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

Deliverable No: D4.2

Offline analysis of data (vers. b)

Due date of deliverable:	M24 (31/12/2017)
Actual submission date:	31/12/2017
Version:	1.0
Lead Author:	Konstantinos Deltouzos (UoP)
Lead partners:	Konstantinos Deltouzos, Evangelia Zacharaki, Emilia Papagiannaki, Spyros Kalogiannis, Thomas Papastergiou, Dimitrios Vlachakis, Vasilis Megalooikonomou (UoP), Ilias Kalamaras (CERTH), Carlo Mancuso, Tommaso Faetti, Gianluca De Toma (Smartex)

Horizon 2020
European Union funding
for Research & Innovation

Change History

Ver.	Date	Status	Author (Beneficiary)	Description
0.1	10/11/2017	draft	Konstantinos Deltouzos,Emilia Papagiannaki,Evangelia Zacharaki, Spyros Kalogiannis, Dimitrios Vlachakis, Vasilis Megalooikonomou (UoP), Ilias Kalamaras (CERTH), Carlo Mancuso, Tommaso Faetti, Gianluca De Toma (Smartex)	First draft based on the preliminary version of the deliverable (D4.1)
0.2	24/11/2017	draft	Aimilia Papagiannaki (UoP)	Enrichment of Group-wise histogram analysis with additional data
0.3	01/12/2017	draft	Spyros Kalogiannis, Konstantinos Deltouzos (UoP)	Revision of Data Management Section
0.4	05/12/2017	draft	Evangelia Zacharaki, Konstantinos Deltouzos (UoP)	Second draft prepared reviewing the incorporated changes
0.5	15/12/2017	draft	Thomas Papastergiou	Enrichment of the data preprocessing section
0.6	22/12/2017	draft	Vasilis Megalooikonomou (UoP)	Sixth draft after internal review (UoP)
0.7	24/12/2017		Evangelia Zacharaki (UoP)	Seventh draft after internal review (UoP)
0.8	27/12/2017	draft	Ilias Kalamaras (CERTH), Carlo Mancuso, Tommaso Faetti, Gianluca De Toma (Smartex)	Eigth draft after internal review (CERTH, Smartex)
1.0	31/12/2017	final	Konstantinos Deltouzos (UoP)	Final version from deliverable responsible

EXECUTIVE SUMMARY

The FrailSafe project aims to study all domains of frailty and to create new measures of assessments leading to a model which will be able to better understand, detect, predict, delay, or even revert frailty. To achieve these aims plans are made to devise a comprehensive clinical assessment, to develop a real-life sensing and intervention platform, and to provide a digital patient model of frailty, sensitive to dynamic parameters. Recommendations will be provided to delay frailty, and all this through a safe, unobtrusive, acceptable system and cost-effective system.

The aim of work package WP4 is to develop methods for the offline and online management, fusion, and analysis of multimodal and advanced technology data from social, behavioral, cognitive, and physical activities of frail older people and apply them to manage and analyze new data. Results from the analysis of existing and new data will be also used to create user-profiling virtual models of elderly patients.

The main focus of the deliverable D4.1 is to report on the usage of existing and new developed techniques within the FrailSafe project towards offline data management, preprocessing and analysis. In particular, techniques for data pre-processing are examined involving data cleaning (handling of missing, noisy or inconsistent data, identification and/or removal of outliers), dealing with contaminated/noisy data segments, data integration, data transformation (normalization and aggregation), data reduction (production of reduced representations of data using dimensionality reduction (feature selection), discretization and numerosity reduction techniques. New techniques for data reduction and summarization of streaming sensor data are also been developed, in order to explore meaningful measuring units for frailty.

Additionally, the state-of-the-art of existing database technologies was examined in order to support the organization of data (both raw signals and analyzed data), including the support of efficient storage and retrieval capabilities such as multidimensional indexing and content-based queries. An investigation of data compression issues and of the ability to analyze compressed data was performed, with respect to the most representative frailty features acquired from feature selection.

Finally, we present our work in the detection of patterns and associations between clinical indicators and frailty states, and in the analysis of multidimensional time series towards reveling associations among signals and symptoms that are connected to the frailty syndrome.

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:	4.2	**Title:**	Offline analysis of data (vers. b)
Work package number:	4	**Title:**	Data Management and Analytics

Date of delivery	**Contractual**	**31/12/2017 (M24)**	**Actual**	**31/12/2017**
Status	Draft ☐		Final ☒	
Nature	Report ☒ Demonstrator ☒ Other ☐			
Dissemination Level	Public☐ Consortium ☒			
Abstract (for dissemination)	The main focus of the deliverable D4.2 is to report on the usage of existing and new developed techniques towards offline data management, preprocessing and analysis.			
Keywords	Data Management, Data Preprocessing, Offline Data Analysis, Multivariate statistical analysis, Activity classification, Multi-dimensional time series analysis			

Contributing authors (beneficiaries)	Konstantinos Deltouzos, Dimitrios Vlachakis, Evangelia Zacharaki, Emilia Papagiannaki, Spyros Kalogiannis, Thomas Papastergiou, Vasileios Megalooikonomou (UoP), Ilias Kalamaras (CERTH), Carlo Mancuso, Tommaso Faetti, Gianluca De Toma (Smartex)			
Responsible author(s)	Konstantinos Deltouzos		**Email**	deltouzos@upatras.gr
	Beneficiary	UoP	**Phone**	+30 2610 996 994

Table of contents

List of figures

List of Tables

1 Introduction

Managing FrailSafe's multimodal data is a task of great importance. The large data files that contain the raw sensor data generated by the devices, the medical records of the older people, and the analysis results produced by medical experts or by developed software, need to be stored effectively, aiming to fulfill the data access requirements that arise during offline analysis.

After contacting all partners and the vendors that produce the devices, a summary of the expected input data was made. This summary was used as a guide towards the design of the database. Based on the nature of the data of the FrailSafe project, a NoSQL database was decided that will be more appropriate. Among the numerous NoSQL solutions, the Apache HBase was chosen. The motivation behind this choice is that HBase is part of the Hadoop ecosystem, which provides high scalability in data analysis and knowledge discovery algorithms. Towards the analysis of the data, Apache Spark was selected as it is one of the state-of-the-art data processing engines, which can efficiently execute streaming, machine learning or SQL workloads that require fast iterative access to datasets.

The data management and analysis platform has been deployed in the cloud, using the cloud service by Amazon Web Services (AWS) which has been set up by the integrators of the project, Gruppo Sigla. The integration of the submodules which generate the data with the data management and analysis platform has started and is expected to be completed in the following months.

Regarding the task of the offline data analysis, our work has been focused on several areas. Having collected all clinical data from the eCRF platform, the first step to start the data analysis was to summarize and describe the population's behavior. To that end, a group-wise univariate analysis was performed across three different bases: a. Frailty status based analysis, b. Age based analysis and c. Gender based analysis. Additionally, the clinical measurements from eCRF were used for multivariate statistical analysis. Specifically, their predictive ability towards the development of a frailty index was examined. Two different frailty indexes (*FI*) were computed, one aiming to predict the discrete Fried classification score (FI_1) and one trying to estimate a continuous score as a linear combination of the 5 criteria related to Fried classification (FI_2). The ultimate goal is to investigate whether the proposed frailty indexes are more reliable predictors of frailty transition than standard classification scores. A preliminary analysis was performed for deliverable D4.2 using a subset of data while a more thorough evaluation (based on more samples, features and examined techniques) was performed for this deliverable.

Additionally to the assessment of clinical variables, measurements from the FrailSafe devices (ECG, IMUs, games) were analysed and used to build predictive models of frailty based on quantification metrics defined in deliverable D2.1 (proxy outcomes).

Finally, multi-dimensional time series analysis has been targeted towards the problem of activity classification, and towards the prediction of frailty. For the former, temporal, and spectral features extracted from the sensor signals (accelerometer and gyroscope) were combined and used to train motion dependent binary classification models. Each individual model was capable to recognize one motion versus all the others. Afterwards the decisions were weighted by their sensitivity on the training set combined by a fusion function. The proposed methodology was evaluated using SVMs for the motion dependent classifiers and was compared against the common multiclass classification approach optimized using either feature selection or subject dependent classification. Previously developed models (for D4.1) have been tested on FrailSafe data and their failures have been addressed in this deliverable by defining a new protocol for collection of annotated data and by building new models dedicated to the older population.

For prediction of frailty, some preliminary work has started on the investigation of deep learning techniques for seamless extraction of a features' hierarchy and in-depth analysis of the time series data. Since this method focuses on signal processing and analysis the method will be described in deliverable D4.15, while results will be collected at the evaluation phase of the project in which the largest amount of data will be available, absolutely necessary for deep learning techniques.

2 Data Management

Managing FrailSafe's multimodal data is a task of great importance. The huge data files that contain the raw sensor data generated by the devices, the medical records of the older people, the annotations generated by the experts (both clinicians and researchers), and the files that contain the analysis results need to be stored effectively, aiming to fulfill the data access requirements that arise during offline analysis.

2.1 FrailSafe Data Description

2.1.1 Clinical data (e-CRF)

The data which are collected by medical personnel during the clinical evaluation, are inserted to the eCRF (Electronic Case Report Form) using the Clinical Web Portal that Gruppo Sigla has developed. The eCRF is mainly composed by a series of forms assessing the participant's clinical status:

- Generalities
- Medical history
- Clinical examination / measurements
- Balance and gait evaluation
- Fried's criteria of frailty
- Mini-Mental State Examination
- Sensory system evaluation
- Nutritional assessment
- Activities of daily living
- Cognitive/emotional evaluation
- Self-evaluation scales

Additionally, the eCRF hosts forms to collect additional data for the participants such as:

- Housing evaluation
- The Big-five assessment
- Social media questionnaire
- Phone follow-up
- Devices registered to the participant during FrailSafe home visits
- Undesirable events during monitoring time

Gruppo Sigla has developed an API which can be used to export the summary of the stored data into csv files, so that they can be inserted to the FrailSafe database and can be used for the analysis.

2.1.2 Sensorized strap/vest

The sensorized strap/vest which is manufactured by our partner Smartex, is equipped with a series of sensors which provide useful measurements for FrailSafe participants. These measurements can be grouped in these categories:

1. **ECG measurements**: The main measurement of this category is the value of the ECG signal coupled together with a quality index which shows how accurate the measurement actually is. This helps ignoring measurements for which the quality is low because the strap was not placed properly. Using the ECG signal, the vest software calculates useful clinical measurements such as Heart Rate, Heart Rate Variability, and R-R interval distance in ECG signal.
2. **IMU measurements**: The strap (WWS) is equipped with a "light" IMU measuring only the participant's acceleration in X-Y-Z axis (using an accelerometer), while he/she is wearing the strap. The new vest (WWBS) is equipped with one IMU placed on the chest which is measuring the participant's acceleration, the angular rate, and the magnetic field surrounding the body in X-Y-Z axis using accelerometer, gyroscope and magnetometer, respectively. There are additional two IMUs placed on the arms which just send extracted quaternions. These measurements might not be directly connected with clinical parameters, however they are needed in order to run Fall Detection and Activity Classification algorithms.
3. **Respiration measurements:** The strap is also equipped with a piezoresistive point placed on the thorax, which is used to measure the strain on the thorax caused by the participant's breathing. The strap uses this measurement to calculate the Breathing Rate, and the Breathing Amplitude of the participant.
4. **Activity attributes:** Additionally, some measurements are provided about the activity the participant performs while wearing the strap. There is a simple activity recognition (lying, standing, walking, running) which however is not as accurate as the activity classification algorithm developed by the UoP. Also, there is a counter measuring the number of steps the participant has done while wearing the strap, and the step period which shows how fast/slow the steps are being done.

These measurements can be summarized in the following table:

Table 1: Sensorized strap/vest recorded parameters.

Recorded parameter	Description	Values (1 unit)	Sampling rate

ECG Value	Electric signal measuring the ECG	0.8 mV	250Hz
ECGquality Value	ECG signal quality	0-255 (0=poor, 255=excellent)	1/5sec
ECGHR Value	Heart rate	Beats/minute	1/5sec
ECGRR Value	R-R intervals	number of samples between R-R peaks	1/5sec
ECGHRV Value	Heart rate variability	ms	1/60sec
AccX-Y-Z Value	Accelerometer in X-Y-Z axes	0.97 10-3 g	25Hz
GyroX-Y-Z Value	Gyroscope in X-Y-Z axes	0.122 °/s	25Hz
MagX-Y-Z Value	Magnetometer in X-Y-Z axes	0.6 µT	25Hz
RespPiezo Value	Electric signal measuring the chest pressure on the piezoelectric point	0.8 mV	25Hz
RespQuality Value	Respiration signal quality	0-255 (0=poor, 255=excellent)	1/5sec
BR Value	Breathing rate	Breaths/minute	1/5sec
BA Value	Breathing Amplitude	logic levels	1/15sec
Activityenergy Value	estimation of energy activity	is just an estimation (0=no activity, 255=max of activity)	1/5sec
Activityclass Value	Activity performed	0=other, 1=lying, 2=standing/sitting, 3=walking, 4=running	1/5sec
Activity1Pace Value	Step period	ms	1Hz
ActivityPace Value	Pace	steps/min	1/5sec
Q0-Q1-Q2-Q3 values	Quaternions from main electronicdevice (Q0, Q1, Q2, Q3 components)	Q14 format	25 Hz
QEL0-QEL1-QEL2-QEL3 values	Quaternions from external left arm device (Q0, Q1, Q2, Q3 components)	Q14 format	25 Hz
QER0-QER1-QER2-QER3 values	Quaternions from external right arm device (Q0, Q1, Q2, Q3 components)	Q14 format	25 Hz

The WWS/WWBS data are downloaded by the medical personnel once the home visit session has been performed (at the end of the 5-day period) in a compressed file format. This file is then uploaded by the medical personnel to the Amazon cloud and the Data Grabber (described in Section 2.2) loads it to the database.

2.1.3 GPS logger (smartphone)

The GPS logger application for the smartphone (developed by CERTH) collects measurements about the geographic location of the participants. The location is obtained by receiving a signal from GPS satellites, thus it is accurate only for the outdoor localization of the participant (in a macroscopic scale). The specific measurements obtained are the latitude, longitude, and elevation of each geographic location, together with the accuracy of the measurement and the orientation of the movement. Combining subsequent points of the location of the participant, we can derive other measurements with more significant clinical value such as the speed of movement, the distance covered etc. The GPS logger application additionally measures the number of steps the participant has made, using the phone sensors.

These measurements can be summarized in the following table:

Table 2: GPS logger recorded parameters.

Recorded parameter	Description	Sampling rate
Latitude	Satelite estimation of the latitude of the geolocation point	variable
Longitude	Satelite estimation of the longitude of the geolocation point	variable
Elevation	Elevation of the geolocation point (sea level)	variable
Speed	Indicative speed of movement	variable
Accuracy	Accuracy of the geolocation	variable
Bearing	Orientation of the movement	variable
Steps	Step counter (based on android sensor)	variable

These recorded data can also be visualized in order to evaluate the outdoor area in which the participants are moving. In Figure 1 we can see an example of such a visualization.

GPS logger currently uploads the data in an FTP server at the premises of CERTH, and Data Grabber collects data from there. In the near future, CERTH will integrate a service in the Amazon cloud and the uploading will be performed directly there.

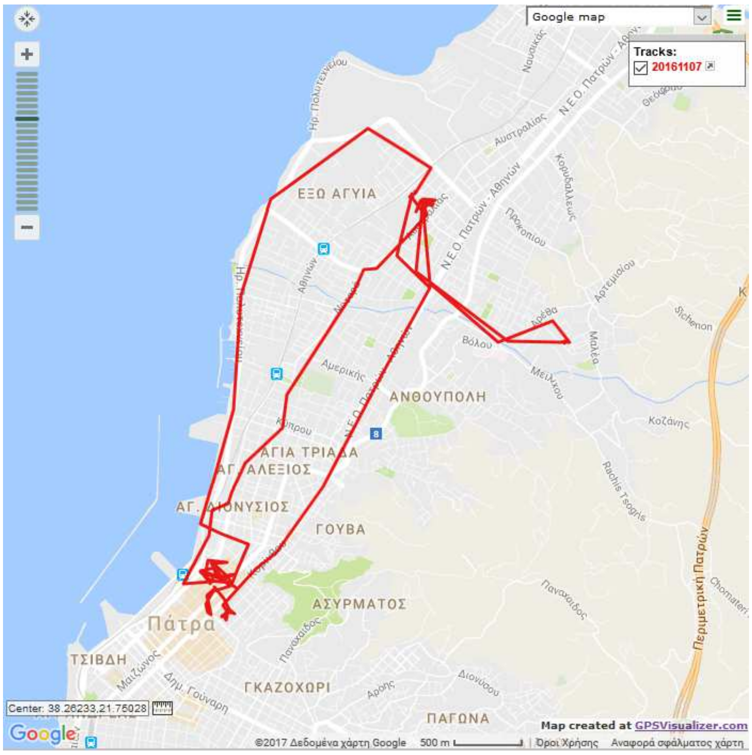


Figure 1: Visualization of a participant's outdoor movement for a specific day.

