that will (i) facilitate the analysis of the collected data and frailty feature extraction, (ii) support the *clinical staff* in his/her decision process ranging from general health preservation monitoring to critical situation management, (iii) allow an adaptation of the user interfacing and (iv) provide a personalized feedback to the *participant* via lifestyle change suggestions, behaviour guidelines and medical intervention strategies (see Figure 32).

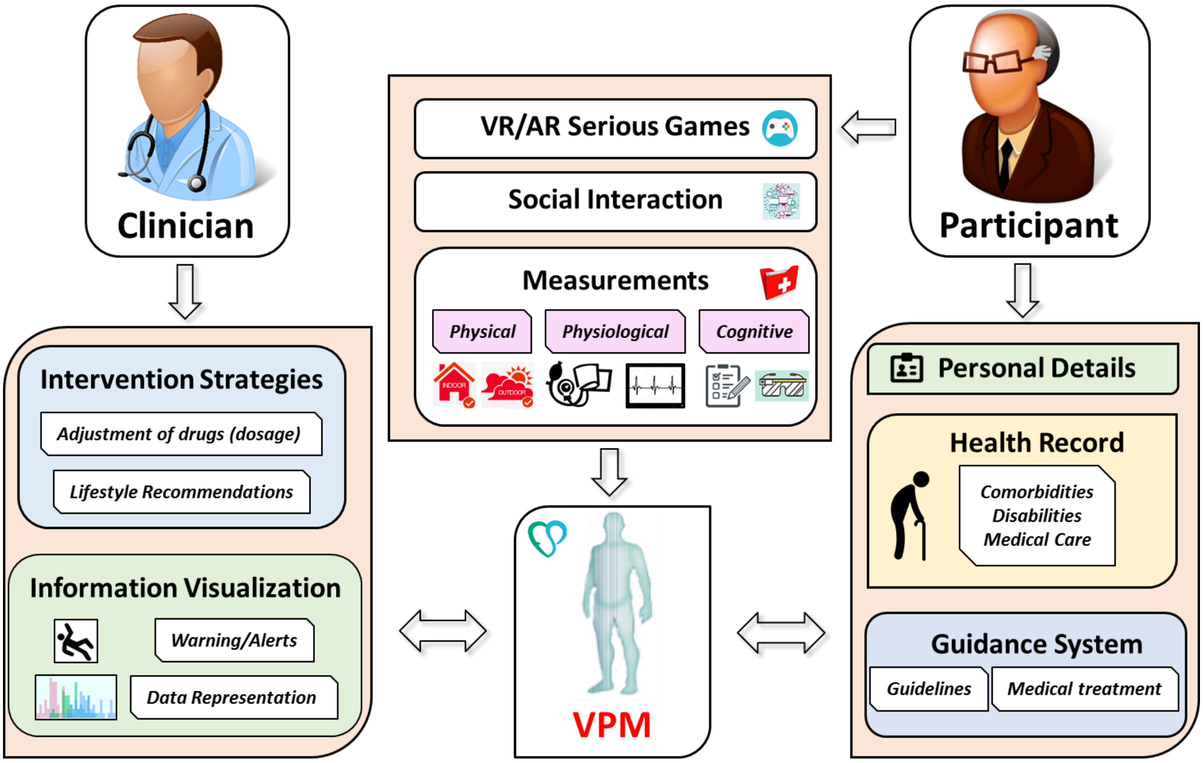


Figure 32. FrailSafe's personalized monitoring system puts the participant's profile (VPM) at the core of the system.

As depicted in Figure 33, the model will be personalised.

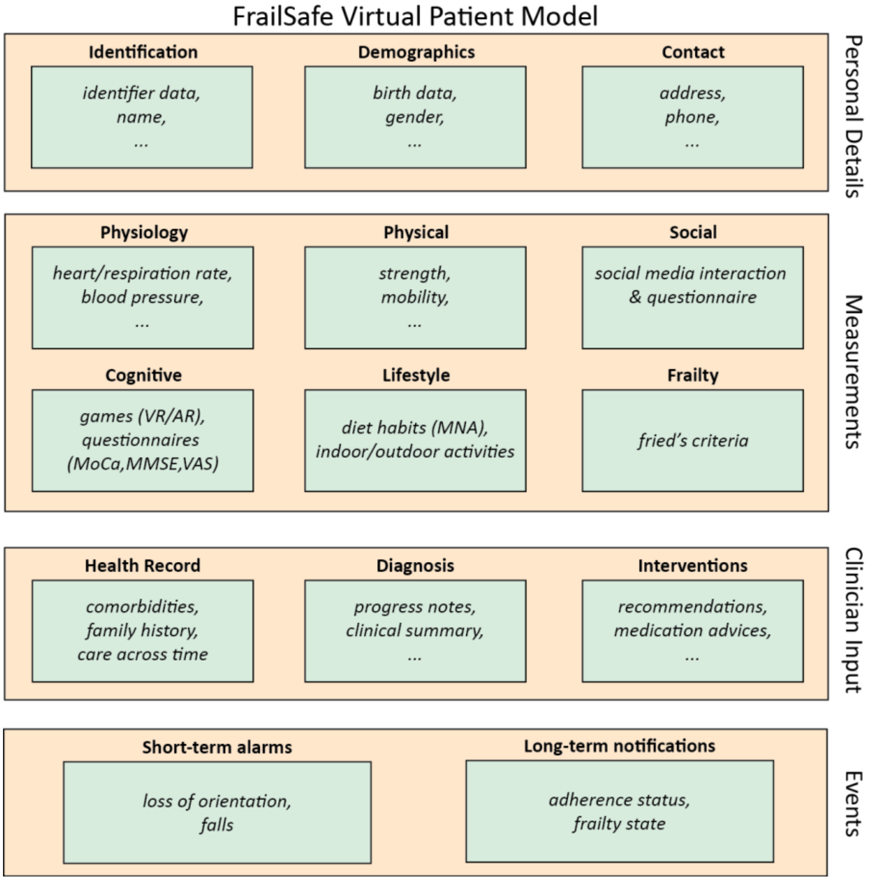

FrailSafe Virtual Patient Model
Identification
identifier data, name, ...
Demographics
birth data, gender, ...
Contact
address, phone, ...
Personal Details
Physiology
heart/respiration rate, blood pressure, ...
Physical
strength, mobility, ...
Social
social media interaction & questionnaire
Cognitive
games (VR/AR), questionnaires (MoCa,MMSE,VAS)
Lifestyle
diet habits (MNA), indoor/outdoor activities
Frailty
fried's criteria
Measurements
Health Record
comorbidities, family history, care across time
Diagnosis
progress notes, clinical summary, ...
Interventions
recommendations, medication advices, ...
Clinician Input
Short-term alarms
loss of orientation, falls
Long-term notifications
adherence status, frailty state
Events

Figure 33 – Overview of the core entities included in the FrailSafe VPM

Frailty related entities are categorized into data related to the (i) user identification, (ii) summary of the data recorded from the integrated sensors as well as the questionnaire and game analysis, (iii) archived medical data essential to the clinicians such as comorbidities and test results, and finally (iv) a list of parameters that are linked to the recognition of short-term (for example fall detection) and long-term events (change of frailty metric). In other words, the VPM will be the participant's virtual alter ego, reflecting his/her medical condition.

Thanks to the VPM and monitoring system, the participant will benefit from a system that gives him/her an optimal overview of his/her (pre-) frail state and at the same time, provide clinicians with a tool that enable them to take the right decision concerning frailty prevention.

Features

The VPM can be seen as a relational database management system (RDBMS), a special system software that is used to manage the organization, storage, access, security and integrity of data. Thus, the main functionalities of this module can be summarized in the following key components:

- Data Storage Management
  - Users do not need to know how VPM data is stored or manipulated;

- Multi User Access Control
  - The VPM ensures that multiple users (participant, family, clinical staff) can access the database concurrently without compromising the integrity of the database;

- Database Communication Interfaces
  - The VPM can connect to third-party systems to visualize data information;
  - End users can generate answers to queries by filling in screen forms.

Data I/Os

In this section the dependencies with respect to other modules – in matter of I/O data flows – are described.

The VPM module is connected to the core database (see Figure 34) receiving an extensive set of different data, which are collected by all the sources (sensors and applications) via the mobile gateway and stored in the main FrailSafe database and are processed appropriately (via the online and offline feature analysis).

These data are available for accessing through a variety of personalized intervention solutions such as the Information Visualization, the Patient Guidance and Games platforms.

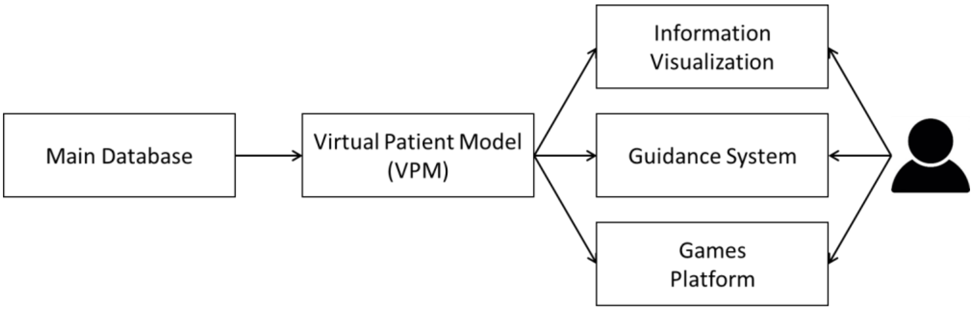


Figure 34. Communication of the VPM module with other FrailSafe components.

More specifically, concerning sensing and monitoring information, the list of the most significant processed data can be classified, according to the sampling frequency they are collected, into two categories: (i) static (offline) and (ii) dynamic (continuous or fixed sampling). In the former class, general information related to the participant identification, demographic information and contact details is mainly included. Current version of participant's electronic health record can also be part of this class.

On the contrary, apart from the recorded sensor measurements which might have significant predictive value for frailty, the data essential to the clinical expert for performing diagnosis and interventions is also included in dynamic entities category.

The latter data can be classified, according to the sampling collection rate into (i) real-time measurements (such as heart rate, respiration rate, mobility, instability/falls) and (ii) daily/weekly measurements (such as strength, blood pressure, arterial stiffness, body mass/weight/surface, cognitive state via games & questionnaires analysis, social interaction via social media analysis, medical adherence and indoor/outdoor activities). Table 21 analytically shows the data collected by the VPM, while Figure 35 illustrates the corresponding relational database model.

#	Parameters	High Level Information Description
1	**Identification**	*1.Participant's id*
2	**Demographics**	*1.Age* *2.Gender*
3	**Frailty**	*1. Frailty Status (Non Frail, Pre-Frail, Frail)* *2. Number of Fried' s criteria satisfied*
4	**Game Analysis**	*For every block of days* *1.Total number of times played/block* *2.Mean number of times played every day* *3.Mean number of attempts to start the game each time* *4.Mean reaction time* *5.Mean Duration of the playing activity* *6.Mean number of pauses/game* *7.Mean Time of pauses*
5	**Heart rate**	*For Every Day-Every block of days* *Heart rate mean value for selected body states* *Examples:* *1.Mean value over time when sitting* *2.Mean value over time when lying* *3.Mean value over time when sleeping* *4.Mean value over time when walking* *5.Mean value over time when walking upstairs* *6.Mean value over time when walking downstairs*
6	**Respiration rate**	*For Every Day-Every block of days* *Heart rate mean value for selected body states* *Examples:* *1.Mean value over time when sitting* *2.Mean value over time when lying* *3.Mean value over time when sleeping* *4.Mean value over time when walking* *5.Mean value over time when walking upstairs* *6.Mean value over time when walking downstairs*
7	**Blood Pressure**	*For Every Day-Every block of days* *Blood Pressure related mean values.* *Subject position: Sitting/Standing* *Specifically:* *1.Mean Blood Pressure when sitting* *2.Mean Blood Pressure when standing* *3.Mean Systolic blood pressure when sitting* *4.Mean Systolic blood pressure when standing* *5.Mean Diastolic blood pressure when sitting* *6.Mean Diastolic blood pressure when standing* *7.Mean Pulse Pressure when sitting* *8.Mean Pulse Pressure when standing*
8	**Weight**	*For Every block of days* *1.Last Value* *2.Mean Value*

		3.Max Value
9	**Body Mass Index**	*For Every block of days* *1.Last BMI value (for each block)* *2.Difference between BMI's values in current and previous block*
10	**Posture**	*For Every Day-Every block of days* *1.Mean time spent standing* *2.Mean time spent sitting* *3.Mean time spent lying*
11	**Steps**	*For Every Day-Plus a mean value for every block of days* *1.Number of Steps* *2.Number of times a walking activity is initiated* *3.Duration of walking activities*
12	**Strength**	*For Every Day-Every block of days* *1.Mean strength value* *2.Max strength value*
13	**Instability/Falls**	*For Every block of days* *1.Falls rate (per block)* *2.Almost/failed falls rate (per block)* *3.Places where falls/almost falls happen (indoors/outdoors)* *4.What type of activity performed* *4.Fall consequences* *5.Physiological state of the subject one minute before for each fall/almost fall* *6.Date of the fall/almost fall*
14	**Indoor Activities**	*For Every day-plus a mean value for every block of days* *For every period of the day* *(morning, noon, afternoon, evening, night):* *(sample activities)* *1.Time spent in the kitchen* *2.Time spent in the bedroom* *3.Time spent in the toilette* *4.Time spent sitting in the kitchen* *5.Time spent lying in bed* *6.Time spent walking inside*
15	**Outdoor Activities**	*For Every day-plus a mean value for every block of days* *For every period of the day* *(morning, noon, afternoon, evening, night):* *(sample activities)* *1.Time spent walking outside* *2.Time spent driving car*
16	**Nutrition**	*For Every day-plus a mean value for every block of days* *Number of meals/day* *Percentage of clinical staff's instructions related to nutrition followed*
17	**Social Interaction**	*For Every day-plus a mean value for every block of days* *1.Number of phone calls* *2.Number of text messages* *3.Time spent speaking at the phone*

		4.Time spent on skype *5.Number of minutes interacting in social media(fb, twitter, instagram)*
18	**Co-morbidities**	*1.Number of active co-morbidities* *2.Co-morbidities' names* *3.Status (Active, Inactive)* *4.Note if each co-morbidity is the leading one (yes, no)* *5.Note if each co-morbidity has a serious effect in participant's functionality (yes, no)* *5.Starting date* *6.ending date (if it's not active)* *7.Description (example: number of stroke events)*
19	**Medication**	*1.Medication names* *2.Reason for the medication* *3.Status (active, inactive)* *4.Starting date* *5.Ending date* *6.Participant's guidance*
20	**Adherence**	*For every day-plus a mean value for every block of days* *1.Percentage of clinical staff's instructions followed by the participant*

Table 21. FrailSafe VPM database representation.

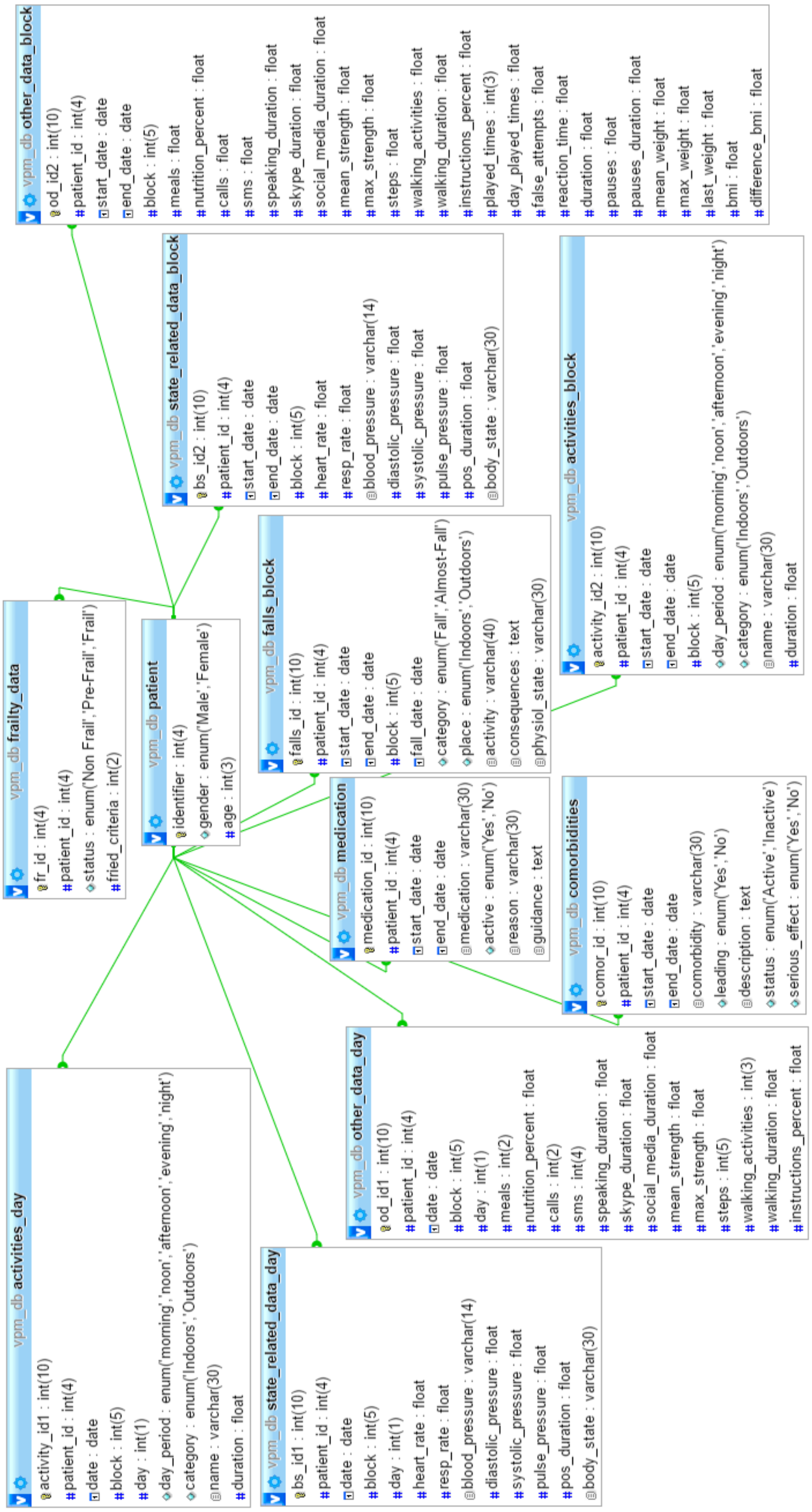


Figure 35. Relational model of FrailSafe Virtual Patient model.

3.3.3.5. Module Intervention System

Description

The intervention system will consist of three important sub-systems:

- **Games Platform Module:** A dynamically synthesized and highly innovative virtual and augmented reality game suite that convers different scenarios measuring parameters of the behavioural, cognitive and physical domains.

- **Recommendations Modules:** A personalized semi-automatic clinician assistance tool to comply with pharmacological (adjustment of medication or drug dosage) and non-pharmacological suggestions (lifestyle, daily activity, exercise, nutrition advices) for each participant. To help diagnosis delivery by medical professionals, this module will augment the optimal planning of the intervention strategies in an automated or self-management manner.

- **Information Visualization Module:** A fully parametric and customizable data visualization framework that supports direct and intuitive visualization of all collected parameters. The aim is to provide knowledge to users in an optimal an intuitive way based on visual analytics techniques, e.g. information (warnings, alerts), processes, experiences.

  The basic module will be equipped with different front-ends targeting the different end-user groups of FrailSafe:

  1. The *participant* (and family) front-end will be equipped with intuitive visualizations of specific only parameters understandable by non-medical personnel (namely, *Personal Guidance System* see Figure 32);

  2. The *clinical staff* front-end will include more complex data representations and visualizations;

  3. A *researcher* front-end will be also provided with the aim of involving querying the database, visual analytics and hypothesis testing.

Features

The intervention system module's main feature is to provide a visualization of the current status of the participant towards the clinician and assist him in designing recommendations and interventions.

3.3.3.6. Module Games Back-End

Description

The FrailSafe Games are part of the Software Front-End components of the system but for implementing some of their features they need to rely on consuming a few Server-side Back-end services hosted on and provided by the FrailSafe Cloud.

In short, they need to exchange data in both directions about a participant and about the data produced by that specific participant while playing a game.

Features

The features required in the FrailSafe Games server module are those to let the front-end component of a game (see 3.3.1.5):

- authenticate a specific user against the FrailSafe platform;
- receive a few participant data;
- receive the game settings associated to the participant using a game;
- send the data produced by a game during a gaming session.

While the first three points of this list are provided thanks to the APIs implemented by other components, on the contrary the fourth is the one implemented specifically in this module and it can be referred to as *Game Sessions Data repository*.

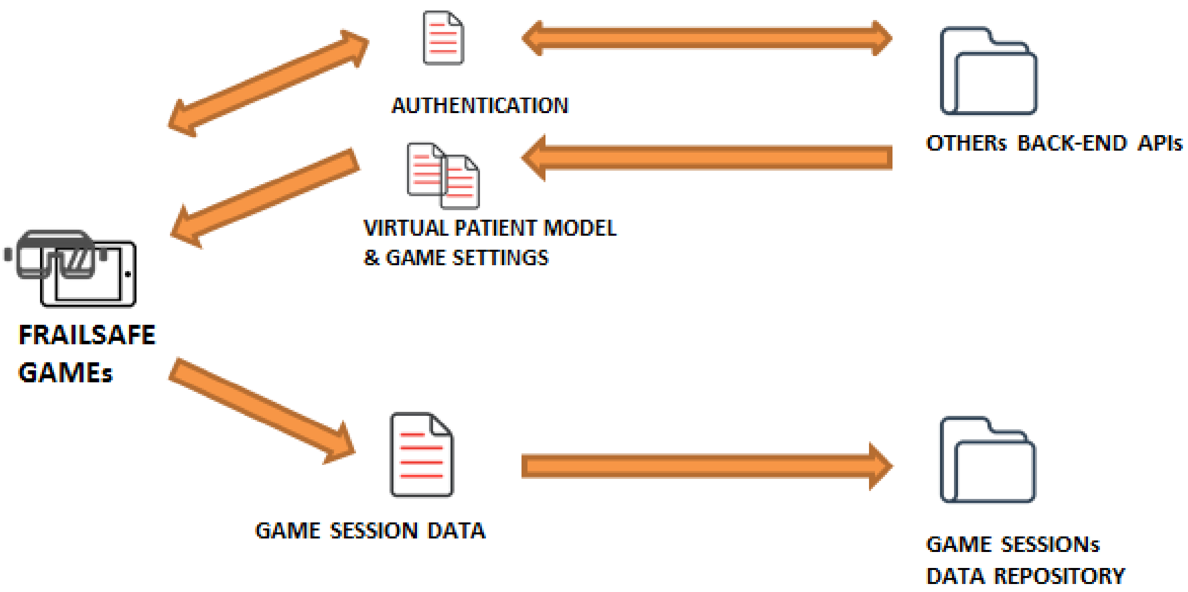


Figure 36 - FrailSafe Games I/Os

Data I/Os

A quick reference schema of the FrailSafe Games interactions between the front-end and the back-end elements is represented at Figure 36. As described here before, the so called *Game Sessions Data Repository* is the one implemented by the FrailSafe Games Server-Side Back-End module. It basically provides a set of authenticated APIs for letting a game to save a game session into a secure cloud storage service.

Table 22 shows a short description of the Inputs of this module.

Input name	Description	Source component
Game Session	*The data produced by a Game during the game-playing time are sent saved into this repository via HTTP calls to the APIs implemented by this module. The contents and the format of the data are specific to each game.*	FrailSafe Games

Table 22: Input parameters of Game Sessions Data Repository.

3.3.3.7. Module Identity and Security

Description

As described at the beginning of the chapter, one of the main features of the server-side processes is to provide the security and the privacy of the users' accounts and platform resources. The aim of the *identity and security module* is to control the accesses to the system resources. This means for example to avoid individuals having unauthorized access to the system or an authorized user requesting a forbidden resource for their own specific role.

As shown in the below picture, the main components of the module are:

- User Management;
- Authentication;
- Authorization;
- Privileges Management.