2.1.4 Beacons

CERTH has developed an application for the smartphone, which can be used with the beacons to perform indoors localization of the participant. Each measurement obtained from the developed app contains the room name that the participant is located and the time the participant entered the room. Combining subsequent measurements we can derive the information of the aggregated time the participant has spent in each room.

Table 3: Beacon recorded parameters.

Recorded parameter	Description	Sampling rate
Room name	Label describing the room that the participant is in.	variable

2.1.5 Games

So far, the older persons were exposed to three games, the Force Analyzer and the Red Wings games which were developed by Brainstorm and the Virtual Supermarket game which was developed by CERTH.

The Force Analyzer shows a panel to the user which resembles a meter and asks him/her to apply maximum force on the dynamometer for as long as he/she can. The game records a log file with measurements about the force applied at each time point. Combining the subsequent measurements of the log files, we can derive the average and maximum force applied as well as the endurance of the older person.

During the Red Wings game, the user is navigating a small plane by applying force in the dynamometer. The log file records measurements connected with the game such as the speed that the plane is moving, the distance it has covered, the height which it is at, and the number of lives the player still has. Additionally, as the game is operated by the dynamometer, the force of the participant is being collected. Combining the subsequent measurements of the log files, we can derive the total time the participant played the game, the total distance covered (total score), the maximum force on the dynamometer and the endurance.

The virtual supermarket game simulates a supermarket, where the participant has a shopping list and needs to navigate inside the supermarket, collect the items of the shopping list and pay at the cashier. The game collects measurements about the time the participant needed to buy each item, the time the participant spent on each part of the supermarket, the value of the products, money paid etc.

These measurements can be summarized in the following tables:

Table 4: Force Analyzer recorded parameters.

Recorded parameter	Description	Sampling rate
Force	The force applied by the user.	60Hz

Table 5: Red Wings recorded parameters.

Recorded parameter	Description	Sampling rate
Distance	The distance achieved.	60Hz
Force	The force applied by the user.	60Hz
Height	The height of the plane.	60Hz
Lives	The lives remaining.	60Hz
Speed	The plane's speed.	60Hz

Table 6: Virtual supermarket recorded parameters for each session.

Recorded parameter	Description
Total time	Total session duration.
Item types bought	How many types of items were bought (regardless of the quantities for each type).
Item types in list	How many types of items were listed in the shopping list (i.e. which was the goal).
Item quantities bought	The total quantity of the items of all types bought.

Item quantities in list	The total quantity of the items of all types listed in the shopping list (i.e. which was the goal).
Item types not in list	How many types of items were bought that were not listed in the shopping list.
Item quantities not in list	The total quantity of the unlisted items that were bought.
Items value	The total price of the items (and quantities) that were bought.
Money paid	The total price paid by the user at the end.

Initially the data were collected by Brainstorm and CERTH at their local premises, and were sent periodically to the FrailSafe database in batches. During the last months Brainstorm has integratedits services to the FrailSafe cloud and data are sent directly to the FrailSafe database. CERTH is expected to complete this integration in the next period.

2.1.6 Auxiliary medical devices

In the FrailSafe project, there are some auxiliary medical devices which are used to measure clinical parameters. These devices are used a limited number of times, thus they do not generate a large number of data for each participant. The auxiliary devices are:

- **Mobil-o-graph (by Agaedio)**: This device is used during clinical examination to measure the arterial stiffness of participants (applied only to France clinical site). It collects measurements about blood pressure, heart rate, cardiac output, vascular resistance, augmentation pressure etc.
- **Impedance scale (by FORA)**: This device is also used only during clinical examination and collects measurements about weight, body fat, BMI etc.
- **Blood pressure monitor (by FORA)**: This device is used during the FrailSafe sessions, so the participant is operating the device 3 times per day in order to measure his/her blood pressure.

2.2 FrailSafe Database (HBase)

One of the main tasks of WP4 is to gather all the data which are either generated by the various medical devices, or collected by the medical personnel, and store them consistently into a database. The data are then analyzed and aggregated towards providing the Virtual Patient Model with which the clinicians will interact to design their interventions. The data collection and aggregation conceptual plan is shown in Figure 2.

In the heart of the system, there is a cluster of 4 Amazon EC2 machines, which hosts two services:

a) A distributed NoSQL database (Apache HBase) which stores the FrailSafe data, and
b) A distributed data processing engine (Apache Spark), which is used to process and aggregate the data.

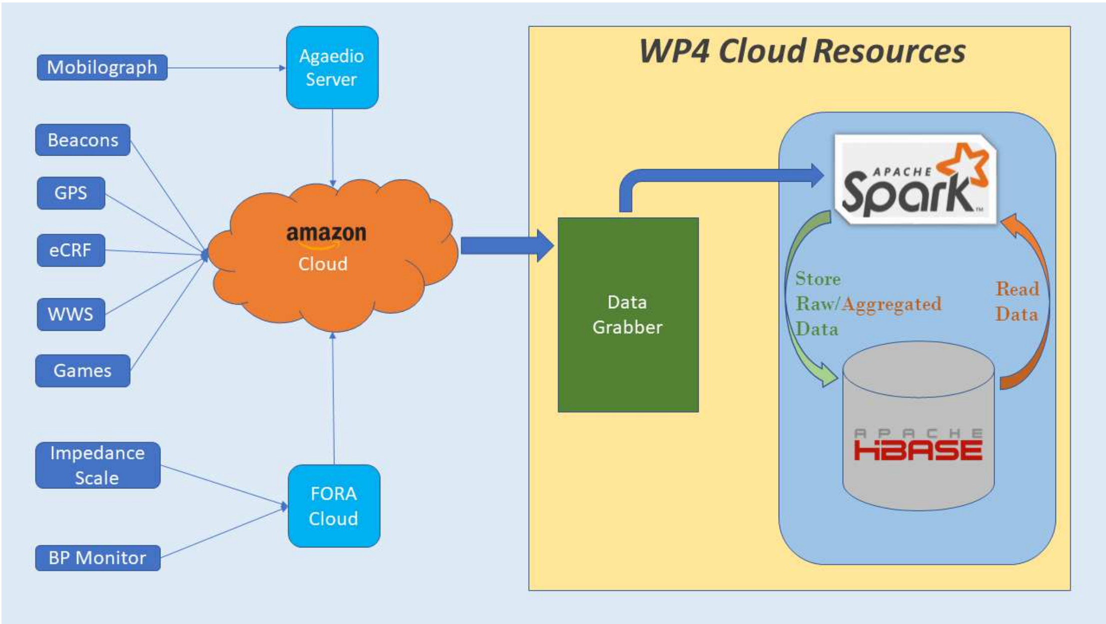


Figure 2: WP4 cloud resources for data collection and aggregation.

There is an additional Amazon EC2 machine called the "Data Grabber", which is responsible for collecting the data uploaded to the Amazon cloud by the different submodules of the FrailSafe project, or the external servers of the machine vendors (Agaedio and FORA). For each data stream, a different process has been designed and implemented:

- eCRF data: This includes all clinical parameters which are collected from the medical personnel and inserted to the clinical web platform using forms. The developed process includes several steps (i) communication with the eCRF API and retrieval of csv files containing the clinical data, (ii) pre-processing of the data, (iii) aggregation of selected features, and (iv) storage of clinical and aggregated data into the FrailSafe database.

- WWS/WWBS data: This includes the physiological data recorded by the wearable vest, which are downloaded by the medical personnel and then uploaded to the clinical web platform in a compressed format. The developed process includes the following steps: (i) communication with the eCRF API and retrieval of compressed wwsx files containing the physiological data, (ii) pre-processing of the data, (iii) perform activity classification algorithm, (iv) aggregation of physiological parameters based on activity, (v) generate alerts based on the aggregated physiological parameters, and (vi) storage of physiological and aggregated data into the FrailSafe database.

- Outdoor Localization data: This includes data recorded in the smartphone by the outdoor localization app and sent to the FrailSafe cloud. The developed process includes the following steps: steps (i) communication with the outdoor localization cloud repository and retrieval of GPS files, (ii) pre-processing of the data, (iii) aggregation of selected features, and (iv) storage of GPS and aggregated data into the FrailSafe database.

- FORA Blood pressure data: This includes data recorded in the smartphone by the FORA blood pressure app and sent to the FORAcare telehealth cloud. The developed process includes the following steps: steps (i) communication with the FORAcare telehealth cloud repository and retrieval of XML files containing blood pressure data, (ii) pre-processing of the data, (iii) aggregation of selected features and generation of alerts, and (iv) storage of blood pressure and aggregated data/alerts into the FrailSafe database.

- Games data: This includes data recorded in the tablet by the FrailSafe Game Platform and sent to the FrailSafe cloud. The developed process includes the following steps: steps (i) communication with the Game Platform cloud repository and retrieval of csv files containing both game log files and summaries, (ii) pre-processing of the data, (iii) storage of game log files and summaries into the FrailSafe database.

- Indoor localization data: This includes data recorded in the smartphone by the indoor localization app and sent to the FrailSafe cloud. The developed process includes the following steps: steps (i) communication with the indoor localization API and retrieval of JSON containing indoor localization data, (ii) pre-processing of the data, (iii) aggregation of selected features, and (iv) storage of indoor localization and aggregated data into the FrailSafe database.

- Social media data: This includes data recorded in the social media sensing portal. There is an API developed that facilitates the social media sensing portal to store or retrieve participants' metadata, raw text data, and analysed features. Additionally, the API allows the social media sensing portal to store aggregated data/alerts into the FrailSafe database.

- Mobilograph data (not integrated yet): Currently data collected in the Agaedio servers are being sent periodically to the FrailSafe integrators (Sigla) by e-mail and we are inserting them manually to the FrailSafe database. In the next period we plan to automate this process.

2.2.1 Database schema

Dealing with the multimodal data generated in the FrailSafe project is a demanding task. We started our efforts by designing a scheme capable of capturing the complexity of the data and their relations. In this section we will show some parts of the ER (Entity-Relationship) diagram that correspond to the most important data to be stored.

The overview of the schema is presented in Figure 3 and captures the data that are recorded by the FrailSafe devices, the data acquired from the clinical evaluations, and the data that are generated as a result of the offline analysis process.

Furthermore, the utilization of this schema gives us the ability to store both clinical and physiological data and metadata and thus, gives us the ability to directly search for relationships and correlations among the data.

Sensor recording-related entities

One of the most important entities of the database is the Sensor Recording. By this term we refer to the physiological data coming from the sensorized vest that the FrailSafe participants wear. Important information that should be stored for every sensor recording includes:

- The subjects' ID.
- A timestamp.
- The type of sensor (e.g. ECG, Heart Rate,Respiration Rate).
- The value of the recording.

The actual data (the signals) from the recording are also stored in one or multiple files and thus, one recording is associated with a file. A more detailed description of the recording-related information is shown at the ER diagram inFigure 4. We note here that not all entity attributes are shown due to spacing issues.

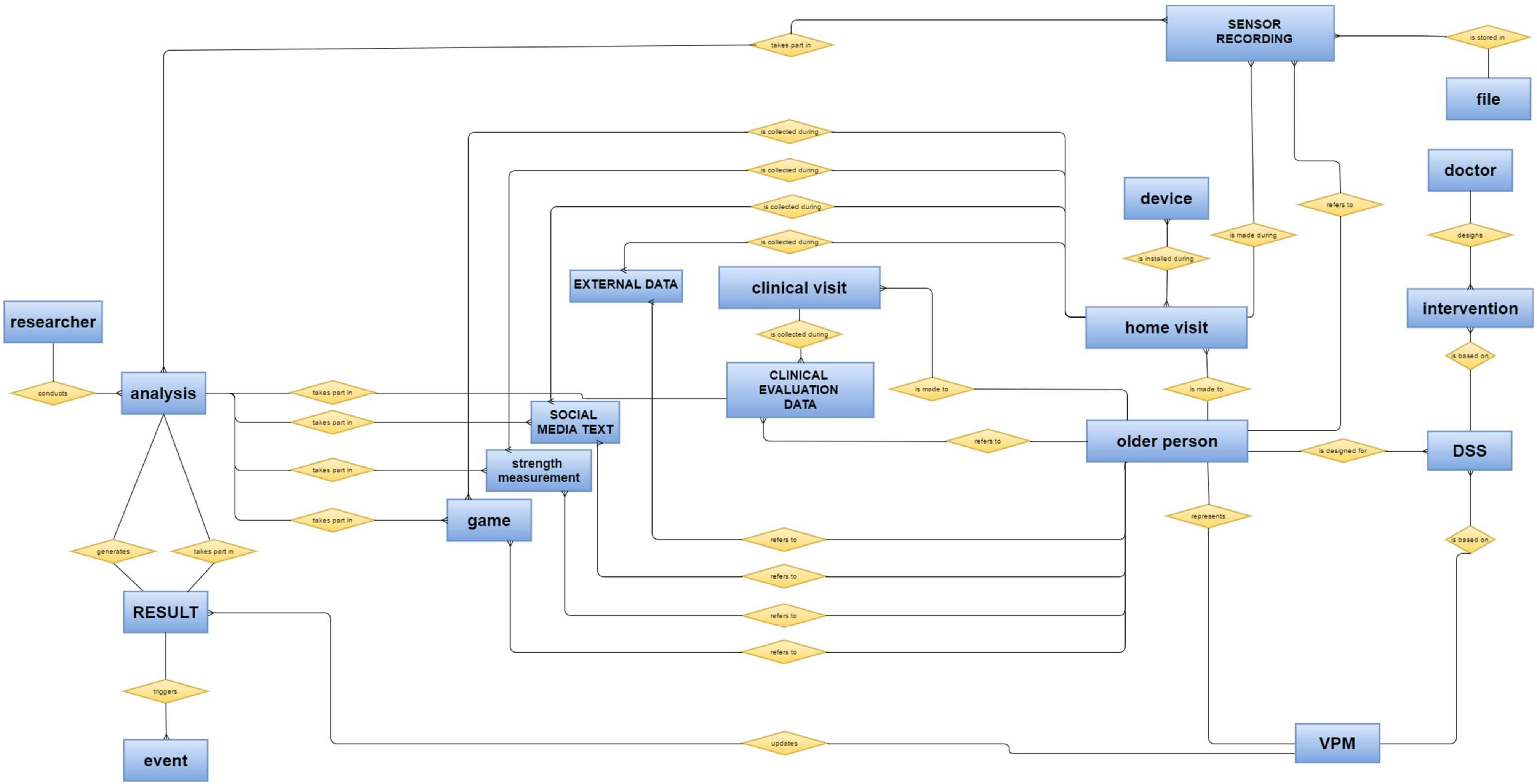


Figure 3: Database Schema for FrailSafe.

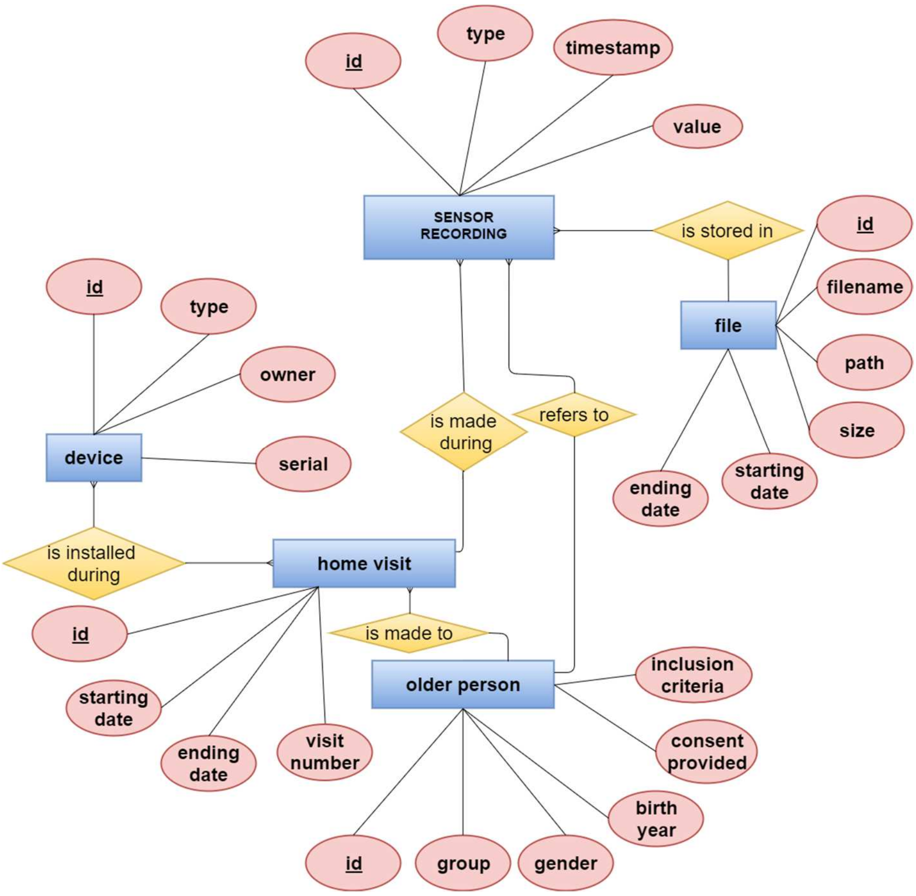


Figure 4: Sensor recording ER diagram.