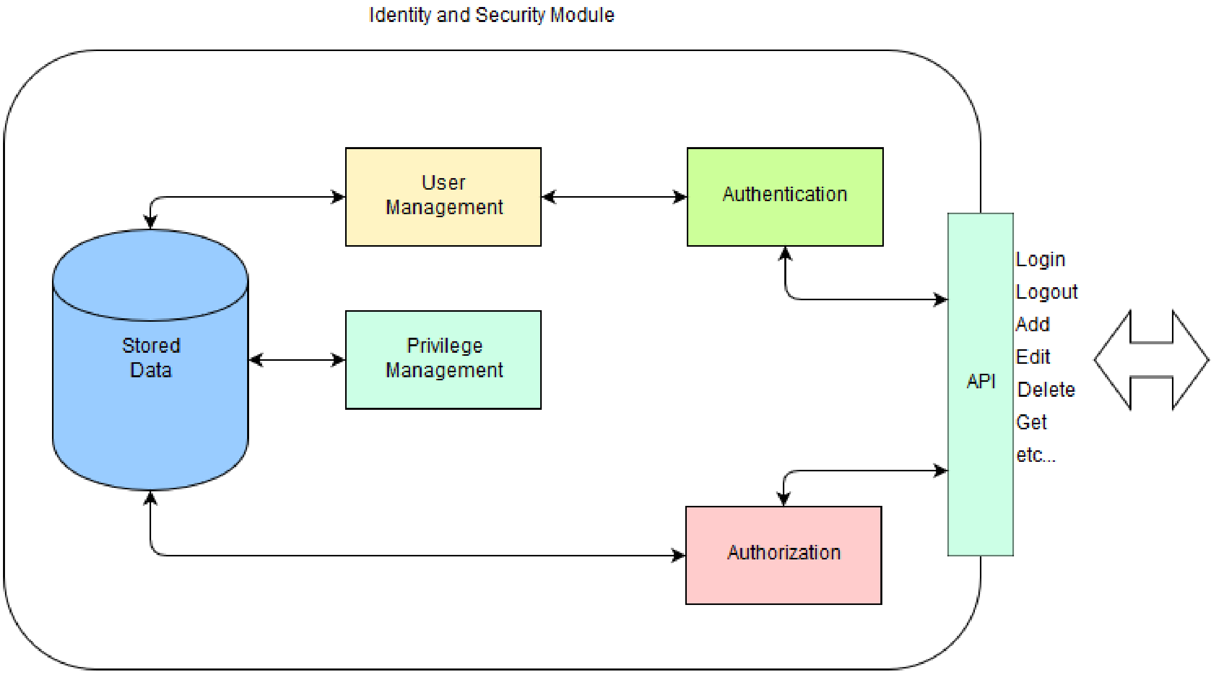


Figure 37 – Identity and Security Module Architecture

User Management

This component has the objective to manage all the operations about the data of the users of the whole ecosystem. The user management system provides functionality to manage both users and personal profiles. Personal profiles are used for personal information such as names, addresses, etc. User information contains user identification number, username, password, etc. for authentication. User information is used for high-level authentication and privileges definitions.

As shown in the picture above, the system provides a set of APIs facilitating the user registration process, password management, user data editing and so on.

Authentication

It is the process of identifying a user, usually based on a valid couple of credentials – i.e. a username and a password - before allowing the access to the system resources. Authentication merely ensures that the individual is who he/she claims to be, but says nothing about the access rights of the individual.

This operation is typically made by a Login API that requests to the User Management if the credentials are valid and if it has success the response is an ID token that allows to get access to the system resources for limited time.

For the FrailSafe Project, a solution using JSON Web Token (JWT) is under evaluation, which is an open standard that defines a compact and self-contained way for securely transmitting information between parties as a JSON object. This information can be verified and trusted because it is digitally signed. JWTs can be signed using a secret (with the HMAC algorithm) or a public/private key pair using RSA.

In the authentication scenario, once the user is logged in, each subsequent request will include the JWT, allowing the user to access routes, services, and resources that are permitted with that token. JSON Web Tokens are a good way of securely transmitting information between parties, because as they can be signed, for example using public/private key pairs, we can be sure that the senders are who they say they are. Additionally, as the signature is calculated using the header and the payload, we can also verify that the content hasn't been tampered with.

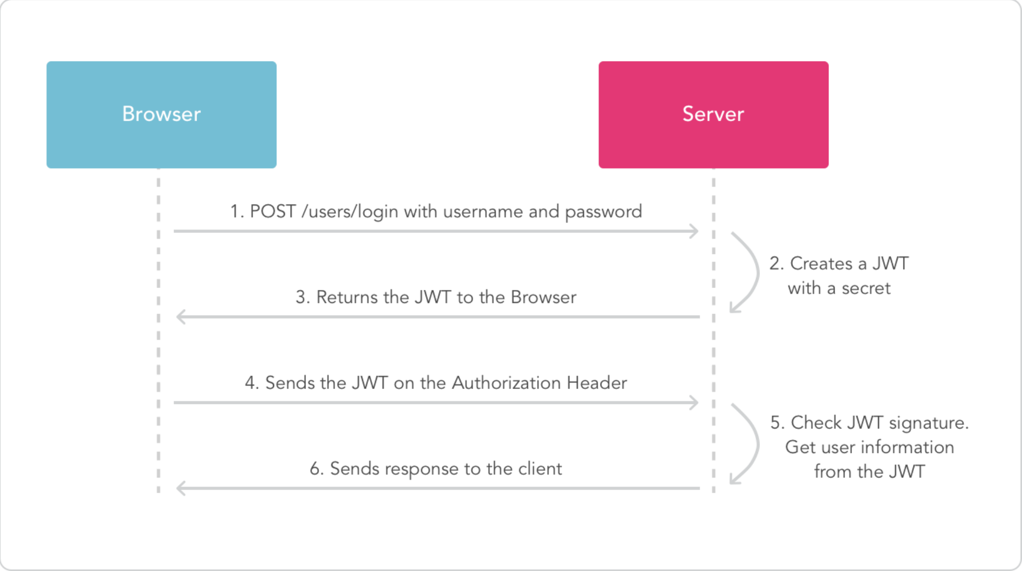

Browser
Server
1. POST /users/login with username and password
2. Creates a JWT with a secret
3. Returns the JWT to the Browser
4. Sends the JWT on the Authorization Header
5. Check JWT signature. Get user information from the JWT
6. Sends response to the client

Figure 38 - JWT Process

Authorization

It is the process of granting or denying a user the access to the system resources based of the own roles. In a Role-Base-Access-Control (RBAC) model, users are assigned roles, and roles are assigned access permissions to protected resources. Users can have any number of different roles concurrently, and this combination of roles determines the user's level of access to protected system resources/functionalities.

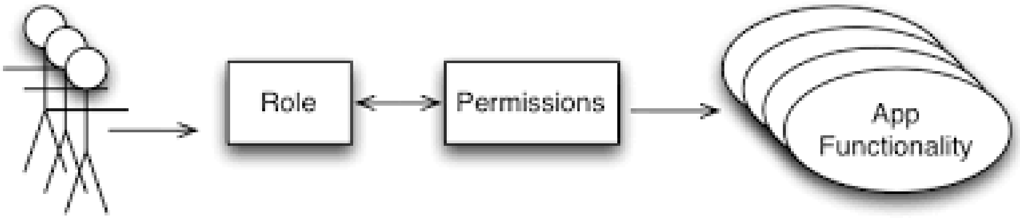
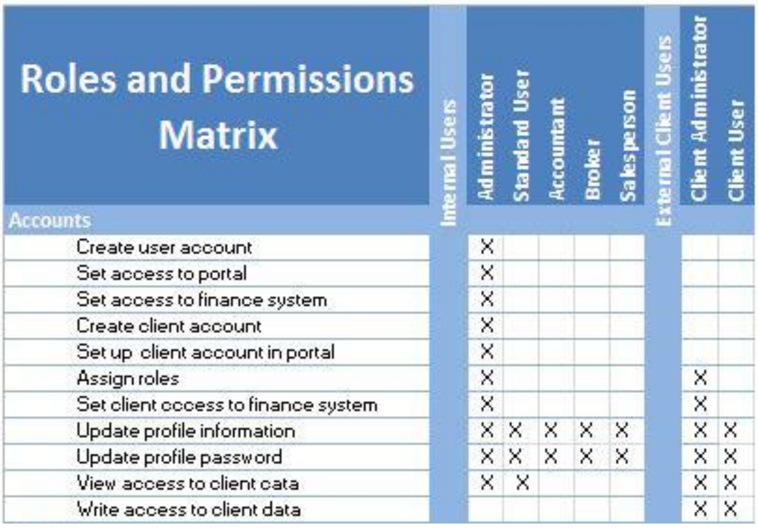


Figure 39 - Role definition

Privilege Management

This component defines the roles and the related right access of each one. Roles are a way to facilitate the granting of multiple privileges or rules to users according to job competency, authority or responsibility. Generally, roles and related privileges are defined in development phase with configuration files or with tables stored in the database which map the roles with the system resources/functionalities.



Roles and Permissions Matrix	Internal Users	Administrator	Standard User	Accountant	Broker	Salesperson	External Client Users	Client Administrator	Client User
Accounts									
Create user account		X							
Set access to portal		X							
Set access to finance system		X							
Create client account		X							
Set up client account in portal		X							
Assign roles		X						X	
Set client cccess to finance system		X						X	
Update profile information		X	X	X	X	X		X	X
Update profile password		X	X	X	X	X		X	X
View access to client cata		X	X					X	X
Write access to client data								X	X

Figure 40 - Roles and Permissions Matrix

4. SYSTEM DEPLOYMENT AND INTEGRATION

This section of the document deals with the integration of the different parts of the FrailSafe project described above. Moreover, this section will show the deployment strategy that will be adopted by the project. Deployment and integration represent a fundamental part of the work done within the FrailSafe platform, so they must be clearly described.

Having a well-integrated platform is one of the main objectives of the Project, and this could only be achieved by defining a clear approach that must be shared among all the components. In the following, we will enter more in the detail of these matters.

As we have seen in the previous sections, the FrailSafe platform is composed of different modules and different parts that must be able to talk to each other to work. Cooperation among components is an essential part of the platform, and hence we have to think about an effective integration approach.

As we can see from the description of the architecture, the FrailSafe platform is structured as a Service Oriented Architecture (SOA), where each component is seen as a service that could be consumed by others.

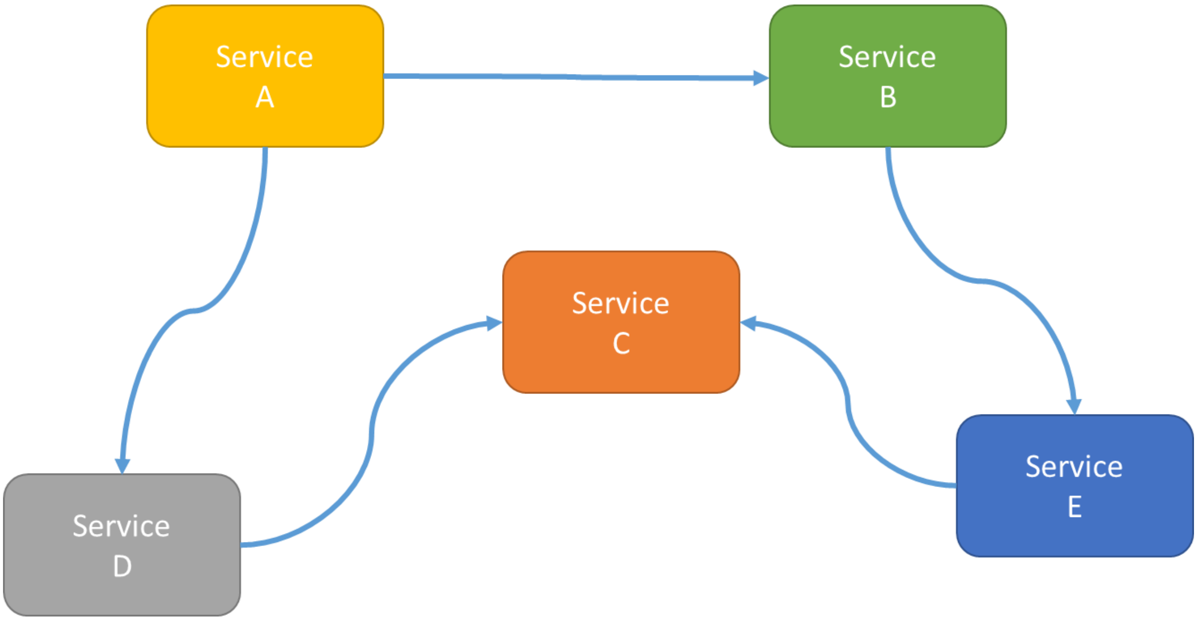


Figure 41. Service Oriented Architecture example

Apart from the advantages of a Service Oriented Architecture in the development of the platform components, the adoption of this paradigm gives us also many benefits for the integration of the platform as a whole. Since each service is seen as a “black box” from the consumers, the developments could be carried out independently from the rest of the system taking as a reference only the specification of the services to expose. This is a notable advantage, especially in a project like FrailSafe when the different parts of the system are developed by different teams belonging to different organizations. Even the versioning of each module can be handled without particular issues, since each new version can be deployed in the system without compromising the whole platform.