Analysis-related entities

Another very important task in the database is to keep track of every analysis that takes place and the corresponding results. For this purpose, our database schema includes entities that are able to describe the whole analysis process: the purpose of the analysis, the utilized data and the results.

As we mentioned earlier, since marking events of interest is a very common analysis task, we added the Event entity for the description of events. Also, the result of the analysis might significantly affect the Virtual Patient Model, and subsequently the Decision Support System that is connected to it.

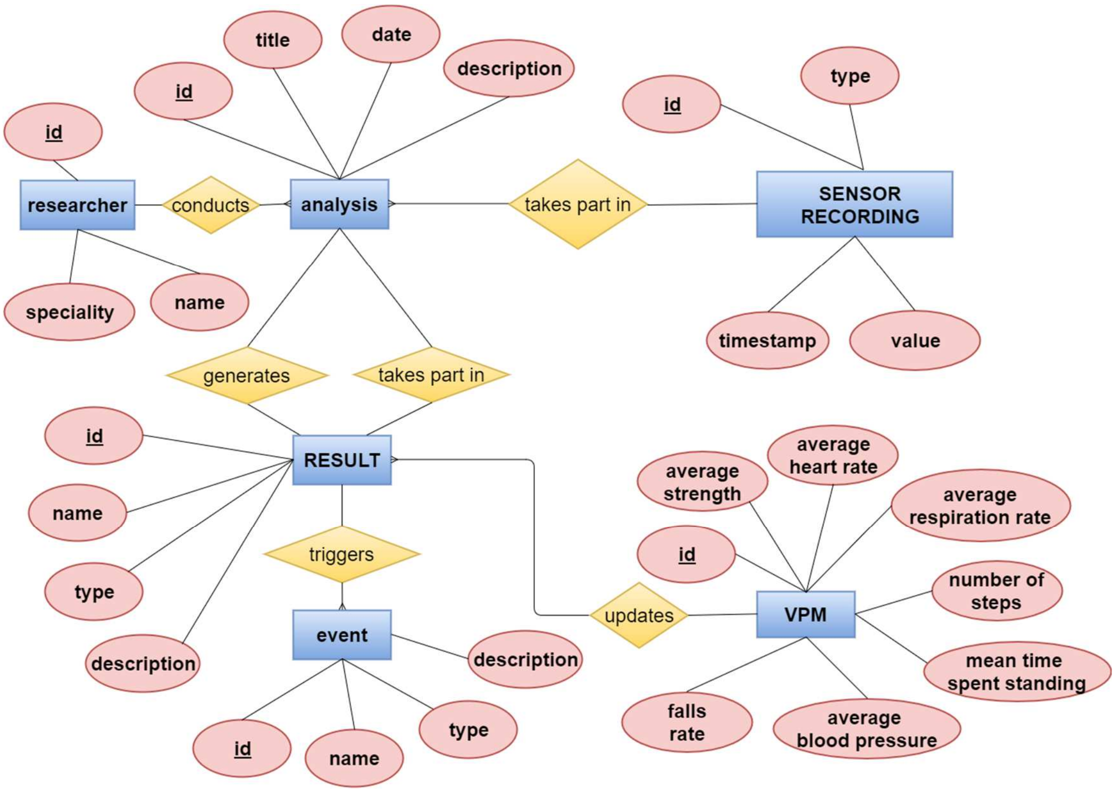


Figure 5: Data Analysis ER diagram.

2.2.2 NoSQL Databases

Based on the nature of the data of the FrailSafe project, a NoSQL (Not-only-SQL) database was decided that will be more appropriate. A NoSQL database is one that has been designed to store, distribute and access data using methods that differ from relational databases (RDBMSs), where data is placed in tables and data schemas are carefully designed before the database is built. NoSQL Databases especially target large sets of distributed data.

NoSQL technology was originally created and used by Internet leaders such as Facebook, Google, Amazon, and others who required database management systems that could write and read data anywhere in the world, while scaling and delivering performance across massive data sets and millions of users.

Today, almost every company and organization has to deliver cloud applications that personalize their customer's experience with their business, with NoSQL being the database technology of choice for powering such systems.

Types of NoSQL Databases

Several different varieties of NoSQL databases have been created to support specific needs and use cases. These fall into four main categories:

Key-value data stores: Key-value NoSQL databases emphasize simplicity and are very useful in accelerating an application to support high-speed read and write processing of non-transactional data. Stored values can be any type of binary object (text, video, JSON document, etc.) and are accessed via a key. The application has complete control over what is stored in the value, making this the most flexible NoSQL model. Data is partitioned *and replicated* across a cluster to get scalability and availability. For this reason, key value stores often do not support transactions. However, they are highly effective at scaling applications that deal with high-velocity, non-transactional data.

Document stores: Document databases typically store self-describing JSON, XML, and BSON documents. They are similar to key-value stores, but in this case, a value is a single document that stores all data related to a specific key. Popular fields in the document can be indexed to provide fast retrieval without knowing the key. Each document can have the same or a different structure.

Wide-column stores: Wide-column NoSQL databases store data in tables with rows and columns similar to RDBMS, but names and formats of columns can vary from row to row across the table. Wide-column databases group columns of related data together. A query can retrieve related data in a single operation because only the columns associated with the query are retrieved. In an RDBMS, the data would be in different rows stored in different places on disk, requiring multiple disk operations for retrieval.

Graph stores: A graph database uses graph structures to store, map, and query relationships. They provide index-free adjacency, so that adjacent elements are linked together without using an index.

How NoSQL Differs from Relational Databases

NoSQL databases are not a direct replacement for a relational database management system (RDBMS). For many data problems, though, NoSQL is a better match than an RDBMS, as they are designed to support different application requirements (Table 6).

Table 6: Comparison between RDBMS and NoSQL.

RDBMS	NoSQL
Centralized applications (e.g. ERP)	Decentralized applications (e.g. Web, mobile and IOT)
Moderate to high availability	Continuous availability; no downtime
Moderate velocity data	High velocity data (devices, sensors, etc.)
Data coming in from one/few locations	Data coming in from many locations

Primarily structured data	Structured, with semi/unstructured
Complex/nested transactions	Simple transactions
Primary concern is scaling reads	Concern is to scale both writes and reads
Philosophy of scaling up for more users/data	Philosophy of scaling out for more users/data
To maintain moderate data volumes with purge	To maintain high data volumes; retain forever

Advantages over RDBMSs

The advantages of NoSQL databases are no secret, especially when cloud computing has gained wide adoption.

NoSQL databases were created in response to the limitations of traditional relational database technology. When compared against relational databases, NoSQL databases are more scalable and provide superior performance, and their data model addresses several shortcomings of the relational model. More specificallyNoSQL databases have been widely adopted in many enterprises for the following reasons:

Elastic scalability: NoSQL databases use a horizontal scale-out methodology that makes it easy to add or reduce capacity quickly and non-disruptively with commodity hardware. This eliminates the tremendous cost and complexity of manual sharding that is necessary when attempting to scale RDBMS.

Big data applications: Given that transaction rates are growing from recognition, there is need to store massive volumes of data. While RDBMSs have grown to match the growing needs, but it's difficult to realistically use one RDBMS to manage such data volumes. These volumes are however easily handled by NoSQL databases.

Database administration: The best RDBMSs require the services of expensive administrators to design, install and maintain the systems. On the other hand, NoSQL databases require much less hands-on management, with data distribution and auto repair capabilities, simplified data models and fewer tuning and administration requirements. However, in practice, someone will always be needed to take care of performance and availability of databases.

Economy:RDBMSs require installation of expensive storage systems and proprietary servers, while NoSQL databases can be easily installed in cheap commodity hardware clusters as transaction and data volumes increase. This means that you can process and store more data at much less cost.

Performance: By simply adding commodity resources, enterprises can increase performance with NoSQL databases. This enables organizations to continue to deliver reliably fast user experiences with a predictable return on investment for adding resources—again, without the overhead associated with manual sharding.

High Availability: NoSQL databases are generally designed to ensure high availability and avoid the complexity that comes with a typical RDBMS architecture that relies on primary and secondary nodes. Some “distributed” NoSQL databases use a masterless architecture that automatically distributes data equally among multiple resources so that the application remains available for both read and write operations even when one node fails.

Global Availability: By automatically replicating data across multiple servers, data centers, or cloud resources, distributed NoSQL databases can minimize latency and ensure a consistent application experience wherever users are located. An added benefit is a significantly reduced database management burden from manual RDBMS configuration, freeing operations teams to focus on other business priorities.

Flexible Data Modeling: NoSQL offers the ability to implement flexible and fluid data models. Application developers can leverage the data types and query options that are the most natural fit to the specific application use case rather than those that fit the database schema. The result is a simpler interaction between the application and the database and faster, more agile development.

2.2.3 HBASE

Apache HBase is a massively scalable, distributed big data store in the Apache Hadoop ecosystem. It is an open-source, non-relational, versioned database which runs on top of Amazon S3 (using EMRFS) or the Hadoop Distributed Filesystem (HDFS), and it is built for random, strictly consistent real-time access for tables with billions of rows and millions of columns. Additionally, Apache HBase has tight integration with Apache Hadoop, Apache Hive, and Apache Pig, so you can easily combine massively parallel analytics with fast data access. Apache HBase's data model, throughput, and fault tolerance is a good match for workloads in ad tech, web analytics, financial services, applications using time-series data, and many more.

We selected Apache HBase for the storage of the FrailSafe data, as column-family databases are the most suitable for the nature of our case study. More specifically, in FrailSafe project except for the abstract medical information, sensor raw data are collected either in large batches or in real-time. The frequency of the measurements is very high (e.g. 25Hz), hence the DBMS has to be capable of managing hundreds of gigabytes of data efficiently in real-time. HBase is considered as the most suitable option as: (1) it is a distributed, scalable, big data store, (2) it is suitable for random, real-time read/write access to Big Data, (3) it is capable of hosting very large tables with billions of rows × millions of columns, (4) it is optimized for queries over large datasets. Its features are described in detail below.

Features and Benefits

Deep Integration with Apache Hadoop: Since HBase has been built on top of Hadoop, it supports parallelized processing via MapReduce. HBase can be used as both a source and output for MapReduce jobs. Integration with Apache Hive allows users to query HBase tables using the Hive Query Language, which is similar to SQL.

Strong Consistency: The HBase project has made strong consistency of reads and writes a core design tenet. A single server in an HBase cluster is responsible for a subset of data, and with atomic row operations, HBase is able to ensure consistency.

Failure Detection: When a node fails, HBase automatically recovers the writes in progress and edits that have not been flushed, then reassigns the region server that was handling the data set where the node failed.

Real-time Queries: HBase is able to provide random, real-time access to its data by utilizing the configuration bloom filters, block caches, and Log Structured Merge trees to efficiently store and query data.

Fast Performance at Scale: Apache HBase is designed to maintain performance while scaling out to hundreds of nodes, supporting billions of rows and millions of columns. Additionally, it can be combined with Apache Phoenix for low-latency SQL over massive HBase tables or creating secondary indexes for increased performance.

Flexible Data Model: Apache HBase is wide-column store, allowing you to define arbitrary columns for each row for filtering purposes. Additionally, HBase adds a timestamp to each cell and can keep previous versions, allowing easy storage and access to the lineage of a dataset. Each cell is a byte array and can store a payload in the MB range, giving flexibility in data types stored. Apache Phoenix and Apache Hive enable SQL access over Apache HBase tables.

2.2.4 Hadoop framework

Apache HBase runs on top of the Hadoop, which is an open-source software frameworkused for distributed storage and processing of big data using theMapReduce programming model. It consists of computer clusters built fromcommodity hardware. All the modules in Hadoop are designed with a fundamental assumptionthat hardware failures are common occurrences and should be automaticallyhandled by the framework.

The core of Apache Hadoop consists of a storage part, known as Hadoop Distributed File System (HDFS), and a processing part which is a MapReduce programming model. Hadoop splits files into large blocks and distributes them across nodes in a cluster. It then transfers packaged code into nodes to process the data in parallel. This approach takes advantage of data locality, where nodes manipulate the data they have access to. This allows the dataset to be processed faster and more efficiently than it would be in a more conventional supercomputer architecture that relies on a parallel file system where computation and data are distributed via high-speed networking.

The base Apache Hadoop framework is composed of the following modules:

- *Hadoop Common* – contains libraries and utilities needed by other Hadoop modules;
- *Hadoop Distributed File System (HDFS)* – a distributed file-system that stores data on commodity machines, providing very high aggregate bandwidth across the cluster;
- *Hadoop YARN* – a platform responsible for managing computing resources in clusters and using them for scheduling users' applications;and
- *Hadoop MapReduce* – an implementation of the MapReduce programming model for large-scale data processing.

2.3 FrailSafe Data processing (Apache Spark)

Apache Spark is a fast, in-memory data processing engine with elegant and expressive development APIs to allow data workers to efficiently execute streaming, machine learning or SQL workloads that require fast iterative access to datasets. Within the FrailSafe project, an Apache Spark cluster is used for processing and aggregating batches of sensor data and storing them into the HBase.

Spark has several advantages compared to other big data and MapReduce technologies like Hadoop and Storm. First of all, Spark gives us a comprehensive, unified framework to manage big data processing requirements with a variety of data sets that are diverse in nature (text data, graph data etc.) as well as the source of data (both batch and real-time streaming data). Spark enables applications in Hadoop clusters to run up to 100 times faster in memory and 10 times faster even when running on disk. Spark offers programming interface in several languages such as Java, Scala, and Python. In addition to Map and Reduce operations, it supports SQL queries, streaming data, machine learning and graph data processing. These capabilities can be combined and run in a single data pipeline as we do in the FrailSafe project (see 2.3.5).

2.3.1 Hadoop and Spark

Hadoop as a big data processing technology has been around for 10 years and has proven to be the solution of choice for processing large data sets. MapReduce is a great solution for one-pass computations, but not very efficient for use cases that require multi-pass computations and algorithms. Each step in the data processing workflow has one Map phase and one Reduce phase and you'll need to convert any use case into MapReduce pattern to leverage this solution.

The Job output data between each step has to be stored in the distributed file system before the next step can begin. Hence, this approach tends to be slow due to replication & disk storage. Also, Hadoop solutions typically include clusters that are hard to set up and manage. It also requires the integration of several tools for different big data use cases (like Mahout for Machine Learning and Storm for streaming data processing).

In order to perform complicated procedures, one should have to string together a series of MapReduce jobs and execute them in sequence. Each of those jobs might be high-latency, and none could start until the previous job had finished completely.

On the other hand, Sparkallows programmers to develop complex, multi-step data pipelines using the directed acyclic graph (DAG) pattern. It also supports in-memory data sharing across DAGs, so that different jobs can work with the same data.Spark runs on top of existing Hadoop Distributed File System (HDFS) infrastructure to provide enhanced and additional functionality.

2.3.2 Features

Spark takes MapReduce to the next level with less expensive shuffles in the data processing. With capabilities like in-memory data storage and near real-time processing, the performance can be several times faster than other big data technologies. It also supports lazy evaluation of big data queries, which helps with optimization of the steps in data processing workflows. It provides a higher-level API to improve developer productivity and a consistent architect model for big data solutions.

Additionally, Spark holds intermediate results in memory rather than writing them to disk which is very useful especially when you need to work on the same dataset multiple times. It is designed to be an execution engine that works both in-memory and on-disk. Its operators perform external operations when data does not fit in memory. It can be used for processing datasets that are larger than the aggregate memory in a cluster.