4.1. Integration Approach

This set of advantages drive us to pay more attention to the communication between the modules. Since each module is an independent black-box that exposes services, we have to manage the way services are consumed. This mainly regards the definition of a protocol of data exchange between the service and the consumer, that is an essential part of the integration process. In the SOA field there are many different protocols and approaches that could be listed, but the most used are essentially two: Simple Object Access Protocol (SOAP) and Representational State Transfer (REST) architecture. The first one has been used widely in the last decade, since it was the de-facto standard for the SOA, but in the last years REST has gained in popularity mainly due to the fact that is more application-agnostic than SOAP and, moreover, its architecture is based on HTTP, natively supported by a wide range of devices.

Another important aspect that a good integration strategy has to take into account is the format of the data that flows between consumers and services; we can find different types of data, starting from the less structured plain text to the more structured eXtensible Markup Language (XML) or the JavaScript Object Notation (JSON).

Putting all these things together, we can describe the integration approach that will be adopted in FrailSafe as divided substantially in three parts:

- Exploit the Service Oriented Architecture, and deploy each module as an independent service inside the platform;
- Let each module expose its services through Application Programming Interfaces (APIs) that use the REST architecture;
- Use JSON as reference for data exchange.

This approach brings us to the well-known API-based integration, that uses services and API as key points to enforce flexibility and management of an integrated platform.

After having defined the integration approach, we need to focus on the specific methodology to follow when integrating a system. Since the development of the FrailSafe platform is an iterative process based also on the user-centric design - see D1.2 - we have to expect that different versions of every component will rise up during the project. Moreover, we have to take into account that not all the components will be available at the same time. These remarks drive us to the conclusion that applying a “one-shot” integration is almost impossible, apart from the fact that this integration scheme is no more applicable to modern ICT platforms.

For the FrailSafe project, we must follow an approach that iterates the integration among the different releases of the components. With respect to this, a well-organized Service Oriented Architecture can help to achieve this goal.

Our integration approach can be represented by the following pillars:

- All the modules must be developed according to the specifications and must guarantee the backwards compatibility with the older versions of their APIs;
- When a service is released for the first time, it will be deployed and integrated with the whole platform;
- Every time that a new version of the service is released, the old version is replaced with the new one and only the new features will be integrated with the platform;
- Every time a service is deployed, it is tested to verify if all the exposed APIs work as expected;
- A protocol for publication and release of the system components must be communicated by the integrator and followed by all the partners with the aim of guaranteeing the quality and maintainability of the integrated system.

As we can see from the above discussion, the deployment process plays a key role in the whole integration strategy, since it is crucial to obtain a platform that is maintainable and that respects the specifications. Nevertheless, a good deployment infrastructure minimizes the risks of creating security flaws that could be exploited by attackers.

4.2. Deployment infrastructure

Once we have defined the general approach for deployment and integration, we have to shift our attention on the infrastructure where we will put all the developed modules. Starting from a high level, we can say that there are substantially two different infrastructural approaches that can be used: centralized and distributed. In the first case, all the components reside in the same infrastructure and are generally connected using a local network (e.g. LAN). In the second case, the components are distributed in different infrastructures that reside in physically separated environments, and the connection among them is typically realized using a geographical network (e.g. the internet). Both of these approaches have different pros and cons and, generally, the choice among them is driven by the system that we want to realize. In more detail, centralized systems are to be preferred when we have to implement applications and services that can centralise data and processing, taking advantage of the faster (and more secure) communication that we can achieve using a local network among components. On the contrary, when we create a system that runs on different devices, generally at the end-user side, the distributed approach is to prefer.

Dealing with a Service Oriented Architecture gives us more freedom about the choice of the infrastructure to adopt, since this particular kind of architecture fits well on both. From the FrailSafe platform point of view, we are dealing with a system that offers services to clients that connect to it. These clients are typically distributed (smartphones, web browsers) while services are accessed in a centralized fashion. For these reasons, we can imagine to use both the deployment schemas, the distributed one for the frontend and the centralized one for the backend. The following picture shows this architectural separation.

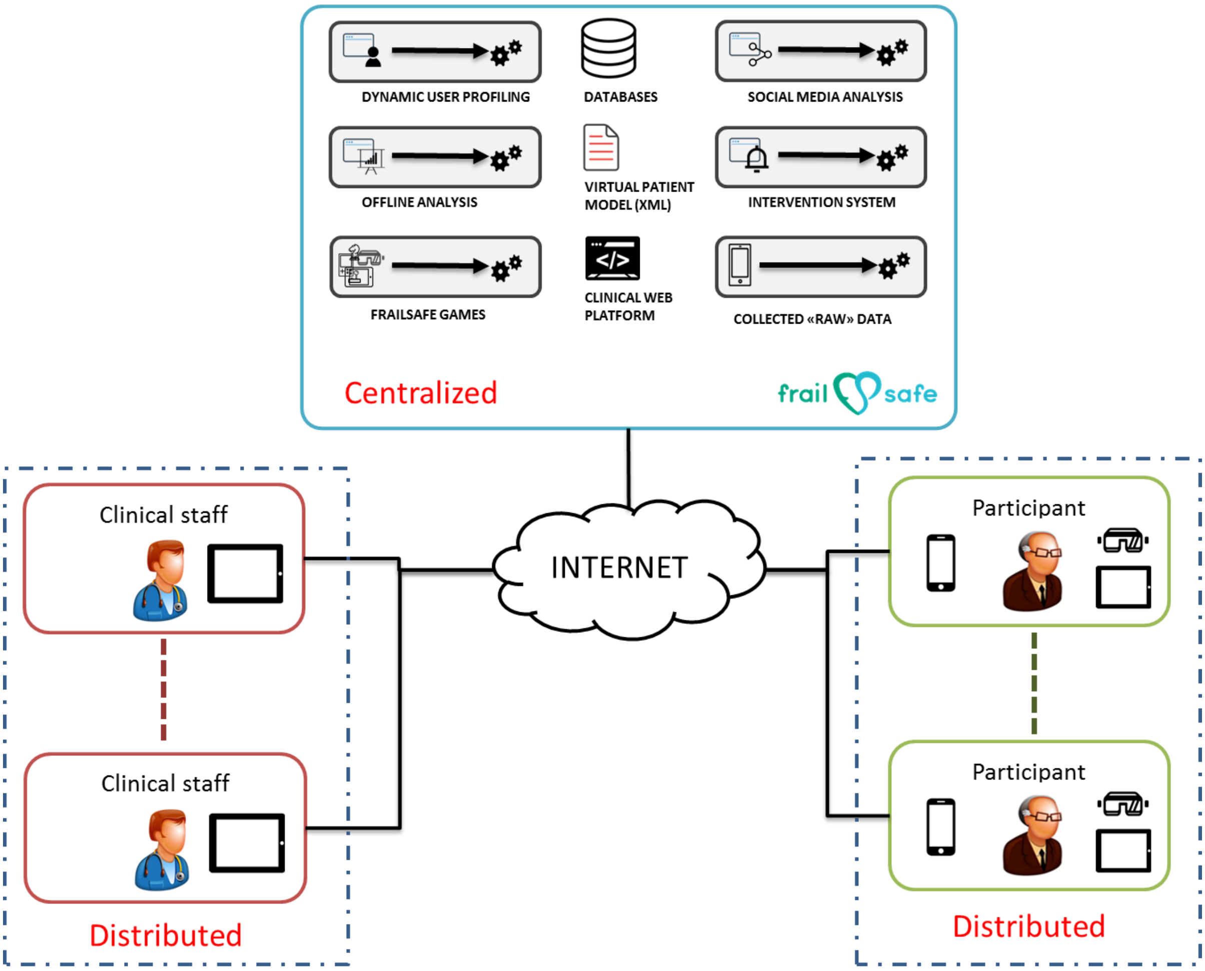


Figure 42. Architectural infrastructure separation

For the frontend part, the distributed system mainly refers to the applications that will be provided to participants and clinicians to interact with the system. These interfaces were described in previous sections and their deployment will rely on some vendor specific technologies, since they will be implemented for devices such as smartphones and tablets.

Regarding the deployment of the backend part, we can choose among different types of systems, but they are substantially divided into three categories:

- On-premise: host all the infrastructure and all the components on a private data center, owned, in our case, by a project partner;
- On public cloud infrastructure: use services offered by a public cloud provider to host the infrastructure and components;
- Hybrid: use both the on-premise and public cloud solution jointly.

The decision about the kind of strategy to use involves many different aspects, and it is discussed in the 6.1 Annex Chapter of this document together with the presentation of the main related issues. Briefly, in FrailSafe was chosen to use the public cloud infrastructure since this is the solution that fits better on our scenario, also helping us to enforce the integration of the whole platform.

The discussion among which kind of cloud service provider to adopt is still ongoing in the project consortium, also because we need to carefully analyse each provider in terms of:

- Operational costs;
- Services that we can use and exploit to empower the platform;
- Security that the provider can offer;

- Compliances of the provider with the European regulations in terms of Privacy and data processing.

Given that, we can describe the infrastructure that we will create to host all the services in the cloud.

a. *Modules deployment*
   Each FRAILSAFE module will be deployed as an independent service inside the infrastructure, for example using a virtual server that will host all the applications and dependencies required by the module to run;

b. *Data storage*
   Data persistence is essential to FrailSafe and we must have a data storage layer that is able to efficiently store and retrieve data. Given the system requirements, we must guarantee to:
   - Persist data in a reliable way: data must be replicated in different storage system to guarantee a high durability.
   - Retrieve the data even if there are some software/hardware faults, with a high availability

   For doing this, we can rely on the storage services offered by the cloud provider, that typically offers high level of availability and durability;

c. *Scalability and load balancing*
   The platform must be able to scale up as the requests increases to guarantee service availability. In the same way, it must be able to scale down when services are less used to minimize waste of resources. For doing this, we will use auto-scaling systems and load balancers that are typically offered by cloud providers;

d. *Networking*
   The networking plays an important role in the platform infrastructure. In our case, we need to create a network structure that allows us to create a separation between services that are exposed to the end-users and services that are only used internally. This separation is realized by creating subnets and route tables that handle the route between different subnets;

e. *Security*
   To enforce the security of the entire platform we need to introduce in the infrastructure some components to avoid malicious attacks or abuse in platform usage. This refers mainly to:
   - Firewalls, to block malicious connections.
   - Encryption systems, to keep data encrypted and hence not directly accessible to malicious user in case of data breaches.
   - Security system to avoid malicious users entering the system.

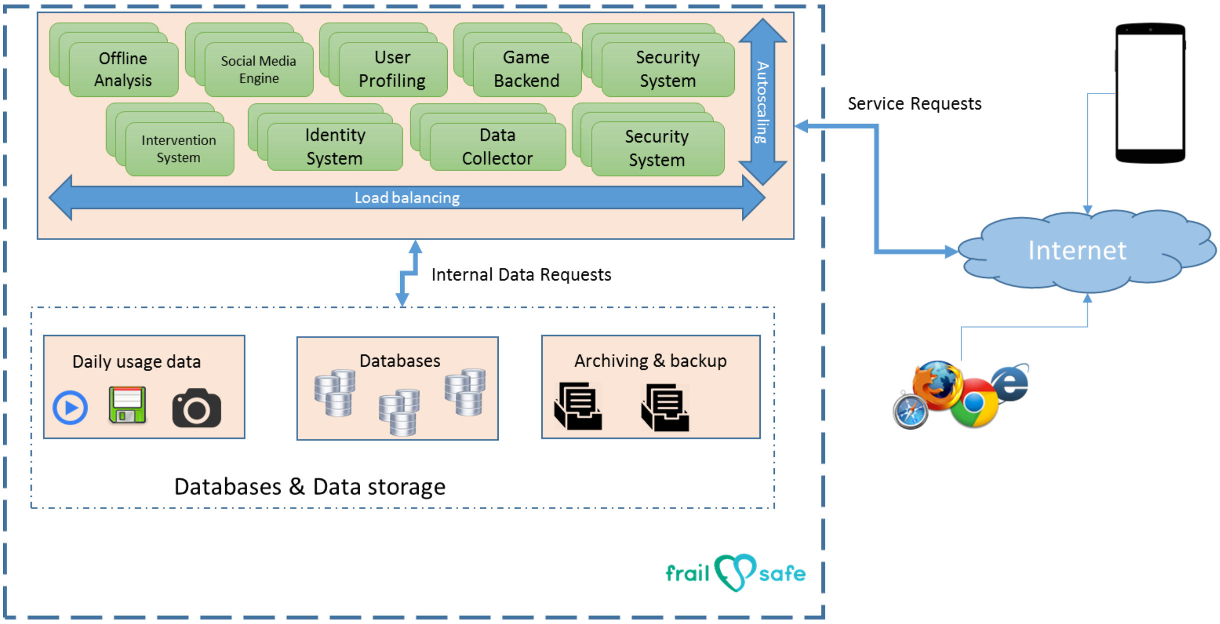


Figure 43. Cloud deployment schema

The picture above (Figure 43) illustrates an example of deployment that could be carried out in a cloud environment. As we can see, modules are deployed as independent services, and the environment is configured in a way that we can scale up or down as needed. Furthermore, we have a data layer that we use for all the data that modules handle. We have divided this data in different categories: Daily Usage Data, Databases and Archiving & Backup. We have made this division since each of them will handle different kinds of data with different requirements in terms of latency, durability, availability, redundancy, etc.

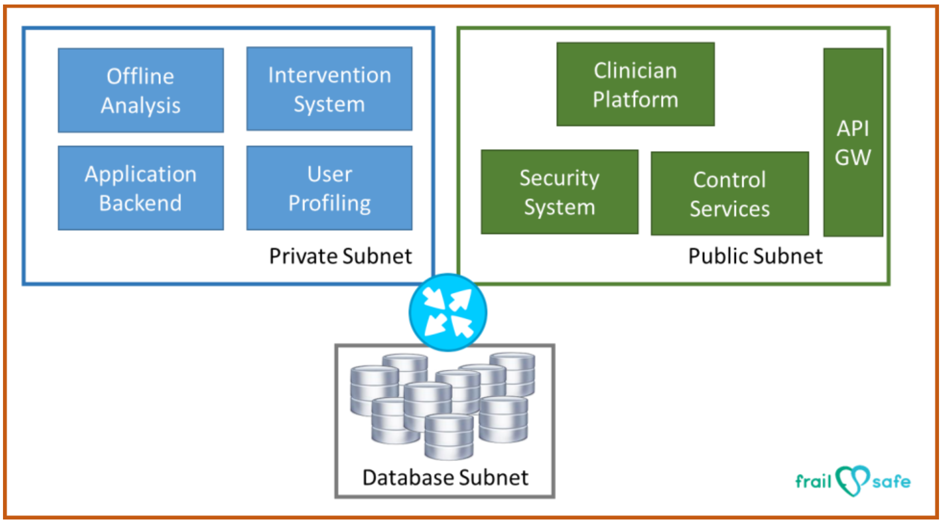


Figure 44. Cloud infrastructure subnets

The picture above shows a way to configure the network infrastructure. We can see that there are three different subnets, namely:

- **Public subnet**: contains all the services that need a directly exposure to the internet, e.g. web servers or control systems such as VPN servers. The modules that belong to this subnet are directly reachable by the internet;

- **Private Subnet**: contains all the other services, that do not require a directly exposure to the internet, e.g. the services that makes elaboration on data. The modules belonging to this subnet are not directly reachable from the internet;

- **Database subnet**: contains all the databases and the data storage of the platform, and is not exposed to the internet

This subnet will be managed by a specific routing table, configured to allow all the subnets to talk to each other: this means that a "public" module will be able to talk with a "private" one (and vice-versa). Similarly, Databases will be reachable by both the "private" and the "public" modules. This network infrastructure has been architected to enforce the security, since only a small set of services are directly reachable from the internet, and hence exposed to security threats. Of course, having such an infrastructure requires some mechanisms to allow "private" modules to reach the internet when needed, and expose some services to the outside. This bring us to introduce two different pieces on our picture:

- A Network Address Translator (NAT) component, that allows "private" modules to reach the internet, for example when a service requires to fetch some resources on the web;

- An API Gateway, that allows "private" modules to expose their API. This component will be illustrated more in detail in the following sections.

4.3. API integration

As seen in the above sections, we have defined all the modules that will be implemented in FrailSafe. These services are connected to each other by means of a service-oriented paradigm, already discussed before. We have also detailed the way a service is exposed, i.e. using REST APIs, and we have defined the data format to use, i.e. the JSON.

Having all the modules exposing services, we have to define a protocol to use them, both by external client and internal modules.

As seen before, there are only a limited number of modules that are directly reachable by the internet, and in most case they expose web pages and resources rather than APIs. This leads to the fact that we need a gateway that allows us to expose all the services to the outside.

Using a unique gateway, we can have several advantages:

- We have a unique entry point for all the API calls. In this way we can use a unique domain name to refer to the API;

- We can have an API access control at the top-level, rather than implement it for each API;

- We can enforce the security by putting the authentication and authorization at the gateway level, avoiding the implementation at the API-level;

- We can lower the overhead in the communication, since the encryption will be made only between the gateway and the clients, leaving all the "internal" communication in plain.

The following picture shows the functionality of the API gateway and how the above-mentioned functionalities will be mapped.

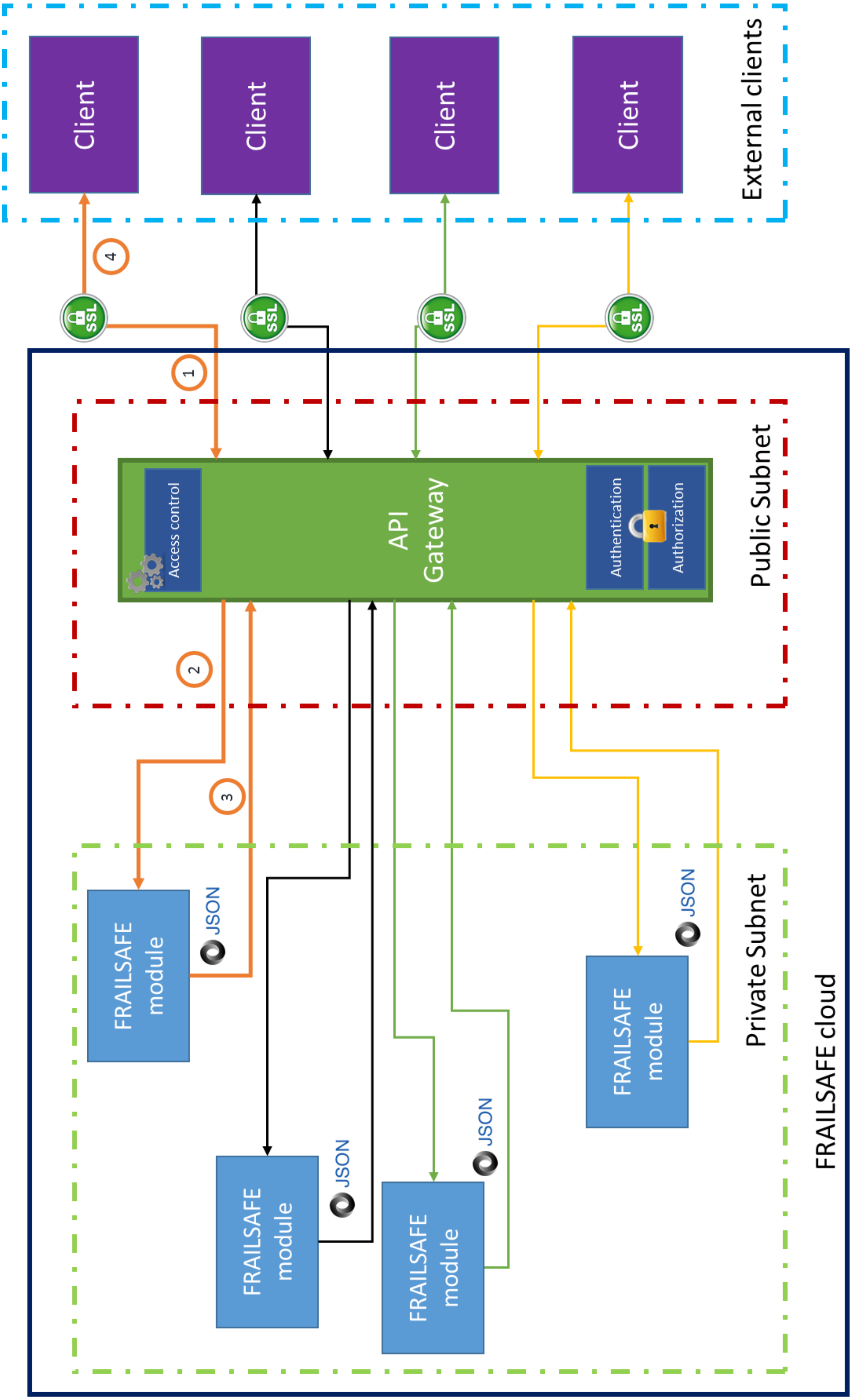


Figure 45. API Gateway integration in FrailSafe cloud

In the picture is also reported a numeric sequence that represents the flow followed by a request made by an external client:

1. An external client made a secure API request directed to the FrailSafe cloud, using the HTTPS protocol;

2. The API gateway redirects the request to the correct module by simply forwarding the whole request (i.e. including parameters and other data) without using any encryption, since from now on all the communications happen within the FrailSafe cloud. It is worth to notice that the request is forwarded if and only if the external client is authenticated and authorized. To achieve these controls, the API gateway verifies the credentials issued by the external client with the FrailSafe security system;

3. The target module receives the request from the API gateway and responds to it. Note that the target module does not have a direct communication with the external client;

4. The API gateway then re-forwards that response to the external client, using the HTTPS protocol.

Using an API gateway as the entry point for all the API calls creates a single point of failure that we must consider. As we see in the next section, the requirements of the system will focus also to the reliability and availability of the overall platform. This is particularly the case of the API gateway, that must be architected and deployed so that it can achieve high percentage of availability, even in case of external attack. These could be obtained by:

- Deploy the gateway in multiple instances, and put them all behind a load balancer and an auto-scaler that take care of instantiating/deleting instances and redirecting the traffic among them;

- Use for the deployment all the well-know best practices for a resilient system;

- Use, as API Gateway, a software that is proved to be efficient against external attack (e.g. DDoS) and can handle faults.

5. SYSTEM REQUIREMENTS

After having described in detail the FrailSafe system architecture and its functional requirements, we shift our focus on the system requirements that the whole FrailSafe platform must respect. We can group them in three different types:

- Network availability: the services must be reached from the network, i.e. network issues must be avoided or recovered in order to minimize the time that services are unreachable;

- Hardware reliability: the hardware where services will be implemented and executed must be reliable and mechanisms to recover any hardware failure must be used to minimize service availability issues due to hardware failure;

- Data Loss prevention: since the core of all the FrailSafe services is data collection and elaboration, this requirement is crucial. Data loss is generally due to hardware failure (i.e. its prevention is a subcase of hardware reliability) or to a problem in writing data to a hard drive, and can be avoided using data replication mechanisms such as RAID and/or journaled file systems (e.g. EXT3 file system or NTFS).

Starting from these basic requirements, a solution for implementing this cloud platform might be to use an existing cloud service provider to deploy all the services. In this way we can completely demand some of the above aspects such as network availability and hardware fault-tolerance because they will be addressed directly from the service providers. Moreover, we can exploit the functionalities offered by the service provider in order to achieve a high scalability and flexibility of the whole platform without any effort in terms of maintenance and management.

Conversely, from a FrailSafe service point of view, each block of the architecture has to take into account several aspects in order to assure a constant service delivery, for example:

- Mechanism to obtain a high service availability, e.g. 99,999% (five nines);

- A robust software fault-tolerance mechanism, e.g. with more instances of the same software or with specific techniques to recover a faulty software instance without interrupting the service;

- Mechanisms to assure data integrity and prevent data corruption;

- Mechanisms to avoid bottlenecks that will result in service interruption/long delay.

Each block has to satisfy these requirements, since the FrailSafe cloud is composed by a lot of modules that are strictly dependent from each other and a fault in one module could cause a performance decay / a service fault in the entire ecosystem.

Some other system and non-functional requirements are described more in details in the following sections, while a summary of functional requirements can be found in the Annex 6.2.

5.1. Security

As described above, the FrailSafe platform will be deployed in a cloud infrastructure and will be accessible from participants, clinicians and other stakeholders. As a consequence of that, security of the entire system must be guaranteed.