Spark will attempt to store as much as data in memory and then will spill to disk. It can store part of a data set in memory and the remaining data on the disk. With this in-memory data storage, it comes with a performance advantage.

Other Spark features include:

- Supports more than just Map and Reduce functions.
- Optimizes arbitrary operator graphs.
- Lazy evaluation of big data queries which helps with the optimization of the overall data processing workflow.
- Provides concise and consistent APIs in Scala, Java and Python.
- Offers interactive shell for Scala and Python. This is not available in Java yet.

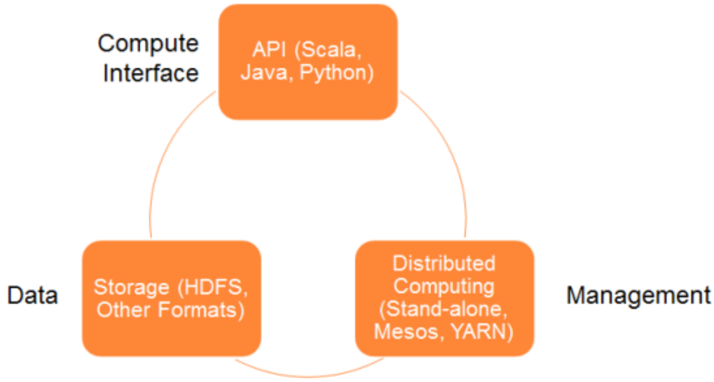
2.3.3 Architecture

Spark Architecture includes following three main components:

Data Storage: The HDFS file system is used for data storage purposes. It works with any Hadoop compatible data source including HDFS, HBase, Cassandra, etc.

API: The API allows the application developers to create Spark based applications using a standard API interface. Spark provides API for Scala, Java, and Python programming languages.

Resource Management: Spark can be deployed as a Stand-alone server or it can be deployed on a distributed computing framework like Mesos or YARN.



The concept in the Spark framework is the Resilient Distributed Dataset (RDD), which can be thought as a table in a database. Spark stores data in RDD on different partitions. RDDs help with rearranging the computations and optimizing the data processing.They are also fault tolerant because an RDD knows how to recreate and recompute the datasets.RDDs are immutable, as performing a transformation to one RDDwill returna new RDD whereas the original RDD remains the same.

RDD supports two types of operations:

- **Transformation:** Transformations don't return a single value, they return a new RDD. Nothing gets evaluated when you call a Transformation function, it just takes an RDD and returns a new RDD.Some of the Transformation functions are map, filter, flatMap, groupByKey, reduceByKey, aggregateByKey, pipe, and coalesce.
- **Action:** Action operation evaluates and returns a new value. When an Action function is called on a RDD object, all the data processing queries are computed at that time and the result value is returned.Some of the Action operations are reduce, collect, count, first, take, countByKey, and foreach.

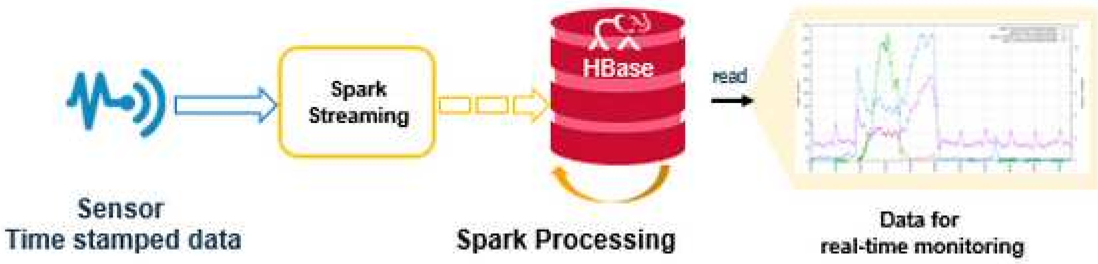
2.3.4 FrailSafe Spark cluster

Within the FrailSafe project, an Apache Spark cluster has been deployed to the Amazon cloud and is used for processing batches of sensor data and storing them into the HBase.As the majority of the FrailSafe sensor data are not transmitted in real-time to the cloud, but it is sent in batches, we have developed a pipeline within the Spark framework that is triggered when a new batch is received. The steps of this pipeline are explained below:

1. **Readinginput data**: During this step, the new files that have arrived are read by the Spark Streaming process, which divides the data stream into small

   batches called Dstreams. Each DStream, which consists of a sequence of RDDs, is then fed to the Spark Engine for processing as shown below.
2. **Processing the data**: This step includes the identification of the sensor type, the creation of a rowkey for each measurement composed by the participant id and the timestamp, and the conversion of the RDDs into HBase Put objects which are used to insert a row into HBase.
3. **Storing the processed data**: Finally, the HBase Put objects created in the previous step are stored (using the Java API of HBase) torespective HBase Tablebased on the type of sensor.

Additionally, Spark Processing engine is used by the Data Grabber in order to aggregate the stored sensor data and generate personalized daily averages for each participants (eg. average heart rate, average respiration rate etc). During the aggregation process, the batches of data are being examined to see if the physiological parameters of the participants are within normal ranges, and if not alerts are generated. The outcomes of the aggregation and alert generation procedures are stored in the VPM table of HBase as show in the following figure.



3 Data Preprocessing

3.1 Clinical data (eCRF)

The analysis of clinical data from eCRF requires a number of steps that include the conversion and standardization of data as well as handling their missing values. Details on the individual steps are provided next.

3.1.1 Initial preprocessing

The clinical data of participants are inserted by clinicians using the eCRF forms of the Clinical Web Portal. We use the eCRF API to access these data and store them to the FrailSafe database. During this process a number of preprocessing transformations are performed:

Handle French/Greek categorical values: As the three clinical sites are in France, Greece and Cyprus, some of the data are expressed in the local language (eg. “Qui” instead of “Yes”). These values are converted to English in order to be consistent among all participants.

Correct numerical physiological parameters: Some of the physiological numerical parameters are inserted to the eCRF using different measurement units (eg. height can be inserted as 1,65 meters or 165 centimeters). We convert numerical values to the same unit in order to be consistent among all participants.

Complete some empty values with default values: Some of the eCRF categorical parameters appear to be empty, however they can be classified to one category by default. As an example, if a participant does not suffer from a specific comorbidity, the eCRF field about the significance of this comorbidity to his general health is empty and we convert it to “No”. Similar completions are made for other eCRF fields.

3.1.2 Preprocessing for multivariate analysis

Exclusion criteria (for samples and variables): An entire record (subject) is excluded if the Fried classification score or any of the 5 criteria used for Fried classification (involuntary weight loss, slow walking speed, poor handgrip quality, reported exhaustion, low physical activity) are not available. In respect to a first step of feature reduction, categorical variables, such as `'gender'`, are currently excluded due to their sparsity that hinders the analysis. Comorbidities are summarized as the number of significant co-morbidities. In respect to medication, the number of different drugs is currently computed ignoring the frequency of delivery.
All numerical and ordinal variables are retained, except of variables that have many missing values (>20%) for which the imputation is highly prone to errors.

Label conversions: Binary variables (`'True'/'False'` or `'Yes'/'No'`) are converted to 1/0. Ordinal variables (categorical but with ordering) are converted to a pseudo—score in the range of [0, 1], as shown in the next Table. A uniform scale was selected.

Variable	Labels from E-CRF	Numeric conversions
frailty	'Non frail' 'Pre-frail' 'Frail'	0 0.5 1
medication delivery	'999' '888' '777'	empty 0.3 0.6
vision	'Sees poorly' 'Sees moderately' 'Sees well'	0.3 0.6 0.9
audition	'Hears poorly' 'Hears moderately' 'Hears well'	0.3 0.6 0.9
activity_regular	'< 2 h per week' '> 2 h and < 5 h per week' '> 5 h per week'	0.3 0.6 0.9
balance_single	'test non realizable' '<5 sec' '>5 sec'	empty 0.3 0.9
smoking	'Never smoked' 'Past smoker (stopped at least 6 months)' 'Current smoker'	0 0.5 1

The rest of the data (the numeric variables) are scaled also in the range [0 1].

3.1.3 Preprocessing for univariate analysis

As it will be discussed below, a univariate analysis of clinical data was conducted, by separating the eCRF variables into sections (domains). For each clinical domain that contained numerical data, box plots were extracted (see Appendix for box plots of each domain) and provided to the clinicians in order for them to evaluate the presence of outliers (a table with the outliers' ID was provided as well).

Handling outliers

Each clinical center studied the IDs of the participants that were detected as outliers from the box plots,and checked whether the reason why the values fell outside the norms was caused by accidentally filling wrong values in the eCRF platform. If so, the outliers were corrected by the clinical team. If that was not the case, then the clinical partners decided that these values were clinically plausible, and should no longer be considered as outliers, meaning that the ranges into which the values would be further separated into, should include the previously mis-detected outliers as equally possible answers.

Filling out missing data

Another important part of the univariate analysis was the identification of missing values. This problem occurred due to the fact that either the clinical team had accidentally forgotten to fill them out during the uploading to the eCRF platform, or for technical reason the uploading could not be performed earlier. Detection of missing values for each participant enabled the clinical team to fill in the corresponding answers and update the values in the eCRF platform.

3.2 Handling missing values

There are several ways to handle missing values, the choice of which affects the subsequent results. The simplest would be to replace missing values with the mean, the median, the most commonly occurring value (appropriate also for categorical data), or an interpolated estimate (usually for time-series data). Such substitutions are more appropriate for univariate analysis because they examine only the studied variable, e.g. the assumption behind the mean substitution is to retain the sample mean. Mean imputation however attenuates any correlations involving the variables that are imputed. For multivariate analysis, regression imputation can be performed, in which a regression model is estimated to predict missing values of a variable based on other variables. The model is learnt from available samples and then used to impute values in cases where that variable is missing. The *Least Absolute Shrinkage And Selection Operator* (Lasso) was selected as the regression model for each variable that had missing values. The training phase involves the computation of a vector of weights (β-coefficients) which, when multiplied with the known variables (x), will approximate the variable to be imputed (y). Mathematically it is expressed as a minimization problem

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 + \lambda||\beta||_1$$

where *X* is the covariate matrix and λ a parameter that determines the amount of regularization. The 2nd term in the equation above enforces sparsity of variables (when a coefficient becomes 0, the corresponding variable can be eliminated from the model), and thus results to more generalizable models. A maximum number of 10 coefficients was enforced. The level of regularization (parameter λ) was varied and for each value the *Mean Prediction Squared Error* (*MSE*) was calculated in a 10-fold cross validation setting. The λ resulting in minimum MSE was finally selected for the prediction model. The training set included all records that didn't have missing values.

For the remaining records, each missing variable was imputed using the corresponding regression model, if the variables it depends on (i.e. the ones with non-zero β coefficients) were available. If the dependencies included missing values, then imputation was not possible and the record could not be used in the subsequent analysis.

3.3 Data from sensors

3.3.1 Sensorized strap (WWS) and vest (WWBS) data

The sensorized wearable garment collects a series of physiological modalities (ECG signal, Respiration rate, Acceleration etc.) for each recorded session. Each modality is sampled by a specific sensor (Piezoelectric sensor, IMU etc.) according to the sensor sampling frequency, which is not uniform. The preprocessing of these data includes the synchronization of the different modalities, the removal of low quality recordings and feature extraction for analysis.

Distinction between strap and vest recordings: The clinicians use the same procedure to upload the recordings regardless if the recorded session has been performed using the sensorized strap or vest. As the values of the recorded parameters may vary significantly depending on the device, we need to be able to distinguish the sessions in which strap was used from the ones that vest was used. This procedure is performed during the preprocessing.

Mapping of each recording to the closest frailty status: The frailty status of each participant is measured periodically during the clinical examination. In between these examinations we collect measurements for the participants using the FrailSafe devices. In many of the analysis algorithms there is the need to have a frailty status associated with each of the recording, thus we performed a mapping procedure which assigns the closest frailty status to each recording.

Removal of low quality measurements: The strap/vest is equipped with an index showing the quality of recorded ECG and respiration measurements. The quality index value is generated every 5 seconds and it corresponds to the samples collected in the last time interval. The reason of having low quality recordings can be because of noise or wrong appliance of the strap/vest. During the preprocessing process, we excluded the measurements that were below the threshold suggested by the device manufacturer in order to have more reliable analysis results.

***Time synchronization*:** All sensor data are time synchronized to allow multi-channel analysis. The frequency of 25Hz was selected as reference space, since several of the sensor data are sampled at this rate. The recordings of ECG signal were downsampled from 250Hz to 25Hz, whereas other recordings (Breathing Amplitude, Breathing Rate, Heart Rate, R-R intervals, Heart rate variability, Activity classification) were upsampled. Some of the recordings were slightly time-shifted (~15msec) in order to be synchronized with the rest.

***Outlier detection*:** After time synchronization histograms of data are extracted from the time series in order to gain data insights. As we see in Figure 6 histograms per channels are extracted. Red lines indicate the 5% and 95% percentiles of each channel and the green vertical line indicates the mean of the recordings per channel. It can be observed that there are outlier values which do not allow us to have descriptive histograms. Especially in respect to the ECG Heart Rate Variability (channel 6) and the ECG RR interval (channel 7) there are some very large outlier values, probably due to imperfect placement of the devices. For example for ECG HR variability we observe some values about 3,500, that could not be plausible values.

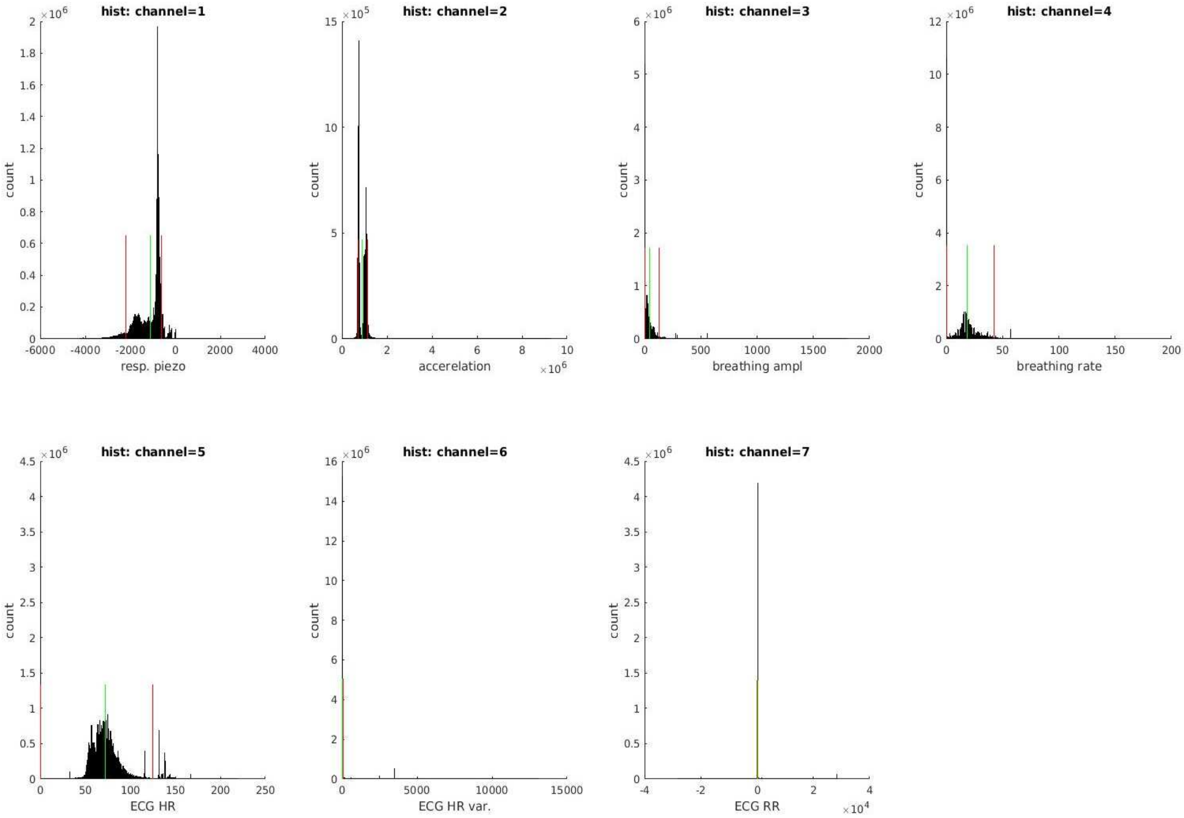


Figure 6. Data insights before cleaning: Per channel histograms.