With this respect, we must ensure that:

- Access will be guaranteed only to authenticated actors;
- Access to each platform resource will be granted only to authorized actors;
- Communications between actors and platform will be protected against cyberattacks.

As we have seen before, the FrailSafe platform has been architected to satisfy all these requirements, namely by the insertion of an Identity and Access Management module (see 3.3.3.7) and by securing all the communication that comes from the outside of the platform using a secure protocol such as HTTPS.

Security has also to be guaranteed by external attacks that could be made by malicious users.

The system must be able to effectively contrast these attacks, that could be divided in three categories:

- Attacks to disrupt the service, also known as Denial of Service (DoS);
- Impersonating another user and try to get his/her data (e.g. Man-In-The-Middle attack);
- Stole users' data by entering in the system in an unauthorized manner (e.g. security breaches).

A well-architected system is not enough to prevent this kind of attack, hence all the best practices for software development and deployment have to be implemented.

Additional security measures include:

- **Secure access** – allow secure HTTP access (HTTPS) so that we can establish secure communication sessions with services using SSL;

- **Encrypted data storage** –the data and objects are stored in databases encrypted automatically using well-known encryption standards, such as possibly Advanced Encryption Standard (AES) 256;

- **Security logs** – System provides logs of all user activities.

5.2. Privacy

Data privacy is another important requirement that a system like FrailSafe must implement also in its early design stage. Since in FrailSafe we have to deal with sensitive and personal data, we must respect all the regulations that are in force in the European Community.

About this latter point, the new General Data Protection Regulation (GDPR), put forth by the European Commission in 2012 and finally generally agreed upon by the European Parliament and Council in December 2015, is set to replace the previous Data Protection Directive 95/46/ec. This new regulation contains a number of new protections for EU data subjects and threatens significant fines and penalties for non-compliant data controllers and processors once it comes into force in the spring of 2018.

Better data protection rules mean you can be more confident about how your personal data is treated, particularly online. The protection rules put citizens back in control of their data, notably through:

- **The right to be forgotten:** When you no longer want your data to be processed and there are no legitimate grounds for retaining it, the data will be deleted. This is about empowering individuals, not about erasing past events or restricting freedom of the press;
- **Easier access owned data:** A right to data portability will make it easier for you to transfer your personal data between service providers;

- **Putting citizen in control:** When citizen's consent is required to process your data, you must be asked to give it explicitly. It cannot be assumed. Saying nothing is not the same thing as saying yes. Businesses and organisations will also need to inform you without undue delay about data breaches that could adversely affect you;
- **Data protection first, not an afterthought:** 'Privacy by design' and 'privacy by default' will also become essential principles in EU data protection rules – this means that data protection safeguards should be built into products and services from the earliest stage of development, and that privacy-friendly default settings should be the norm – for example on social networks;
- **Pseudonymous data:** The GDPR defines pseudonymization as "the processing of personal data in such a way that the data can no longer be attributed to a specific data subject without the use of additional information." To pseudonymize a data set, the "additional information" has to be "kept separately and subject to technical and organizational measures to ensure non-attribution to an identified or identifiable person." In sum, it is a privacy-enhancing technique where directly identifying data is held separately and securely from processed data to ensure non-attribution.

The aim of the European Data Protection Regulation is to harmonise the current data protection in place across the EU member states. The fact that it is a "regulation" instead of a "directive" means it will be directly applicable to all EU member states without a need for national implementing legislation. We will follow this process implementing the new rules, as they will be approved.

For the FrailSafe platform we have implemented the "Privacy by design" principle that has been introduced by the GDPR, and hence all the architecture and the deployment strategy has taken care about the privacy issues since the beginning of the design steps.

It is also important to note that what is defined here represents also the guidelines that must be reflected in the design that is made of the different modules of the ecosystem.

Privacy is also strictly related to the previous Security issue about data protection and pseudonymous, the aim is to protect the subject and not only the data, hence one solution to guarantee privacy is to make the data useless.

To implement this latter point, we can think to protect data with encryption, so that data remains illegible and virtually useless if and when it falls into the wrong hands. A default strategy of 'encrypt everything' will be critical to ensuring compliance, and today's advances in technology mean that encryption is no longer an expensive and heavy process – in fact, it is now faster and easier than ever before to secure data with encryption.

That said, while this is a good start, effective protection must go beyond encryption – particularly when considering the threat from within. Access controls are an important extra layer of defence, ensuring that only users with the appropriate level of authorisation can access certain data sets. Once a user is granted access to an encryption key, their usage is fully controlled and accounted for, enforcing rules on data entitlements, preventing the sharing of access with another user as well as other factors such as time of day.

5.3. Reliability and Availability

The importance of the privacy and security issues are, in some way, linked to another important requirement: reliability. This refers to the ability of the system to offer reliable services and a reliable data storage system. We must architect a system that is able to store the data in a way that we can overcome hardware failures without losing data. To achieve this, we will apply some of the well-known practices of data redundancy by replicating data among different hardware storage, that are often offered by different cloud providers.

To build a reliable system, we have also to consider the availability of the system, since users must be able to access FrailSafe services without experimenting service interruption.

Having a high available system often implies to put in power a complex infrastructure, that is also distributed among different places, to ensure that some service interruption in a zone can be recovered by routing requests to other available zones. Cloud computing has facilitated in achieving this target by allowing us to spread and replicate our services in different geographically-distributed zones. In this way we are able to achieve very high service level and availability.

5.4. Scalability

Moving towards availability and reliability, scalability is another aspect that in the system requirements needs to be considered.

Scalability is the characteristic of the system to grow its resources to manage an increasing workload.

Scaling a system is the operation that allows to have the same level of service, that the users expect, when are necessary more resources to satisfy a growing demand due to an increase number of users or sessions or transactions, etc.

The challenge, when a system architecture is being designed, is to identify the points where it is needed a flexible and scalable solution to avoid fails or malfunctioning. For this reason, it is necessary to pay attention to every aspect of the system design and in every phase of the development up to tests.

Currently, the cloud providers offer the opportunity to scale computing resources when it is necessary. This is an example of flexible solution where the required resources are increased or decreased in relation to the real requirements. The advantage is the avoidance of a wastefulness of computational capacity and money. This characteristic is also called “Elasticity” where the system fits the resources needed to cope with loads in a dynamic manner, so that when the load increases, the system adds more resources and when demand wanes it shrinks back and removes unneeded resources.

5.5. Usability

Usability represents a set of requirements obtained from a preliminary research, using task analysis, surveys, interviews, observations and so on to help the development team to focus on the goals of the design. These requirements should be also the starting point during the quality testing, to verify that the system features and functionalities have been designed according to the user’s needs and preferences.

The parts with which the users interact, such as interfaces, processes, etc., should be perceived easy to learn and also easy to remember especially in an infrequent use of the system. Its use should be simple the first time and all the other times without having to consult a guide or to ask instructions.

The main part of the requirements of a system is represented by the functional ones that have as subjects: input, processing, and output. These requirements don’t indicate how the system is easy to use, for this reason usability is a non-functional requirement because it doesn’t concern a specify part of the system functionality but how this is perceived by the user.

It is important that these usability requirements can be verified to be sure that the system gets the grade of usability that we have been designing. Achieving this aim is difficult in practice and the approaches to cover all parts of it are not a lot.

We can consider five usability factors:

1. **Ease of learning**. Novices and expert users have to learn to use the system easily;
2. **Task efficiency**. The system has to be efficient for the frequent user;
3. **Ease of remembering**. After a no-use period, the user has to remember to use the system without guide or instructions;
4. **Understandability**: During the use of any function the user has to perceive what the system does;
5. **Subjective satisfaction**: The user has to feel satisfied with the system.

Developers should measure these factors and specify the necessary level for each factor that it is need to reach in the different aspect of the system to guarantee a satisfying grade of usability.

5.6. Data Exchange protocol

As said in previous chapters, the FrailSafe platform structure can be considered as a Service Oriented Architecture (SOA), where each module is seen as a service that could be consumed by others.

Exchanging data across digital networks requires that each module speaks the same "language" and the communication between them must be clear and secure for both.

To accomplish this, every service has to use common protocols which define:

- How one module can request data from another module;
- Which specific parameters are needed in the data request;
- What would be the structure of the result of the request;
- What error messages to display when a certain rule for communication is not observed, to make troubleshooting easier.

The two main protocols used for data exchange are SOAP (Simple Object Access Protocol) and REST (Representational state transfer). Successively we analyse the main pros and cons between them.

Advantages of SOAP compared to REST:

- Protocol transport independent (REST requires HTTP);
- Good in distributed environments (REST presumes communication point-to-point);
- High grade of standardisation;
- Pre-build extensibility in the form of the WS standards;
- Built-in error management;
- Automation when used with some language products;

On the other side, advantages of REST compared to SOAP:

- To interact with the Web service are not necessary expensive tools;
- Easy to learn;
- More efficient (SOAP uses XML for all messages, REST can use smaller message formats);
- Fast (no extensive processing required);
- Considering the current design philosophy, it is closer to the web technologies.

About the data exchange, the two main formats used are XML (Extensible Markup Language) and JSON (JavaScript Object Notation).

With its familiar markup language, very similar in aspect to HTML, XML has been a common way to structure data. In the communication between systems, this format uses a set of standards for data exchange and the structure of the data is parsable in an iterative way. In

XML the main advantage is its structure. It is possible to add or remove data in the result structure, but the standard and predictable format allows keeping the previous parsing process without loss of information when the data changes over time. On the other side, the drawback of XML is its size, because data structure contains many characters used to define the data format.

JSON is derived from JavaScript language and it can be considered the most common format for the data exchange between systems. Its structure focuses more on content information and less on formatting. Data is represented as key-value pairs and for this reason one its main advantage is the lower size with respect to XML. This is very important when it is necessary to keep data interchange packets as compact as possible.

About the communication, HTTP (HyperText Transfer Protocol) is the protocol for transmitting and receiving information between the server and the client and its task is to manage the mutual understanding between the modules to exchange information or data in a proper way.

In SOA, every communication between the modules is based on this protocol the only difference can be if it needs to be more secure or not.

When the services are directly exposure to the internet, it is necessary to pay serious attention to the communication between requester and provider to avoid any interception by others, the so called man-in-the-middle attack, and the solution widely used is the Secure HTTP (HTTPS).

Over a secure SSL connection, the requests and the responses between server and client are encrypted and decrypted; in few words, once the connection is established, the two subjects use an established algorithm and keys to send messages to each other in a secure way.

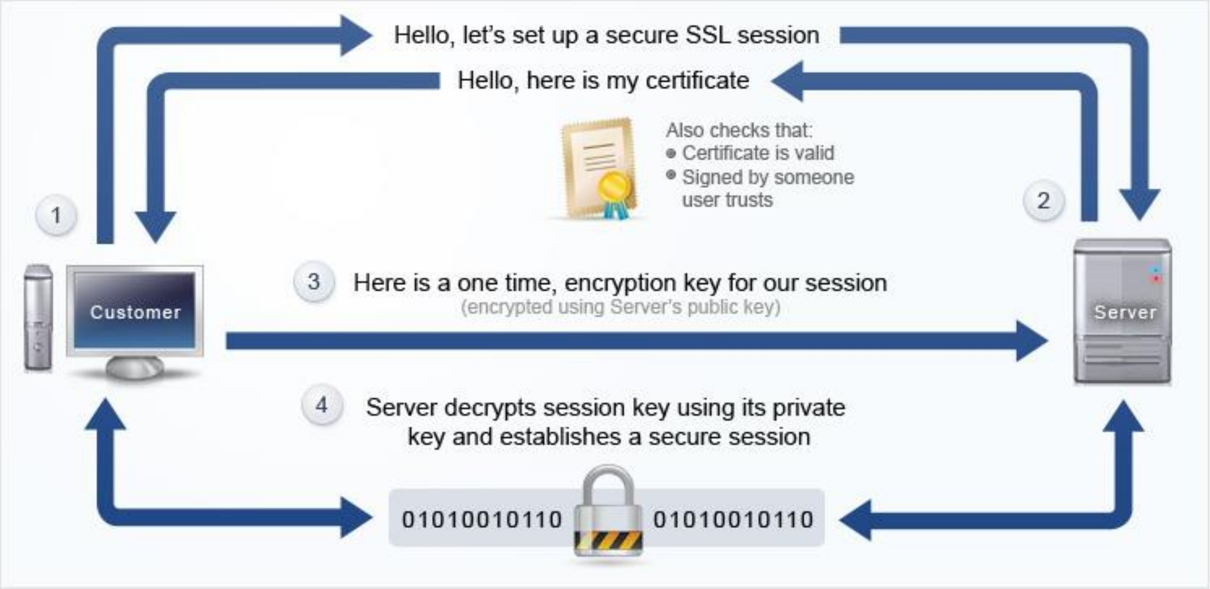

Hello, let's set up a secure SSL session
Hello, here is my certificate
Also checks that:
Certificate is valid
Signed by someone user trusts
1
2
Customer
Server
3
Here is a one time, encryption key for our session
(encrypted using Server's public key)
4
Server decrypts session key using its private key and establishes a secure session
01010010110
01010010110

Figure 46 - Secure SSL Connection

6. ANNEXES

6.1. Platform Implementation

6.1.3. FrailSafe infrastructure

The FrailSafe platform plays a central role in the project, since it serves as ICT support to the study, the detection, and the prevention of the frailty condition. The primary purposes of the platform are to:

- Reliably collect and store data coming from clinicians, sensors, patients;
- Provide a secure way to host and handle that data;
- Host the FrailSafe data analysis services pertaining to the above data in order to provide frailty-related metrics;
- Host the FrailSafe services exposed to users and/or clinicians, for example the FrailSafe Virtual Community Platform and the Intervention System;
- Host the services regarding FrailSafe applications and Games;
- Be suitable for performing those exploitation initiatives[9] identified for FrailSafe's results after the end of the project.

To meet all these requirements, a well-integrated infrastructure has to be designed.

The most common ways to implement this kind of infrastructure are:

- On-premise solution: host the modules and infrastructure in a private (owned or rented) data center;
- Distributed: allows the components to be distributed among partners' datacenters, and communicate over the Internet;
- On public cloud: rely on the infrastructure and services offered by a third party provider for hosting the modules;
- Hybrid: use a combination of the aforementioned methods.

6.1.4. FrailSafe platform requirements

Making effective decisions during the FrailSafe platform implementation stage presumes a better understanding of the requirements of the FrailSafe platform. These requirements are derived by considering different points of view given the multitude of the platform users. Since alternative strategies present strengths and weaknesses for the different user groups, a selection procedure must be employed in order to find the implementation strategy which best fits the project needs.

6.1.4.1 Security and Privacy

The FrailSafe project heavily relies on sensitive data, specifically on personal medical data. Hence, security and privacy issues are of paramount importance and must be carefully considered and faced. This is especially true in view of the recent hardening of the relevant European Regulations on the subject of data protection and the latter has become very precise about personal data acquisition and handling (i.e. the Data Protection Directive - 95/46/EC - and the new General Data Protection Regulation – EU GDPR n. 2016/679).

[9] See WP8 work.

These security considerations also cover the physical aspect of data security. A major issue of physical security is the guarantees regarding the status of the data host. In the FrailSafe project, these matters are carefully taken into account across the different Work Packages and their respective Tasks. During the platform design phase the security and privacy factors were given the proper treatment[10].

6.1.4.2 Reliability of the system

A point relevant to security is system reliability. The latter is defined as the system ability to offer its predesignated service level by means of software and service availability as well as data persistency. This concept is important for FrailSafe, since the system must be highly available while guaranteeing at the same time that user generated data (i.e. patients and clinicians) will remain intact despite random and occasional hardware or software failures.

6.1.4.3 Scalability

During the design stage of a complex system developed under a research project, final product scalability is a critical factor. Since the test cases outlined in the project agreement cover a limited number of platform users, a reasonable amount of computational and storage power is required at this point in order to handle the available data volume. However, in order to completely exploit the project results the system should easily scale up to handle a larger amount of users and data. This is a crucial feature that must be considered in the design and the development process of the platform.

Other points to be considered are the cost-effectiveness solution tradeoff and the solution feasibility given the FrailSafe consortium’s expertise and facilities.

6.1.5. FrailSafe platform deploy strategy

After considering the above matters, we can conclude that the best strategy to implement the FrailSafe platform is to use services offered by public cloud providers. The reasons behind this choice are listed below.

It is clearly evident that the complexity of the FrailSafe platform suggests resorting to a modular architecture where each module is dedicated to a single task and interacts with the other modules by means of a specific communication protocol. This architectural concept is broadly known in the scientific literature as Service Oriented Architecture (SOA) and is widely used in the design of highly integrated yet flexible platforms. Moreover, SOA can be also applied seamlessly to a distributed computational environment for production or research purposes. Besides these advantages, the SOA paradigm covers the requirements for security and reliability as will be explained below. The SOA paradigm is also employed to achieve high system reliability of the system, which is, as stated earlier, a key FrailSafe requirement.

As SOA is an architectural paradigm, it can be implemented in a number of ways depending on the application factors. As mentioned earlier, within the FrailSafe context security plays a key role. Under this point of view, the nature of distributed environments presents certain challenges, since security should be also distributed to each deployment point. Hence, if not treated appropriately, multiple points of attack and multiple points of vulnerability may exist.

[10] See mainly D9.6, “Ethics, Safety and mHealth Barriers (regulation, legislation, etc.) Manual (vers. a)”.

Moreover, controlling and maintaining a distributed system is not trivial because of the potentially different security schemas and policies applied at each data center. Another important related factor which can potentially weigh against a distributed deployment is that, in order to guarantee full compliance to the EU Regulations, several different data centers must be checked. This might be a problem that can have a severe impact on the success of the project, if not handled properly.

At this point, the On-Premise strategy might seem as the best alternative for implementing the FrailSafe platform. However, this is not completely true. Certainly, the On-Premise solution guarantees complete control, both physical and logical, of the both the infrastructure and the underlying platform modules. On the other hand, though, the FrailSafe requirements clearly dictate the need for a very robust end effective infrastructure allowing a highly available and reliable system with the possibility of easily scaling up. Note that these kinds of requirements are not commonly found in SMEs or research entities ICT infrastructures, because in general such data centers require huge investments. On the contrary, big enterprises that already deliver high scale products in the market can and do afford such data centers.

For these reasons, it has been decided to use services offered by a public cloud provider. In this way every requirement regarding the FrailSafe platform is fulfilled without the need and the related costs to maintain a complex data center infrastructure, since a general cloud provider adopts a “pay-as-you-go” charging policy. This paradigm consists of billing the customer only for the time that a service is used; in this way the advantages of getting rid of the fixed cost, typical of the On-Premise strategy, can be exploited. The only costs incurring from the distributed policy will be those linked with the usage of the infrastructure.

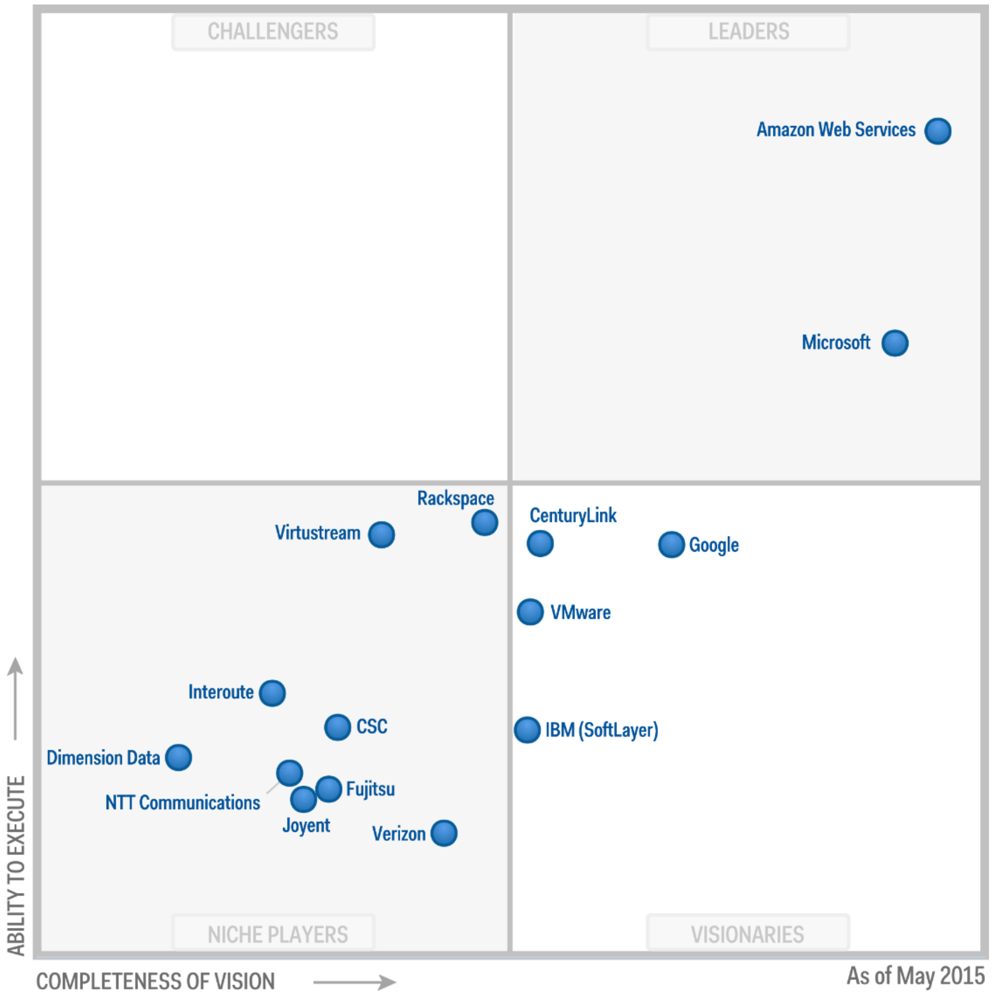
6.1.6. Public cloud Provider Offerings

During the very recent years, the importance of cloud computing in enterprise settings, including business logic and operations, has significantly increased. Right now a huge offer of various cloud services is available in the market. Before the analysis of cloud service providers, it must be highlighted that, for the project purposes, a provider offering cloud services by means of Infrastructure as a Service (IaaS) is required. This is a very important point because the implementation team should be able to use the infrastructure of the provider in addition to its services (e.g. the storage). With respect to this, the major players in the IaaS market are:

- Microsoft Azure
- Google
- Amazon Web Services
- IBM
- SalesForce
- Rackspace
- Oracle

The above listed providers combined account for about the 70% of the market share. Additionally, many of the providers of the remaining 30% are in a way or in another associated with these major players, often acting as re-sellers.

In this list we can also find a small set of providers that are widely used by enterprises and public sectors entities, mainly because they have the most advanced services offered and they are the most cost-effective solution in the market. This set is composed by AWS, Microsoft and Google, as shown by the 2015 Gartner Magic Quadrant Report for IaaS providers showed below.

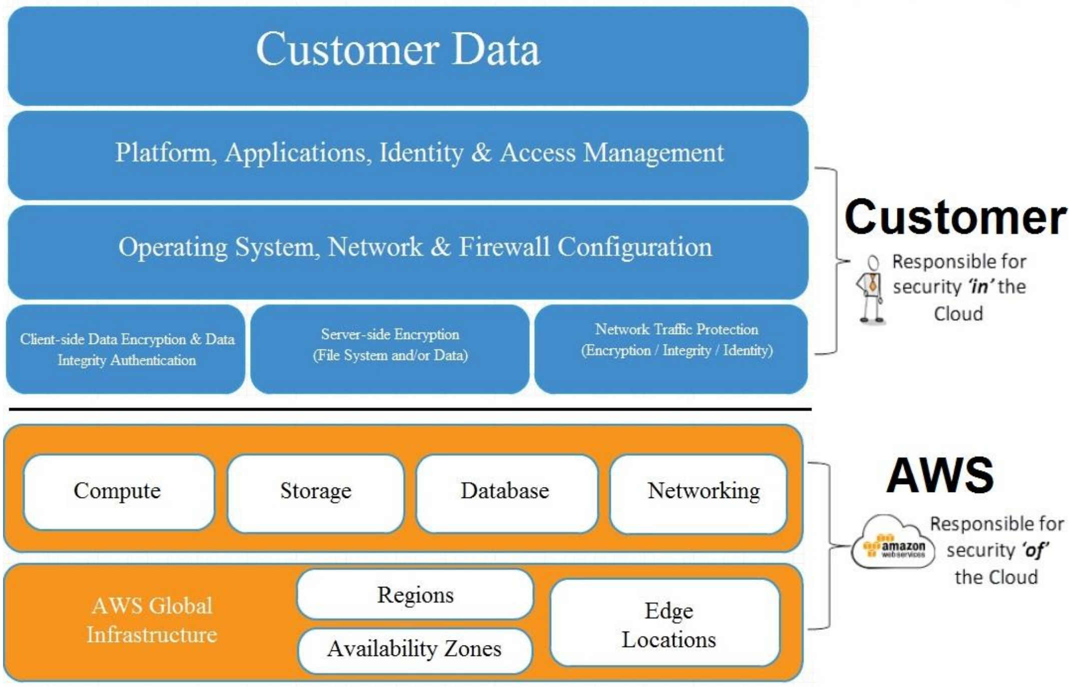


Gartner Magic Quadrant Report for IaaS providers

For the FrailSafe platform implementation, the services offered by Amazon Web Services (AWS) were initially selected. This choice was driven by several reasons: first and foremost, AWS is widely recognized as the leader in the IaaS market. Moreover, Gruppo SIGLA, which is responsible for the integration and hence for the management of the cloud and its deployment within the project consortium, has extensive experience with AWS which can be used in FrailSafe to speed up and rationalize all the processes of the platform design, implementation, and deployment. Gruppo SIGLA is also the integration leader in a FP7 project, called PEGASO (http://www.pegasof4f.eu/), that is using AWS to implement its platform, so also the experience in this research project could be used to deliver better results and also as a valuable matter of comparison.