Based on these observations we decided to discard the lower 5% values and the upper 5% values of each channel, which correspond to the 5% and 95% percentile per channel. After discarding these outlier values we extracted again the per channel histograms, which are shown in Figure 7.

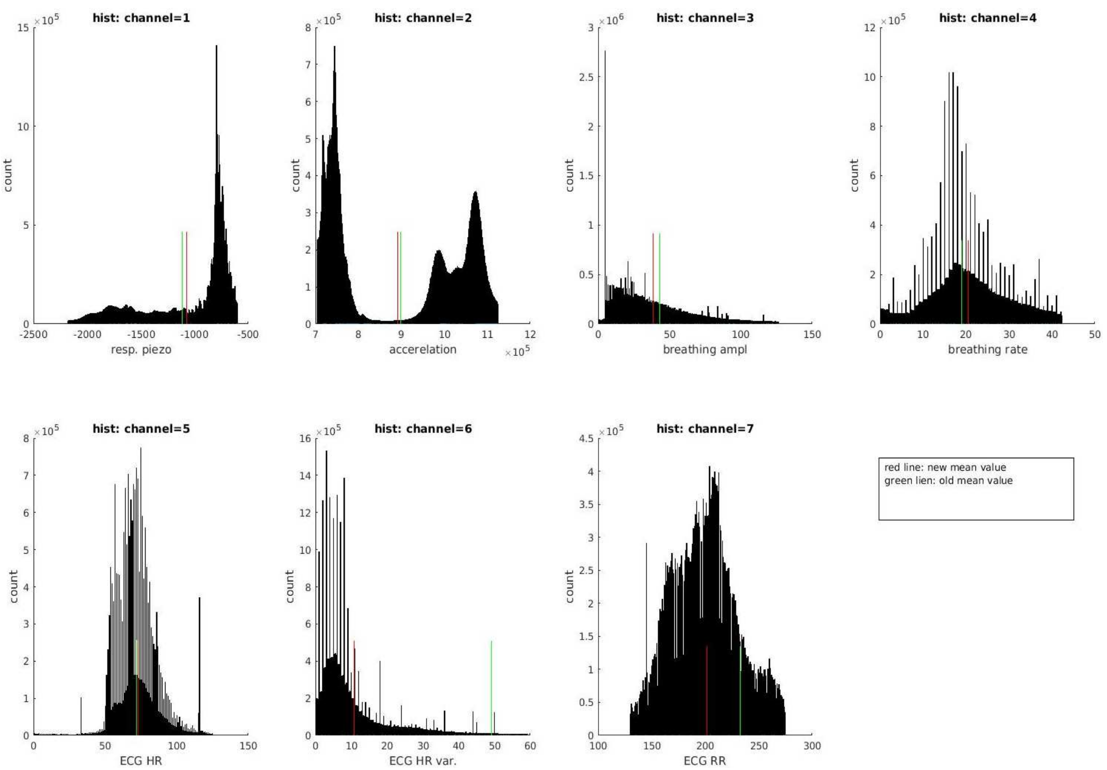


Figure 7 Data insights after outlier cleaning.

It can be observed from Figure 7 that after cleaning the outlier values, the probability distributions for each channel are more descriptive. We observe furthermore that for channels 1-5 the mean value of the recordings has not changed significantly (the new mean value is denoted by a red vertical line and the old mean value is denoted with a green vertical line). In contrast for channels 6-7 the change in the mean value of the recordings is greater, suggesting that a significant number of outliers affected our data.

Feature extraction: Finally, during the preprocessing we extract features from the recordings and use them for the correlation analysis with proxy outcomes (see section 4.3.2 for more details). The features that are extracted are statistical metrics such as the average, the standard deviation (std), the 5% and 95% percentiles, the most frequent value (mode), the kurtosis, the skewness, the energy, and the entropy of each recorded parameter (Heart rate, Respiration rate, Acceleration etc.).

3.3.2 GPS data

Associate GPS recordings with participants: The GPS data are being recorded in the smartphone by the GPS logger app and are then uploaded to the cloud. The initial version of the app did not associate the recording with a participant, thus we developed a process that handled this issue. For each recording uploaded to the cloud, a cross-check was performed using the information stored in the eCRF about the FrailSafe home sessions. The information in the eCRF contains the serial numbers of the devices given to each participant during the home sessions, and the dates of these sessions. This allowed us to map each recording to a participant.

Combine log file data: The GPS logger app uploads a zip file for each session, containing two log files; one in GPX format and one in TXT format. During the preprocessing we unzipped the initial file, and combined the fields of the extracted log files into a new one.

Mapping of each recording to the closest frailty status: The frailty status of each participant is measured periodically during the clinical examination. In between these examinations we collect measurements for the participants using the FrailSafe devices. In many of the analysis algorithms there is the need to have a frailty status associated with each of the recording, thus we performed a mapping procedure which assigns the closest frailty status to each recording.

Feature extraction: Each recording consists of consequent GPS coordinates and their corresponding timestamps. In order to calculate the distance between consequent GPS points, we converted the decimal degrees of the coordinates into radians and then applied the haversine formula. This formula calculates the great circle distance between two points on the earth. Followingly, we calculated the movement speed between consequent GPS points and used it to categorize participants' movement (vehicle, walking, standing). Finally the recordings are grouped into tracks, and aggregation functions are applied in order to extract additional features such as track duration, track length, track radius, max speed, average speed etc.

3.3.3 Game data

Distinction between dynamometer and touch screen*:* The developed games can be played either by the usage of the dynamometer, or by simply using the touch screen of the tablet. This affects the analysis, as sessions played using the touch screen record default values for the force of the participants. During the preprocessing we distinguished which games were played using the dynamometer and which ones using the touch screen of the tablet.

Mapping of each recording to the closest frailty status: The frailty status of each participant is measured periodically during the clinical examination. In between these examinations we collect measurements for the participants using the FrailSafe devices. In many of the analysis algorithms there is the need to have a frailty status associated with each of the recording, thus we performed a mapping procedure which assigns the closest frailty status to each recording.

Feature extraction: One of the recorded parameters in Red Wings game is the force applied by the participant on a specific time point. In order to evaluate the participant’s hand grip ability, features as the average and maximum force applied are important, as well as how long the participant can apply force in the dynamometer. Towards this direction we have created a new feature calculating the value of the surface generated during game time (calculation of the integral of the instantaneous force function developed by the participant during game time).

Finally, during the preprocessing we extract features from the recordings and use them for the correlation analysis with proxy outcomes (see section 4.3.2). The features that are extracted are statistical metrics such as the average, the standard deviation (std), the 5% and 95% percentiles, the most frequent value (mode), the kurtosis, the skewness, the energy, and the entropy of each recorded parameter (Force, Speed, Distance etc.).

4 Data Analysis

4.1 Group-wise histogram analysis of clinical data

Having collected all clinical data from the eCRF platform, the first step to start the data analysis was to summarize and describe the population's behavior. To that end, a univariate analysis was performed across three different bases:

1. Frailty status based analysis
2. Age based analysis
3. Gender based analysis

Regarding the frailty-based analysis, the participants were split in three categories, Frail, Pre-Frail, Non-Frail, according to the Fried categorization that had been performed at the beginning of the study. Similarly, for the gender-based analysis the population was grouped in male and female participants, while for the age-based analysis the participants were divided in three equally separated categories according to the distribution of the variable "birth year".

After splitting the dataset in the aforementioned categories, the univariate analysis was performed across the domains that had been defined by the clinicians in D2.3. The table with the updated domains is shown below:

Table 7: Clinical metrics. Domains investigated by the clinical evaluation.

Items	
Medical Domain (M)	Number of Comorbidities (M)
	Comorbidity's impact (M, P, s, ψ)
	Polymedication (M, p, c)
	Hospitalisations (M)
	Orthostatic hypotension (M, p)
	Visual impairment (M, S, p)
	Hearing impairment (m, S, c)

General Condition Domain (M, ψ)	Unintentional weight loss (M, ψ)
	Self-reported exhaustion (M, p, ψ)
Lifestyle domain (P, M, ψ,s)	Smoking (M, ψ, p, s)
	Alcohol (M, Ψ, S)
	Physical Activity (P, M, ψ, s)
Functional capacity domain (M, P, s, c, Ψ)	Basic Activities of Daily living (M, P, s, c, Ψ)
	Instrumental Activities of Daily Living (M, P, s, c, Ψ)
Physical Condition (P, m, c)	Balance (single foot standing) (P, m)
	Gait-related task speed* (P, c) (Timed Get Up and Go test)
	Gait - speed 4 m (P, m)
	Lower limb strength (P, m)
	Grip strength –dynamometer (P, m)
	Low physical activity (P, M, s, ψ)
	Falls (P, m, Ψ)
	Fractures (P, M)
Nutritionnal domain (M, Ψ, c, s)	Too low BMI (M, Ψ, p, c, s)
	Too high BMI (M, Ψ, P, c, s)
	High waist circumference (M, Ψ, P, c, s)
	Lean body mass (M, P, ψ)
	MNA screening and total (when applicable) score (M, Ψ, p, c, s)
Cognitive Domain (C, ψ, m, s)	MMSE scores (C, ψ, m)
	MoCA score (C, ψ, m)
	Subjective memory complaint (C, ψ, m, s)
	Natural language analysis (C, Ψ)
Psychological Domain (Ψ, S, c)	GDS-15*(Ψ, S, c)
	Self-rated anxiety (Ψ, S, c)
	Natural language analysis (C, Ψ)

Social Domain (S, Ψ, m)	Living conditions (S, Ψ, p, m)
	Leisure activities (S, Ψ, p, m)
	Membership of a club (S, Ψ, p, m)
	Number of visits and social interactions per week (S, Ψ, p)
	Number of telephone calls exchanged per week (S, ψ, m)
	Approximate time spent on phone per week (S, ψ, m)
	Approximate time spent on videoconference per week (S, ψ)
	Number of written messages sent by the participant per week (S, ψ, m, p)
Environmental Domain (S, P, m)	Subjective suitability of the housing environment according to participant's evaluation (S, P, m)
	Subjective suitability of the housing environment according to investigator's evaluation (S, P, m)
	Number of steps to access house (P, S, m)
Wellness domain (Ψ, S, M, P, c)	Quality of life self-rating (Ψ, S, M, P, c)
	Self-rated health status (M, Ψ)
	Self-assessed change since last year (M, ψ)
	Self-rated anxiety (Ψ, S, M, P, c)
	Self-rated pain (M, P, ψ)
Tags (reflecting impact of each item on each of the aspects of frailty) Physical/functional: P dominant, p recessive Medical: M dominant, m recessive Social: S dominant, s recessive Cognitive: C dominant, c recessive Psychological: Ψ dominant, ψ recessive	
Abbreviations: BMI: Body Mass Index, GDS-15: Geriatric Depression Scale 15 items, MMSE: Mini Mental State Examination, MNA: Mini Nutritional Assessment, MoCA: Montreal Cognitive Assessment.	

Data was first preprocessed, as described in previous section, and was further forwarded for the univariate analysis.After data preprocessing, some descriptive measures were extracted for each variable of each domain, including the total number of participants of each group, as well as the minimum, maximum and mean value of each numerical variable. Subsequently, we divided the values of the numerical variables into ranges that had mostly been defined by the clinicians in order for the data to be clinically meaningful, while in some cases the values' split was performed according to percentiles of the data distribution, or even randomly. For ordinal variables there was no need to implement such a procedure, as the values are already divided in categories.

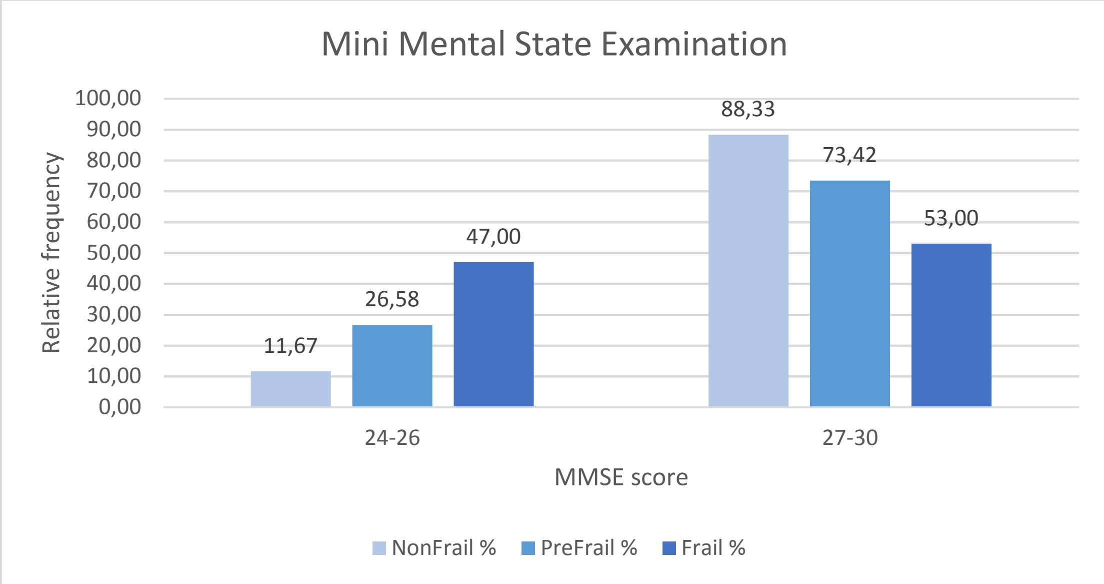
The next step was to count the number of the participants of each group (e.g. Frail, Non-Frail, or Female, Male) whose responses belonged in each of the categories defined by each variable's ranges. To that end, a table was extracted for each domain's variable that contained the variable's ranges and the number of the participants of each group that fell in each range (tables are summarized in the Appendix). For visualization purposes, the percentage of the number of participants that fell in each range was calculated, so that it would be easy to compare different groups with different number of participants each.

Two examples of the table described above are shown below, along with the corresponding chart.

The aforementioned analysis steps and the corresponding results were forwarded to the clinical team for further evaluation of the population. Observing the participants' summarized data from a clinical point of view provides the privilege of deciding in a relatively quick way whether a variable separates the population into groups well or not.

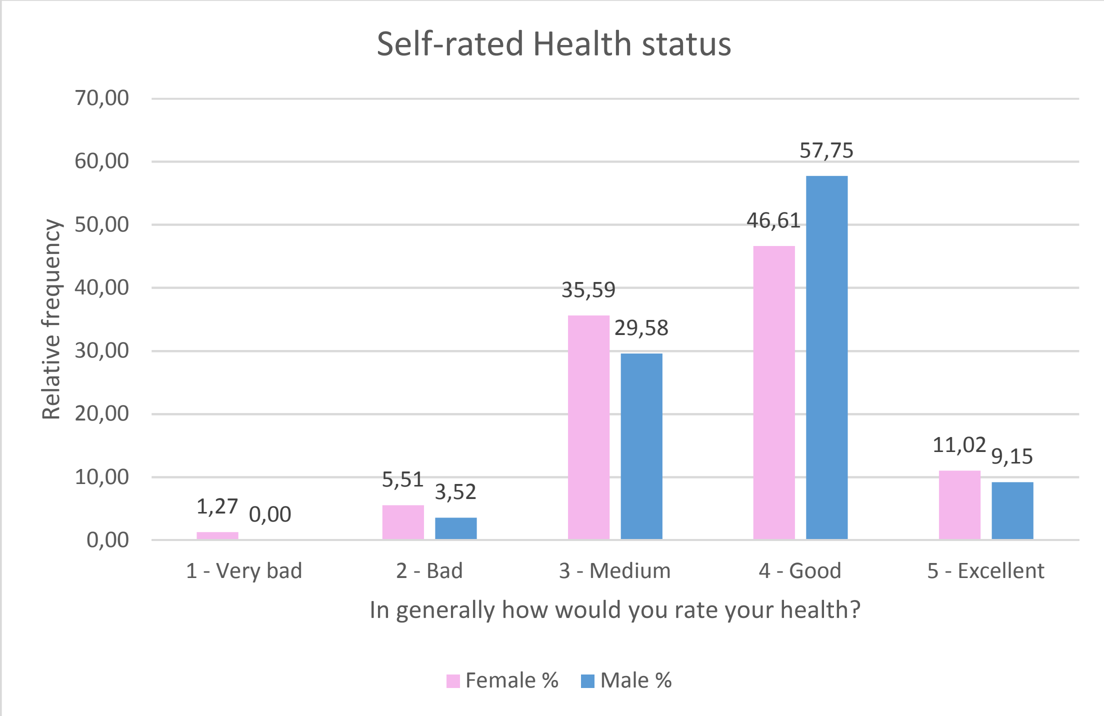
1. *Frailty-based univariate analysis example for numerical variable*

Cognitive Domain						
Mini Mental State Examination						
Ranges	**NonFrail**	**PreFrail**	**Frail**	**NonFrail %**	**PreFrail %**	**Frail %**
24-26	14	42	47	11,67	26,58	47,00
27-30	106	116	53	88,33	73,42	53,00



2. Gender-based univariate analysis example for ordinal variable

Wellness Domain				
Quality of life self-rating				
Row	**Female**	**Male**	**Female %**	**Male %**
1 - Very bad	3	0	1,27	0,00
2 - Bad	13	5	5,51	3,52
3 - Medium	84	42	35,59	29,58
4 - Good	110	82	46,61	57,75
5 - Excellent	26	13	11,02	9,15



With the completion of the 2nd clinical evaluation, the new measurements were incorporated in the analysis and specific charts for this evaluation's participants were re-produced following the aforementioned steps. The analysis was performed only with respect to frailty status. Two random chart examples are depicted below.

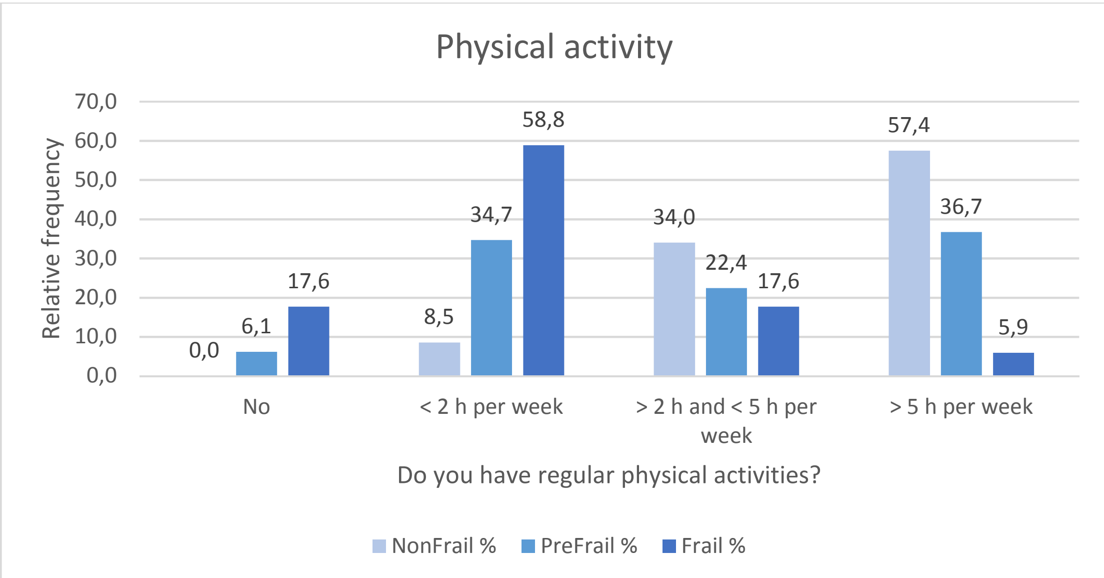


Figure 8: Lifestyle Domain example from 2nd clinical evaluation's participants.

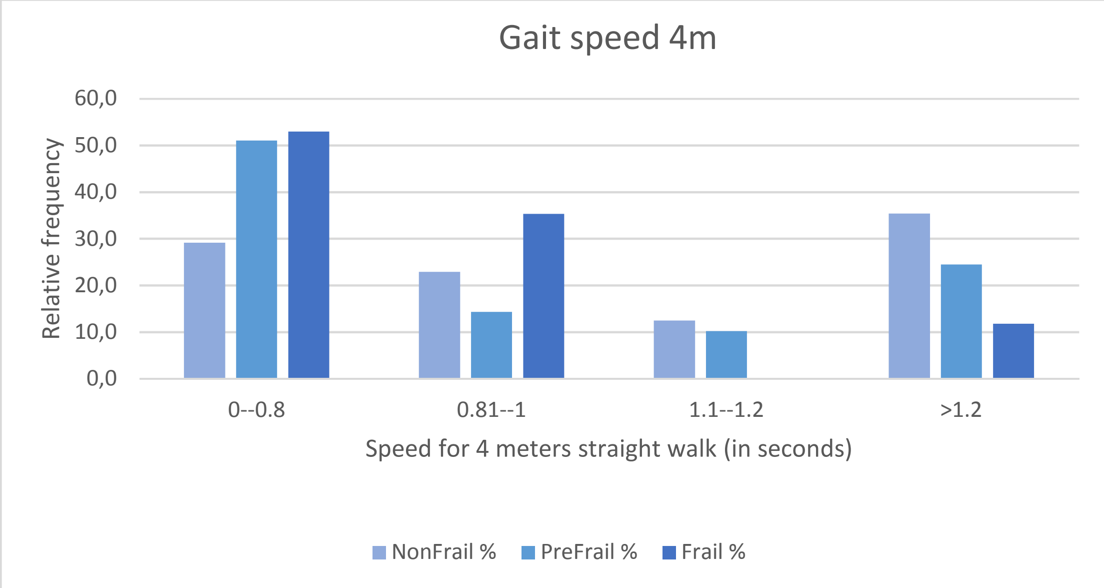


Figure 9: Physical Domain example from 2nd clinical evaluation's participants.

4.2 Multivariate statistical analysis of clinical data

After preprocessing and conversion to numerical data as described in section 3.1, the clinical measurements from eCRF were used for statistical analysis. Specifically, their predictive ability towards the development of a frailty index was examined. Two different frailty indexes (*FI*) were computed, one aiming to predict the discrete Fried classification score (FI_1) and one trying to estimate a continuous score as a linear combination of the 5 criteria related to Fried classification (FI_2). The ultimate goal is to investigate whether the proposed frailty indexes are more reliable predictors of frailty transition than standard classification scores. This hypothesis will be assessed in the evaluation phase.

In the following experiments, multiple sessions of the same individual are treated as independent measurements. The aim is to learn the variability of the population ignoring patients-specific transitions in order to build a generalized, and more robust than the Fried score, frailty index using the variables from the clinical evaluation. The dataset used in the analysis consists initially of 561 samples (sessions) and 177 variables. Samples and variables are reduced during the analysis following the preprocessing steps and exclusion criteria described in the Section 3.

4.2.1 Correlation of variables from clinical evaluation with Fried (FI_1score)

Lasso linear regression was performed to select a subset of variables and estimate their β coefficients, in order to build a predictive model having the best possible correlation with Fried's score.

$$FI_1 = \sum_{j=0}^{m} \beta_j\, x_j \approx Fried$$

where$x_j$are the variables from the clinical evaluation,m is the number of β_j coefficients (number of variables), β_0 is the intercept and $x_0 = 1$.

For this purpose, 5-fold cross validation was performed. Specifically, for each fold the model was built using the training set while performance was evaluated on the test set. A set of different values for λ (a parameter controlling the number of retained coefficients and thus the risk for overfitting) was tested and the one with the smallest fitting error on the test set was selected for each fold. The Spearman correlation between the Fried and the estimated score was calculated for each fold. The range of values is shown in the boxplot below.

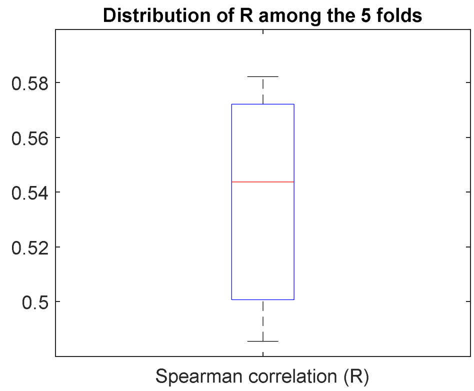


Figure 10: Spearman correlation between Fried and estimated scoreFI_1

The variables were sorted according to their level of significance (absolute value of beta) and the most frequently selected variables (across the 5 folds) were retained for the final model. The number of selected variables was such that the cumulative sum of ordered (by descending order of magnitude) beta coefficients was 90% of the total sum of beta coefficients.

Results

According to the frequency of selected variable and the previously defined threshold, 6 variables were retained and used to rebuild the *FI₁* score by linear regression. The selected variables are shown in the following Table together with the value of βcoefficients, sorted by decreasing significance (absolute value). Spearman's rank correlation coefficient (R) between Fried's score and the proposed *FI₁* score was 0.57.

Table 8: Clinical variables constituting the frailty score FI_1.

Clinical variables	**β (for *FI₁*-score)**	**Selection frequency**
Intercept (β_0)	*0.326*	
gait_speed_4m	0.408	*100%*
raise_chair_time	0.334	*100%*
depression_total_score	0.295	*100%*
activity_regular	0.282	*100%*
pain_perception	0.174	*100%*
balance_single	0.132	80%

4.2.2 Calculating the optimal combination of Fried-related criteria (*FI₂*score)

A similar analysis (as above) was performed but this time the variable to be predicted is a continuous frailty index (let's denote itwith Y) expressed as a linear combination of the 5 criteria related to Fried score:

$$Y = \sum_{k=1}^{5} \alpha_k f_k, \text{ subject to } \sum_{k=1}^{5} \alpha_k = 1$$

where $f_k \in$ {involuntary weight loss, slow walking speed, poor handgrip quality, reported exhaustion, low physical activity}. We want to calculate again a score that takes into account all clinical variables x_j

$$FI_2 = \sum_{j=0}^{m} \beta_j x_j \approx Y$$

where m is the number of β_j coefficients (number of variables), β_0 is the intercept and $x_0 = 1$.In this case, we want to estimate jointly the β-coefficients and α-coefficients, such that for every record x_i, $i = 1, \dots, n$, where n is number of records,

$$\sum_{j=0}^{m} \beta_j x_{ij} \approx \sum_{k=1}^{5} \alpha_k f_{ik},$$

$$\text{subject to } \sum_{k=1}^{5} \alpha_k = 1$$

$$\text{and } \beta \text{ sparse,}$$

In a vector form, if $a = \begin{bmatrix} a_1 \\ \vdots \\ a_5 \end{bmatrix}$ and $\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_m \end{bmatrix}$, we seek for $p = \begin{bmatrix} a \\ \beta \end{bmatrix} \in \mathbb{R}^{m+6}$. Given a dataset $X_{n\times(m+1)} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{nm} \end{bmatrix}$ that should be mapped to frailty criteria

$F = \begin{bmatrix} fx_{11} & \cdots & f_{15} \\ \vdots & \ddots & \vdots \\ f_{n1} & \cdots & f_{n5} \end{bmatrix}$, this problem can be expressed as a generalized Lasso regression under equality constraints:

$$p = argmin\ \frac{1}{2}\|Wp\|_2^2 + \lambda\|Dp\|_1\ ,$$

$$\text{subject to } Ap = 1$$

where $W = [F \quad -X] \in \mathbb{R}^{n\times(m+6)}$. $D \in \mathbb{R}^{n\times(m+6)}$ is a penalty matrix used to enforce sparsity constraints only on β , and is $D = [\emptyset \quad I]$, where $I \in \mathbb{R}^{(m+1)\times(m+1)}$ is the identity matrix and $\emptyset \in \mathbb{R}^{(m+1)\times 5}$ is a matrix filled with zeros. Various algorithms to solve the constrained Lasso, including quadratic programming, were investigated following the work in (Gaines2016) and finally the alternating direction method of multipliers (ADMM) was used based on the Gurobi (http://www.gurobi.com) solver.

Results

The coefficients (weights) for the calculated prediction models according to the 2nd approach are shown by decreasing significance (absolute value) in the Table 9.

Table 9: Clinical variables constituting the Y-score and the frailty score FI_2.

Fried's criteria	α (for *Y*-score)
Intercept (β_0)	*0*
weight_loss	0.3876
low_physical_activity	0.3336
exhaustion_score	0.1150
gait_speed_slower	0.0923
grip_strength_abnormal	0.0714

Clinical variables	β (for FI_2-score)
Intercept (β_0)	*-0.20*
depression_total_score	-0.1351
pain_perception	-0.1155
activity_regular	0.1141
screening_score	0.0949
katz_index	-0.0578
falls_one_year	-0.0548
raise_chair_time	-0.0517
balance_single	0.05
gait_speed_4m	-0.0387
leisure_club	0.0365
gait_optional_binary	-0.0362
living_alone	-0.0163
health_rate_comparison	0.0139
mmse_total_score	0.0098
smoking	-0.0083
age	-0.0056
memory_complain	-0.0038

The Spearman correlation between the Y-score (constructed by the 5 Fried criteria) and the Fried score was 0.95, while the Spearman correlation between the FI_1-score (constructed by 17 selected clinical variables) and the Fried score was 0.56.

4.2.3 Assessment of frailty transition using the different scores

The frailty scores were constructed using 415 sessions from 292 subjects in total (after discarding sessions with missing values). Out of the 292 subjects, 81 had multiple sessions (2 or 3). For these subjects, we calculated the change in the value of all four indices (FI_1, FI_2, Y, Fried*)* over consecutive sessions. The figure below shows the histograms of the transition using each of the frailty indices. Negative values mean that the frailty decreased, thus the health condition of the subject improved, and correspondingly positive values mean that frailty increased.

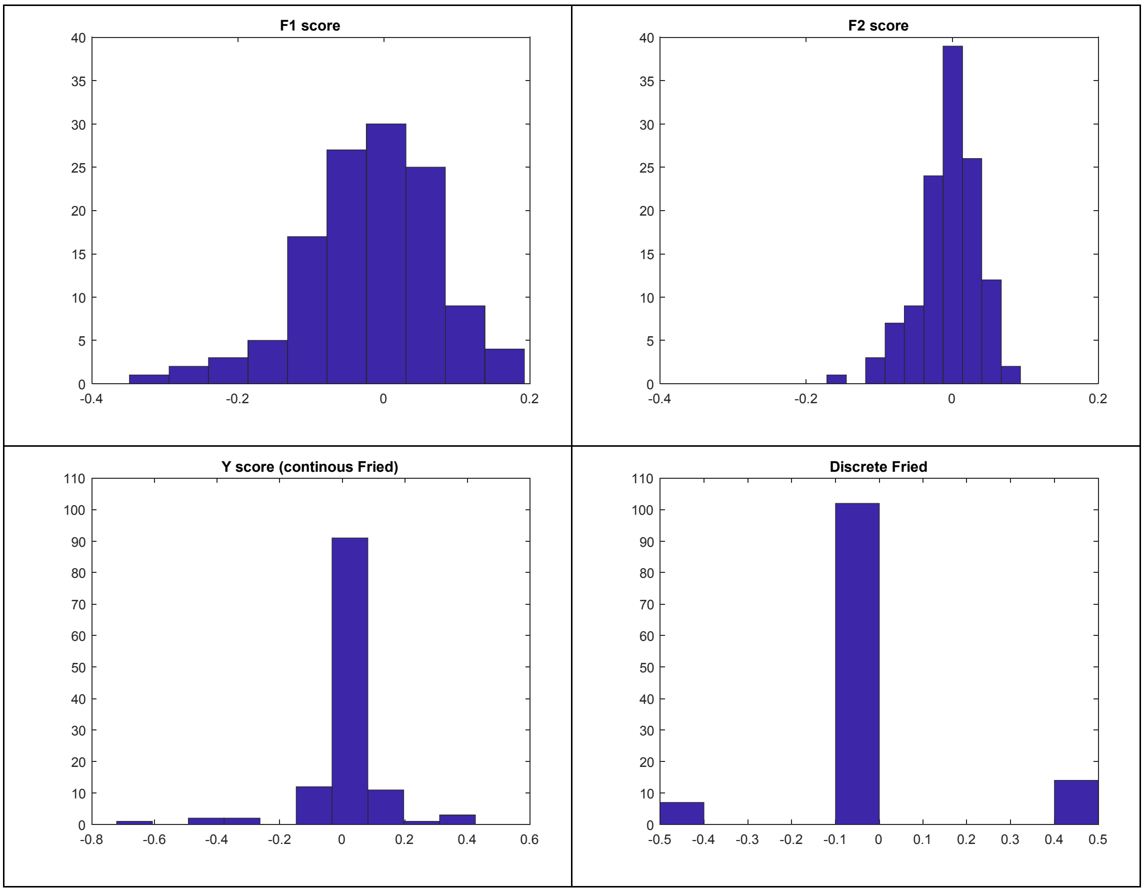


Figure 11: Histograms of change of score values over 2 consecutive sessions for the two estimated frailty indices, $FI_1$and FI_2 in the 1st row, and the estimated continuous Fried score(Y-score) in the 2nd row on the left and the discrete Fried score on the right.