Another factor weighing in favor of AWS is that it complied to the privacy and security directive in effect at the time of the start of the project – i.e. January 2016. The full list of certifications and compliances can be found at https://aws.amazon.com/compliance/. AWS has a specific model for handling the security and privacy matters called “Shared responsibility model” (https://aws.amazon.com/compliance/shared-responsibility-model). This model is used to find which the user responsibilities are, i.e. the AWS customer, regarding privacy and security of data access and transmission and which are the corresponding AWS responsibilities. In conjunction with the AWS compliances and the “Shared responsibility model”, a system that fully responds to the EU Directives can be designed and developed. Last but not least, Amazon has two data centers in Europe, in Ireland and Germany.

Therefore, any data generated by the project is guaranteed to be stored in these locations and it will not be moved from there for any reason whatsoever[11].


Customer Data
Platform, Applications, Identity & Access Management
Operating System, Network & Firewall Configuration
Client-side Data Encryption & Data Integrity Authentication
Server-side Encryption (File System and/or Data)
Network Traffic Protection (Encryption / Integrity / Identity)
Customer
Responsible for security 'in' the Cloud
Compute
Storage
Database
Networking
AWS Global Infrastructure
Regions
Availability Zones
Edge Locations
AWS
amazon web services
Responsible for security 'of' the Cloud

AWS Shared Responsibility Model

6.1.6.1. State of art

The scenario generating the conditions for the initial selection, changed at month 5 of the project timeline after the adoption of the new General Data Protection Regulation (EU GDPR n. 2016/679) on the 27th of April 2016. The choice of the cloud provider, that was initially AWS as mentioned above, has been the subject matter of discussion with the FrailSafe consortium, the Project Officer, representatives from other H2020 eHealth projects – i.e. in particular PreventIT but also iPrognosis and City4Age - and finally it could somehow depend on the suggestions of the European Commission officers working on it.

It is in the interest of the FrailSafe consortium to identify a cloud provider that can fulfill all the rules that the Regulation needs. For this reason, at present, the consultancy of some legal experts in privacy matters has been also required, to address the issues and the choice of the right cloud provider. The most likely alternative suggested by these experts, in case AWS will finally have to be excluded by the CE for FrailSafe or not approved in a time useful for the correct progress of the project, is to adopt the services of one of the CISPE[12] cloud providers.

In conclusion, it is important to underline that the selection of a cloud provider compliant with the new data protection regulation has the principle priority in order to have at the end of the project a product ready for market even going to the detriment of performance of the whole system.

[11] See "Is data stored in AWS replicated, and does it ever leave the EU region?" at https://aws.amazon.com/compliance/eu-data-protection/?nc1=h_ls

[12] https://cispe.cloud/

6.2. Functional Requirements

A set of functional requirements have been identified and defined for the FrailSafe platform and all the features implemented within it.

They are described here below, with a priority range defined from 1 (lowest) to 5 (highest).

Requirement ID	FUNCTIONAL-001
Name	WWS (Wearable wellness system)
Type	Functional
Description	The system should be able to record measurements taken through the WWS. These measurements include heart rate signals, respiratory signals, posture information, active time, etc.
Rationale	The information gathered by the wearable vest is among the core information on which the FrailSafe project is based upon. The vest can be worn by the older person at large time periods, allowing for the collection of large amounts of frailty-related data.
Fit criterion	The accurate and real-time recording of heart rate signals, respiratory signals, posture information, active time, etc. by the system.
Priority	3
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-002
Name	Indoor activity monitoring
Type	Functional
Description	The system should be able to monitor the indoor location of the user and the time he/she spends at the different rooms and positions.
Rationale	The indoor activity records of the user will be used to investigate common patterns of movement and social behaviours. Moreover, indoor position might be useful in case of alerts.
Fit criterion	The indoor position of the user is recorded with 0.5 meters’ accuracy.
Priority	3
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-003
Name	Indoor activity monitoring devices
Type	Functional
Description	The indoor activity monitoring devices (e.g. beacons) should be placed to positions around the user in a way that they surround him/her, and not to close distances between them, at the same height (~1.5m) and if it is possible there should not exist obstacles in front of them.
Rationale	In order to know user's position to the maximum extent possible it is important to place beacons in the right way.
Fit criterion	The ability of the clinician to place beacons according to instructions.
Priority	5
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-004
Name	Outdoor activity monitoring
Type	Functional
Description	The system should be able to monitor the outdoor location of the user and the time he/she spends at the different locations.
Rationale	GPS and step counter records will be used to track and log the position of the person while outdoors, to investigate how outgoing and social is. Moreover, GPS position might be useful in case of alerts.
Fit criterion	The outdoor position of the user is recorded with 2 meters' accuracy.
Priority	3
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-005
Name	Activity monitoring auto-upload
Type	Functional

Description	The measurements of indoor and outdoor activity should be sent automatically in hourly intervals to auto-send targets (FrailSafe FTP server etc.), if an Internet connection is available, or once an Internet connection becomes available.
Rationale	It is important to update the server in order to have data for each user available for further examination from researchers and doctors.
Fit criterion	The collected data are uploaded to the database at most a week after they were collected.
Priority	3
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-006
Name	Auxiliary devices usage
Type	Functional
Description	The system should allow the collection of measurements from auxiliary devices, i.e. the FORA blood pressure monitor and scales, the dynamometer and the Mobil-o-graph.
Rationale	Blood pressure, weight, strength and arterial stiffness are important clinical parameters related to frailty. The respective measurements are expected to provide useful information for the project goals.
Fit criterion	The ability of the older person and the clinician to collect blood pressure, weight, strength and arterial stiffness measurements using the FrailSafe auxiliary devices.
Priority	5
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-007
Name	FORA auxiliary devices unique usage
Type	Functional
Description	Only one auxiliary FORA device of a specific kind (blood pressure monitor or scales) should be linked to a user at a time. When adding a new user, the previous device should be unlinked from the previous user in order to be linked to the new user.

Rationale	In order to have exact measurements for each user, each auxiliary device must be linked to one user each time.
Fit criterion	The ability of the clinician to link one specific FORA device to each user and update the FORA telehealth system for any changes.
Priority	5
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-008
Name	FORA telehealth system, Blood Pressure Tool
Type	Functional
Description	The clinician should be able to link the blood pressure and scales devices with the phone via Bluetooth and then upload the measurements from the phone to the FORA website to the user's unique profile.
Rationale	It is important to have FORA telehealth system updated for every user in order to update the FrailSafe database with up-to-date measurements.
Fit criterion	The ability of the clinician to use FORA telehealth system according to given instructions.
Priority	5
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-009
Name	FrailSafe physical games
Type	Functional
Description	The system should allow the user to play games involving physical activity, such as grip strength and upper body movement.
Rationale	It is important to motivate the older persons to use the various FrailSafe devices and sensors, both for data collection purposes and for rehabilitation purposes.
Fit criterion	The older person is able to play games involving physical activity.
Priority	3

Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-010
Name	FrailSafe cognitive games
Type	Functional
Description	The system should allow the older persons to play games involving cognitive activity. The FrailSafe cognitive games should be able to utilize the collected data and estimate the user's cognitive condition with relevant precision.
Rationale	It is important to develop the suitable algorithmic approaches which estimate user's cognitive precision with accuracy. The ability of algorithmic approaches to estimate user's cognitive condition is a necessary factor for doctors and researchers.
Fit criterion	The computed scores of cognitive condition agree with cognitive diagnostic tools.
Priority	3
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-011
Name	FrailSafe game data update
Type	Functional
Description	FrailSafe database should be updated for each user every time he/she plays a game.
Rationale	It is important to have data for each user regarding scores of physical/cognitive abilities to estimate their physical/cognitive condition and recommend personalized game features (tasks, levels, difficulty, etc.).
Fit criterion	The ability of FrailSafe game to connect and update the database regularly.
Priority	3
Conflicts/Relations	None.
Author	CERTH

Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-012
Name	FrailSafe game feedback
Type	Functional
Description	The FrailSafe games should provide personalized feedback to user each time he/she plays the game including (tasks, levels, difficulty, etc.)
Rationale	It is important to provide the right feedback to user in order to understand better his/her physical/cognitive condition and examine his/her progress.
Fit criterion	The ability of FrailSafe game to provide the right feedback to each user.
Priority	3
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-013
Name	Customer Service
Type	Functional
Description	The system should send alerts to the customer service, in order to inform about adverse events. The system should support the following types of events: • Falls • Loss of balance • Loss of orientation • Heart or respiration problems
Rationale	It is important for Customer service to know as soon as possible when an adverse event happens in order to act according to the situation immediately.
Fit criterion	A generated alert is sent within 10 seconds from the time that the event happened. The false positive rate should be at most 10%. The false negative rate should be at most 1%.
Priority	5
Conflicts/Relations	None.
Author	CERTH

Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-014
Name	Patient-oriented information visualization
Type	Functional
Description	The system should allow the older person to view his/her personal data collected so far through a visual interface utilizing charts, graphs, etc.
Rationale	It is important that the older person views the information that has been collected so far, in order to have an overview of his/her progress. Visualization tools allow for a comprehensive and intuitive data presentation.
Fit criterion	The FrailSafe mobile front-end supports patient-oriented visualizations of the data of a single individual.
Priority	5
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-015
Name	Clinician-oriented information visualization
Type	Functional
Description	The system should allow the clinician to view the collected so far for one or more older persons through a visual interface utilizing charts, graphs, as well as advanced visual analytics methods.
Rationale	It is important that the clinical staff has an overview of the progress of an individual or of a group of participants, so that he/she can make decisions and provide recommendations. Visual analytics tools allow for an intuitive and comprehensive presentation and exploration of the available data.
Fit criterion	The FrailSafe mobile and web front-end supports clinician-oriented visualizations of the data of a single or of many individuals.
Priority	5
Conflicts/Relations	None.
Author	CERTH
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-016
Name	Clinical Web Platform – eCRF features
Type	Functional
Description	The system should allow the clinical staff to implement the data medical assessment protocol for what concerns the management of participants, their being part of a specific group, the execution of the visits, the correct performance of the data collection in questionnaire and file formats, the ability to visualize the state of a specific visit and the values of the data already collected.
Rationale	It is important that the clinical staff has an overview of the progress of the medical assessment for each participant, in order to complete the clinical data collection.
Fit criterion	The FrailSafe Clinical Web Portal implements the eCRF functionalities that support the performance of the medical assessments and the related data collections.
Priority	5
Conflicts/Relations	None.
Author	SIGLA
Revision	Initial version v1.0

Requirement ID	FUNCTIONAL-017
Name	Identity and Security features
Type	Functional
Description	The system should guarantee the secure storage of platform's users data, the means of management of the different roles and the individual users of the system, to secure the access to the system resources and, in general, to provide the means to authorize a user in ownership of a correct set of credentials to log in the platform and its resources.
Rationale	It is important that users could be: • Created and Deleted; • Once created and/or present into the system, Read/Update their set of profile data; • Once created and/or present into the system, obtaining a set of credentials for accessing system resources; • Once created and/or present into the system, being associated to one - or more - roles for allowing them access only to the minimal set of system resource they require;
Fit criterion	The FrailSafe Identity and Security features are implemented in a way proper to gurantee the correct users management and access to system resources in full respect of the required security and privacy level.
Priority	5
Conflicts/Relations	None.

Author	SIGLA
Revision	Initial version v1.0