The histograms of transition values show that the calculated frailty indices FI_1 and FI_2 have more times negative value than positive indicating that health improvements were more often than health decays. The opposite was observed with the continuous and discrete Fried score.

4.3 Analysis of data from sensors (FrailSafe devices)

4.3.1 Activity Classification

A first approach on the problem of classification of ADL was previously described and discussed in the preliminary version of this deliverable (D4.1). In the proposed methodology, accelerometer and gyroscope signals were preprocessed, and feature extraction was performed for the purpose of ADL classification using SMO (Sequential Minimal Optimization) algorithm. The proposed approach was based on former work conducted in the field, and validation of the methodology was performed on a public dataset consisting of young adults. A series of different implementations and development of optimized variations of the firstly introduced methodology have been conducted during the past six months, including creation of different training sets, application of the proposed scheme using data from volunteers of different ages (young adults or FrailSafe participants), and from different kinds of devices (WWS and WWBS), and corrective actions on the performance of the algorithms for optimization purposes (including thresholds and rules for selecting the suitable version of the classification scheme each time a new instance is given as input). The steps that were followed during this phase are described in this section.

Implementation of proposed methodology

The first implementation of the previously described methodology using a FrailSafe device was conducted using 3-axial sensor data from three young participants aged 23-33 years old. Each participant performed the following ADLs: sitting, standing, lying, walking, walking upstairs, walking downstairs, while wearing the WWS device, and repeated these activities two more times (3 trials for each of the 3 subjects). The accelerometer signals that were used for feature extraction and classification were sampled at 25Hz. The classification model was constructed after splitting the subjects into training and test set, using Weka’s Random Forest classification algorithm, but no feature selection was performed.

Although the model performed well when applied to the test set, no evaluation could be performed on FrailSafe participants, since no annotated data had been acquired. This led to the decision of applying an ADL protocol to some participants in all clinical centers, in order to evaluate the constructed model’s performance when applied to FrailSafe data. The participants performed the activities described previously while wearing the WWBS, and records were manually annotated by the nurses. An example of a participant’s accelerometer measurements with respect to the activities performed is depicted in Figure 12.

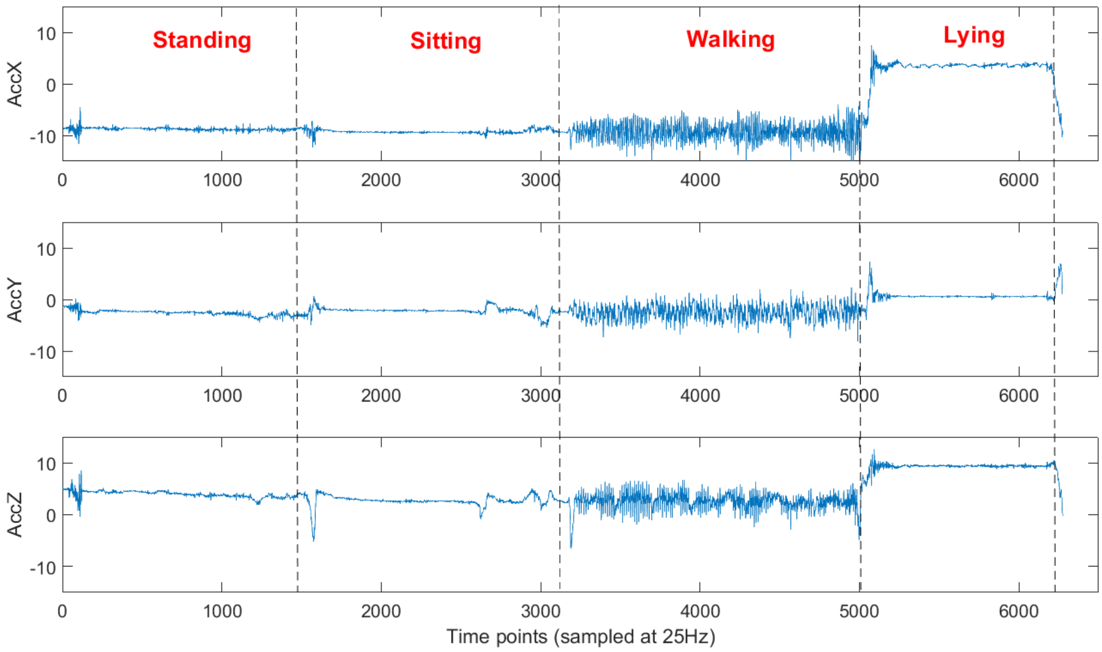


Figure 12: Acceleration signals while performing different ADLs.

When the classification model was applied to FrailSafe annotated data, the accuracy was disappointingly low. This result was not surprising, most importantly because of the difference of age between the training subjects (young adults) and the FrailSafe subjects (older people), but also because the FrailSafe subjects used the WWBS device instead of WWS.

To that end, a new model was constructed based on FrailSafe data, while a series of correcting actions and decisions had to be made in order to adjust the methods to the nature of data coming from older people. The steps that were followed towards constructing older people-oriented models are described right after.

Model re-building and optimization

For the purpose of building the new training model, 8 subjects were used to perform 4-fold cross-validation and standardize the classification parameters, while the rest were left for validation (4 subjects from Nancy, 3 from Materia and 3 from UoP). All 8 subjects came from Nancy clinical center, because the RUSA device appeared to have been consistently placed during the ADL protocol performance (axis X was identified as the vertical axis in all subjects).

Raw 3-axial signals from the accelerometer of WWBS's RUSA device, sampled at 25Hz, were preprocessed as previously reported and sampled in fixed-width sliding windows $W_i, 1 \leq i \leq I$ (frames) of 2.56 sec and 50% overlap. The preprocessed signals were further used for feature extraction. A total of 254 spectral and time-related features were extracted from each frame, and a SVM classifier with RBF kernel from the libSVM package was used to train the classification model.

Although 6 ADL classes were initially taken into account from the annotated data, classes *siting* and *standing* were merged into one class, as well as *walking upstairs* and *walking downstairs*. This decision was for the purpose of achieving higher classification accuracy, since previously reported work suggests that these classes are not easily separable. Additionally, the time instances corresponding to the first five seconds of the beginning of each activity, were automatically annotated as “transition state”, to indicate the time needed to switch between two different activities. To that end, the final classes were: *sit/stand, lie, walk, walk upstairs/downstairs, transition state.* The number of samples that corresponded to each class was different from class to class, a problem known as *imbalanced classes*, leading to the solution of using weights in the classification function.

Concerning the SVM classification parameters, since RBF kernel was used, parameters *C*and *gamma* were explored in reasonable ranges, in order to achieve optimum classification accuracy on each fold’s test set. Furthermore, feature selection was applied on the training set using the Relief-F algorithm for dimensionality reduction and removal of irrelevant features. The best cross-validation accuracy was achieved using only 10 features from the initial 254-dimensioned feature vector. These features are depicted in Table 10.

Table 10: Most significant features.

Ranking	Feauture
1	tBodyAcc_std_Z
2	tBodyAcc_min_Z
3	tGravityAcc_max_X
4	tGravityAcc_min_Z
5	tGravityAcc_std_Z
6	tBodyAcc_std_X
7	tGravityAcc_energy_X
8	tBodyAcc_sma
9	tBodyAcc_correlation_XZ
10	tGravityAcc_energy_Z

Considering that switching between different activities occurs with lower frequency than the frames’ length (1.25 second), we incorporated a rule just before computing the classification accuracy. More specifically, after performing the classification on each fold’s test set, each frame’s final label was defined as the majority of the three neighbor-frames’ labels before and after the current frame. This way the decision for a specific frame’s label is more stable, leading to more robust results and improving the classification accuracy.

The training model after optimizing the aforementioned parameters achieved 89.46% accuracy (computed as the number of correctly classified instances to the total number of instances on each fold and extracting the mean of all folds). The constructed model was then applied to the test data for evaluation purposes. When applied to subjects coming from Materia clinical center, the classification accuracy was 85.81%, while test subjects from Nancy center reached 83.5%. Nevertheless, UoP's test subjects' accuracy was only 31.5%. Although many factors could be hiding behind this result, we decided to further investigate the characteristics of these specific records. Looking closer at the raw accelerometer signals, we realized that the RUSA device appeared to have been placed differently than the other clinical centers' subjects, causing rotation of axes and leading to a great number of misclassified instances.

Table 11: ADL classification accuracy (for correct placement of sensors)

Training accuracy	**Testing accuracy across clinical centers**	
	Nancy	Materia
89.46%	83.5%	85.81%

Resolving the rotation-of-axes issue

After a thorough investigation of how a mis-oriented device's data could be mapped to a differently oriented device's records, we concluded that orientation correction could not be performed automatically, and we should move in a different direction to resolve this issue. For that purpose, we built a new classification model using the same set of subjects for cross validation, in which each triplet of features extracted from x, y and z axis, was reduced to only one feature computed as the mean of the three axes' corresponding features. This approach actually removes the concept of orientation from the data, resulting in a rotation-invariant features. The reduced classification model was optimized using different SVM parameters than the previous model, while from the 96 features that were now extracted, a total of 40 features were selected through the feature selection process. This model's cross-validation classification accuracy was **75.56%**. Classifying the UoP subjects using the reduced model increased the classification accuracy by 27.2%. The significant increase in accuracy indicates justifies our idea of using rotation-invariant features. The still quite low accuracy on the other hand indicates that the orientation of axes is a major factor for correct classification and that the consistent placement of the devices is important.

Since the detection of a device's orientation is not a trivial problem, another issue we had to deal with was the automatic selection of one of the two models, depending on whether the RUSA device had been misplaced or not. The problem was addressed by learning the distribution of measurements in the reference space. A two-fold rule was introduced for that purpose. The first part of the rule relates to detecting whether the vertical axis is the same as in the case of the training subjects (axis X). This was achieved by computing the 80th percentile of the training subjects' vertical axis' measurements, and then checking if the incoming data's axis' X corresponding percentile falls in the same range or not. If so, then the axis-dependent model is selected for classification, otherwise the second part of the rule is examined. This part concerns the overall distribution of measurements along the three axes. More specifically, the histogram of each of the training subjects is extracted and the distance between all pairs of subjects' histograms are computed using a 3-D version of the Kolmogorov-Smirov distance. This distance matrix serves as a reference point for the incoming data's histograms function. The histogram distance between each test subject's measurements and the training subjects' histogram is then calculated, and if it is smaller than the within-training-subjects maximum distance, then the axis-dependent model is selected, otherwise the classification is performed using the axis-invariant scheme.

Comparison of WWBS and WWS measurements

To explore whether any differences occurred between WWBS and WWS originated data, two young volunteers were asked to perform the ADL protocol while simultaneously wearing both the WWBS and the WWS devices. The corresponding measurements were given as input for ADL classification, from which it was revealed that a scaling difference occurred between the two types of measurements. To address this issue, an automated scaling should be performed in reference to the WWBS measurements. Under this premise, we implemented a baseline correction procedure by finding parts of the incoming signal with small standard deviation, and calculating the mean value of these parts (baseline). The corresponding baseline of all axes was calculated for the WWBS training data as well. Followingly, the correction was performed by subtracting the WWS baseline from the signal and then adding the WWBS baseline value.

Meanwhile, WWS measurements were acquired from two FrailSafe participants of Materia while performing the ADL protocol. To evaluate the implemented baseline correction method, we first calculated the classification accuracy of the four subjects (two volunteers and two FrailSafe participants) without the correction. The mean accuracy of these subjects was 37.5%, while after applying the baseline correction it was raised to 61.7%.

Considering the lack of annotated data from the WWS device, enough to train a new classification model, the baseline correction is an efficient solution since it enables us to use an already built model with an incorporated rule for manipulating WWS data.

4.3.2 Correlation of FrailSafe device recordings with proxy outcomes

The recordings from the vest/strap (ECG and IMUs) as well as from the FrailSafe games were used for statistical analysis. More specifically, the ability of these measurements to predict the change of the clinical metrics, defined as proxy outcomes in D2.1, was examined. The rest of the variables will be included in deliverable D4.17 upon completion of the corresponding deliverables that were running in parallel with D4.2. Specifically, variables form the GPS analysis based on the wandering evaluation will be incorporated upon completion of deliverable D4.15, while variables from social domain will be combined and analyzed with the rest upon completion of the deliverables about sensing and processing social media (D4.9, D4.11, D4.13).

As mentioned in D2.1, proxy outcomes are based on the data from repeated clinical evaluations and described by the differences (delta) in clinical parameters that capture the status of separate human functions:

- MMSE (MMSE total score) and MoCa (cognititve total score) - **cognitive function**
- Gait speed (gait speed 4m) - **physical function**
- GDS (depression total score) - **psychological status**
- Weight loss - **general health**
- Health rate – **health status self-assessment**

From the aforementioned parameters only those that are numerical and are considered to have a continuous evolution in time (MMSE total score, cognitive total score, depression total score, health rate, gait speed 4m) were included in the statistical analysis that was performed. A binary variable, such as weight loss, cannot be well predicted by regression models.

The main goal of the analysis is to investigate whether variables extracted from the FrailSafe devices can be used as predictors of the proxy outcomes. This is done by examining their correlation with each of the proxy outcomes separately and then combine the most correlated ones in a unified predictive model. Hence, three different steps were followed:

1. Examination of the correlation of vest/strap recordings with each of the proxy outcomes. To this end, we used the strap/vest measured entities to extract features for the statistical analysis by calculating histogram-based features for each of the measured parameters as shown below. The mode corresponds to the peak of the histogram, indicating the most frequently encountered value. Kurtosis characterizes the relative peakedness or flatness of the histogram, skewness is a measure of the distribution asymmetry and indicates the direction towards which the distribution is shifted, while energy and entropy are statistical measures of randomness and uncertainty.

Table 12: Measurements from vest/strap and extracted histogram-based variables

Parameter	Statistical metrics
Heart Rate	*average,*

Respiration Rate	*standard deviation (std), 5% percentile, 95% percentile, most frequent value (mode), kurtosis, skewness, energy, entropy*
Heart Rate Variability	
Breathing Rate	
Breathing Amplitude	
Acceleration	

2. Examination of the correlation of games recordings with each of the proxy outcomes. For this reason, we used summarized games data from the Virtual Patient Model (VPM) as well as other features which came as a result of calculation of histogram features from the raw values of the recorded game entities.

In both steps (1) and (2) the statistical metrics were calculated by aggregating data collected for each participant per session to compute the evolution of the participants' health status regarding the measured parameters in day-by-day basis.

Table 13: VPM game parameters and extracted histogram-based variables

VPM Game parameters
Max force
Average max force
Average and max endurance
Average and max score
Average and max game duration

Game Parameter	Statistical metrics
Height	*average, standard deviation (std), 5% percentile, 95% percentile, most frequent value (mode), skewness, kurtosis, energy, entropy*
Distance	
Speed	
Lives	
Force	

3. Examination of the correlation of combined vest/strap and games recordings with each of the proxy outcomes.

In steps (1) and (2) lasso linear regression was performed to select a subset of variables and estimate their β coefficients, aiming at building predictive models having the best possible correlations with each of the proxy outcomes separately. The analysis was performed five times (one for each of the examined proxy outcomes) and each time a set of different values for λ, a parameter which controls the number of retained coefficients and thus the risk for overfitting, was tested and the one with the smallest fitting error was selected.

In step (1) 203 recordings from 106 participants constituted the input dataset of the regression model. This dataset resulted from the exclusion of data from: (a) participants which left the study (e.g. due to death or consent withdrawal) and (b) participants for which there were missing values for the clinical entities defined as proxy outcomes until 8/12/2017.

In step (2) recordings from the “Red Wings” game were used due to the plethora of the collected data, compared to the low number of observations regarding the “ForceAnalyzer” and the “Virtual Supermarket” games. As each game evaluates different parameters for the participants, combining the different games would result in a small subset of participants which have played all games and thus the analysis results would not be of significant importance. Thus, a separate analysis of data from “Virtual Supermarket” game was performed and is described later at the end of this section.

Another crucial factor that had influence on the analysis process was the fact that some of the participants had played the “Red Wings” game using the touch screen and not the dynamometer in some of the game sessions. For this reason and in an effort to use all the available data, two models were built, for each of the two different datasets (with and without the dynamometer).

The total number of participants for which “Red Wings” game data had been collected until 8/12/2017 and simultaneously there were no missing data for the clinical parameters defined as proxy outcomes were 156. The total number of recordings for them was 959. For the calculation of this number, data from participants which have been excluded from FrailSafe study were not taken into account. The first dataset included only data extracted from game sessions that where played with the use of the dynamometer, so that the force-related variables could be used as possible predictive variables by the model. This dataset was consisted of 833 recordings from 147 unique participants. The second dataset consisted of recordings from every game session, having the force-related variables excluded. Hence, it consisted of all 959 recordings from 156 participants.

In both cases, (1) and (2), the prediction models were calculated by fitting all the data at once.

Mapping sparse measurements to continuous scores

For the majority of the participants there were only one or two recordings for the proxy parameters, and for this reason it was necessary to estimate their values in intermediate time points for which there were sensor recordings. Hence, the empty cells in the proxy outcomes vectors were filled by performing linear interpolation.

This allowed to have synchronized measurements for both input and output variables.

Fusion of variables

After the determination of the β-coefficients of the vest/strap and games variables in steps (1), (2), the results were used in a correlation analysis of their combination with each of the proxy outcomes in step (3). More specifically, the combination took place by:

a) selecting the most significant vest/strap and games variables (by decreased order of magnitude for the coefficients) which had a cumulative percentage of 90% in steps (1) and (2) respectively. By this means, it is ensured that only the most related vest/strap and game features are examined by the lasso model as predictive entities.
b) mapping the game recordings to the vest/strap recording of the nearest date, as in most cases there were no simultaneous recordings from both sources. Through this mapping the input dataset of the regression model has certainly no empty values.

As a result of all the above, a correlation analysis was performed in step 3 with two different datasets as input to the predictive model: (1) a dataset consisted of combined vest/strap and games variables without the force-related variables, (2) a dataset consisted of combined vest/strap and games variables including force related variables. Since the dynamometer was used only in some of the game sessions the second dataset had fewer recordings. More specifically, the size of the two datasets was 420 and 360 recordings from 71 and 68 participants, respectively.

We performed lasso regression for each of the proxy outcomes by calculating the predictive model using bootstrapping with 5 repetitions. At each repetition, 70% of the records were randomly selected for lasso regression and the resulting model was applied on the remaining 30%, i.e. the 70% were the training subset and the 30% the validation subset. Finally, the correlation of the predicted values with the real ones was calculated for both training and validation test. At the end of all the repetitions, the average and the median values of the correlations were calculated.

Results

As a criterion of the quality of the regression results, the Spearman's rank correlation coefficient (R) between the predicted proxy outcome values and the proxy outcome vector with the real values was used. We classified the regression outcomes in three categories based on the Spearman's correlation index: (1) Low correlation (0-49%), (2) moderate correlation (50-65%), (3) high correlation (66-100%). To consider a result noticeable, the R in the validation test should be included in the moderate or high-correlation categories.

The results in the validation subsets for most of the proxy parameters are included in the low-correlation category for both datasets. The only significant correlation result was given by the model that was built for the “gait speed 4m” proxy parameter using the dataset that contains force parameters. As shown in the table below, the average R in the validation subset is 0.541 and can be considered as a moderate-correlation result.

Proxy	Mean_train_corr	Mean_test_corr	Median_train_corr	Median_test_corr	Std_train_corr	Std_test_corr
Gait_speed_4m	*0.754*	*0.541*	*0.760*	*0.545*	*0.036*	*0.079*

The most frequently selected variables among the 5 bootstrap repetitions are shown next.

FrailSafe devices variables	*Selection Frequency*
Acceleration: skewness	*1*
Force: percentile 95%	*0.8*
Heart rate: mode	*0.6*
Heart rate: percentile 5%	*0.6*
Heart rate variability: skewness	*0.6*
Breathing rate: skewness	*0.4*
Height: mode	*0.4*
Heart rate: std	*0.4*

Analysis of“Virtual Super Market” data

The virtual 3D environment was projected through a desktop Virtual Reality (VR) application (low immersion, no use of special hardware) on a tablet computer. The “Virtual Super Market” (VSM) mimics an everyday shopping experience while it monitors user’s behavior by measuring key-characteristic parameters like the kind and quantity of products purchased. In this game-like environment, users are asked to do their shopping based on a predefined shopping list which contains a number of randomly selected items. After each game session the player’s performance is reported back to a server for further statistical analysis.

In total, the body of participants that played the VSM consisted of eighty (N = 80) elderly people, 78.08 years old in average (SD=5.479). From those, 39 were found to be non-frail (NF), 30 people were found in a pre-frail state (PF) and 11 were frail (FR).

The responses of the participants in the scales (questionnaires) can be found on *Table 3* below along with the results of the non-parametric (Chi-Square) test performed over the three user groups. From those results, only the MNA and the GDS were found to be statistically significant (p<.001), while the results of the MoCA were found marginally statistically significant with p=.058.

Table 14: Neuropsychological test results, daily living and life style parameters for the group of participants playing the VSM game

	Parameters	Groups of participants			Chi-Square
		Non-Frail	*Pre-Frail*	*Frail*	
Social	Social Calls	19.59 (SD=38.706)	12.77 (SD=13.302)	15.36 (SD=9.447)	$X^2(2)=1.765$ (p=.414)
	Social Visits	11.85 (SD=36.167)	4.53 (SD=4.725)	2.55 (SD=1.864)	$X^2(2)=3.288$ (p=.193)
Life style	Activities of Daily Living	29.794 (SD=24.889)	33.733 (SD=24.735)	19.272 (SD=22.948)	$X^2(2)=2.816$ (p=.245)
	Instrumental Activities of Daily Living	29.384 (SD=2.843)	29.333 (SD=3.077)	28.181 (SD=5.564)	$X^2(2)=0.193$ (p=.908)
	Mini Nutritional Assessment	13.102 (SD=1.187)	13.200 (SD=1.030)	10.181 (SD=2.400)	$X^2(2)=15.745$ (p<.001)
Cognition	Montreal Cognitive Assessment Scale (MoCA)	26.435 (SD=2.425)	25.7333 (SD=3.453)	24.181 (SD=2.676)	$X^2(2)=5.710$ (p=.058)
	Mini Mental State Examination Scale (MMSE)	28.179 (SD=1.636)	27.833 (SD=1.261)	27.727 (SD=1.954)	$X^2(2)=1.944$ (p=.378)
Depression	The Geriatric Depression Scale (GDS) – version of 15 items	1.9744 (SD=1.842)	2.333 (SD=2.233)	6.272 (SD=2.866)	$X^2(2)=18.391$ (p<.001)

Overall, daily living and social interactions (indicated with calls and visits) seem not to be affected by the presence of frailty. But nutrition was negatively affected and people with frailty or at pre-frailty stage were at risk of malnutrition or found to be malnourished. The cognition assessment results indicated that MoCA can weakly separate the three user groups, but MMSE cannot. In addition to the above, the depression test results indicated by the GDS test were different in the three user groups and the greater depressive symptomatology was related with the presence of the frailty condition.

Performance Metrics

The scoring parameters used to describe the performance of the players refer to the degree they achieved the game objectives and also to the errors they may have made during playtime. The user's performance metrics are described below in more details:

Score (overallScore): This is a numeric descriptor of the overall performance of the player during a single game session. It was calculated by the combination of the other two scores explained below.

Duration (overallDuration): This is the overall time, measured in seconds) the player needed to complete the game goals (search, find and purchase the products in the list). The time needed to pay in the cashier is included.

Selected item types (itemTypesScore): This is a subscore used to measure the portion of the product types selected by the player to the number of product types (given by the game-controller) in the list of products.

Selected items (itemQuantitiesScore): This is a subscore used to measure the number of the items selected by the player to the number of items in the given list.

Errors related to the product types (itemTypesError): This is the error rate of the player regarding the types of products purchased.

Errors related to the number of products (itemQuantitiesError): This is the error rate of the player regarding the number of products purchased.

Payment errors (moneyError): This is the error rate of the player regarding the payment process (bill selection and total amount paid).

Results

The statistical analysis was performed using the SPSS software (IBM), ver. 19. Unlike the age and the years of education, the scoring variables were found not normally distributed among the groups of participants according to the Shapiro-Wilk test of normality, and thus non-parametric tests were used to test the hypotheses of this study. The test results of the users(players) who participated in this study are numerically presented in Table 14but the actual goal was to compare the mean values between groups in order to confirm or reject the main hypothesis that people in the frail group achieved lower scores and had higher error rates in comparison to the healthier participants.

Series of rank-based nonparametric tests (Kruskal-Wallis H test) were performed on the data collected by the log files of the players. As seen in the last column of Table 15**,** statistically significant differences were found in the overall game score (significant at the .001 level). On the other hand, the hypothesis that the game duration was equal in the groups of participants was rejected with $p > .05$.

Table 15: VSM game results

	Parameters	Groups of participants			All cases
		CG	*mCFG*	*sCFG*	
Success rates	Overall Score	87.624 (SD=29.092)	76.289 (SD=41.499)	55.570 (SD=46.228)	78.966 (SD=37.759)
	Game Duration (in min)	41.886 (SD=21.510)	42.247 (SD=34.444)	32.105 (SD=18.863)	40.702 (SD=26.449)
	ItemTypesScore (%)	87.820 (SD=29.172)	75.833 (SD=41.263)	59.090 (SD=47.792)	79.375 (37.689)
	ItemQuantitiesScore (%)	87.428 (SD=29.082)	76.746 (SD=41.905)	52.050 (SD=45.878)	78.558 (SD=38.155)
Errorr rates	ItemTypesError (%)	-.1346 (SD=.308)	-.258 (SD=.422)	-.454 (SD=.485)	-.225 (SD=.390)
	ItemQuantitiesError (%)	-.088 (SD=.287)	-.214 (SD=.417)	-.571 (SD=.534)	-.199 (SD=.393)
	MoneyError (%)	-.171 (SD=.382)	-.166 (SD=380)	-.285 (SD=.487)	-.181 (SD=.388)

The rest of the success rate variables showed also statistically significant differences:the itemTypesScore was significant at the .05 level and the itemQuantitiesScorewas significant at the .001 level. The rest of the scoring variables related to the error rates showed differences which were significant at the .05 level for the itemTypesError and the itemQuantitiesError, but not for the moneyError.

In a short gender analysis study using the Mann-Whitney U test, it was found no statistically significant differences in performance metrics, namely the overallScore (U = 569, p = .885), itemTypesScore (U = 546.5, p = .634), itemQuantitiesScore(U = 562.5, p = .812), itemTypesError (U = 560, p = .779), itemQuantitiesError (U = 423.5, p = .992), moneyError (U = 390.5, p = .837).

Correlations with other neuropsychological and life style test results

Bivariate correlations tests were performed with the Spearman correlation coefficient calculation. A weak relation was found between depression test score (GDS) and social visits with r_s = -.277 (p = .001), but surprisingly not with social calls r_s = -.058 (p = .479).

Regarding the two neuropsychological tests MoCA and MMSE a moderate correlation strength was found with $r_s = .459$ ($p < .001$). Such a relationship was expected based on the fact that those two screening tools share a similar interest.

The correlations between the game performance metrics could also reveal the degree of the internal cohesion of the test. Indeed, the itemTypesScore was very strongly correlated to the itemQuantitiesScore with $r_s = .967$ ($p < .001$). Similarly, the two types of errors which are related to the product selection, the itemTypesError and the ItemQuantitiesError were found to be very strongly related with r_s=.991 (p<.001) leading to the conclusion that people who made mistakes in the kinds of the products, they made also mistakes in the quantity of products purchased. On the other hand, not so strong correlations were found between the moneyError and the itemTypesError with r_s=.498 (p<.001).

According to a Multinomial Logistic Regression (MLR) analysis performed taking as inputs the two performance metrics (itemQuantitiesScore and itemTypesScore), it can be seen that the model fit is significant with $X^2(4) = 15.662$, $p = .004$. A similar model using the error metrics (itemTypeError and itemTypesError) were not found to be better. Overall, it is more likely that an elderly person has some frailty symptoms if he/she has made some mistakes (errors) while playing the Virtual Super Market game, than if he/she had made no mistakes. Moreover, for no frailty symptoms presence (frailtyStatus=0), the 87.2% was predicted correctly based on the quantities and types of products purchased.

4.3.3 Towards prediction of frailty

Extraction of features within data clusters (activities)

Using the Activity Classification algorithm which we developed, we could analyze the stored data and annotate them accordingly. Then using this annotation, we are able to provide summaries of physiological parameters of participants towards the clinicians. Currently we are providing daily summaries of average/min/max values for Heart Rate and Respiration Rate for each participant during these activities: Sitting/standing, Lying, Walking, Walking Upstairs, Walking Downstairs. Results of the analysis using variables summarized within these activities will be presented in D4.15.

Deep learning for feature extraction and prediction

We started to investigate deep learning techniques for an in-depth analysis of the time series data and for the seamless extraction of a features' hierarchy that will be linked through a deep neural network to a frailty index. The prediction model will be used to provide a frailty indicator during the recordings without the need for a thorough clinical evaluation. An intrinsic challenge lies in the non-uniformity of the data in respect to duration, activities performed during the recordings, as well as the low quality of some signals due to their acquisition in a real-life home environment and not in a controlled experimental setting.

Since this method focuses on signal processing and analysis the method will be described in deliverable D4.16, while results will be reported in D4.17 where the largest amount of data will be available, absolutely necessary for deep learning techniques.

Machine learning for temporal analysis

Also, we will examine the change of this frailty indicator over the evaluation period and assess its temporal consistency. The change of frailty indicators calculated from clinical data has shown to be inconsistent over time manifested as unexpected frailty recession (e.g. transition from frail to non-frail) in a large percentage of subjects (Xue2011). Our hypothesis is that non-subjective reproducible measurements from sensor data are unbiased and could lead to more stable and reliable markers. Results of this analysis will be first presented in D4.17.

Finally, we will correlate our proposed frailty index with measurements from upcoming clinical evaluations and see whether a current index has any predictive ability for near-coming events and changes in the participants' health status, which are not observed by the clinical scores.

References

(Anguita2013) Anguita, D., Ghio, A., Oneto, L., Parra, X., Reyes-Ortiz, J.L., 2013. Human Activity Recognition on Smartphones using a Multiclass Hardware-Friendly Support Vector Machine. IWAAL 2012 International Workshop of Ambient Assisted Living.

(Dietterich1997) Dietterich, T. G., 1997. Machine learning research: Four current directions. Artificial Intelligence Magazines, vol. 18, pp. 97–136.

(Gaines2016) Gaines, Brian R., and Hua Zhou. "Algorithms for Fitting the Constrained Lasso." arXiv preprint arXiv:1611.01511 (2016).

(Hall2009) Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann P., Witten, I. H., 2009. The WEKA Data Mining Software: An Update. SIGKDD Explorations, vol. 11.

(Keerthi2001) Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K., 2001. Improvements to Platt's SMO algorithm for SVM classifier design. Neural Computation, vol. 13, pp. 637-649

(Kira1992) Kira, K., Rendell, L. A., 1992. A practical approach to feature selection. Proc. 9th Int. Conf. Mach. Learn., pp. 249 – 256.

(Kononeko1994) Kononenko, I., 1994. Estimating attributes: Analysis and extension of RELIEF. Proc. Euro. Conf. Mach. Learn., vol. 784, pp. 171– 182.

(Kononeko1997) Kononenko, I., Simec, E., Robnik-Sikonja, M., 1997. Overcoming the Myopic of Inductive Learning Algorithms with RELIEF-F. Applied Intelligence

(Platt1998) Platt, J., 1998. Fast Training of Support Vector Machines using Sequential Minimal Optimization. Advances in Kernel Methods - Support Vector Learning.

(Reyes-Ortiz2013) Reyes-Ortiz, J.L., Ghio, A., Parra, X., Anguita, D., Cabestany, J., Catala, A., 2013. Human Activity and Motion Disorder Recognition: Towards Smarter Interactive Cognitive Environments. ESANN 2013 21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning.

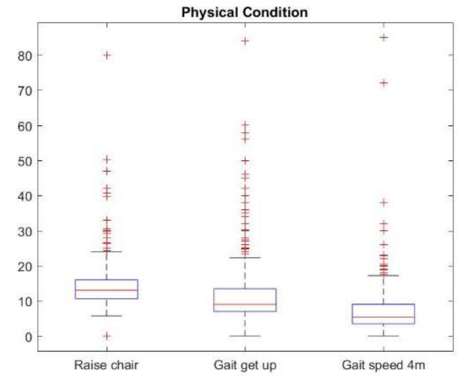
(Sun2006) Sun, Y., Li, J. 2006. Iterative RELIEF for feature weighting. Proc. 21st Int. Conf. Mach. Learn., pp. 913–920.

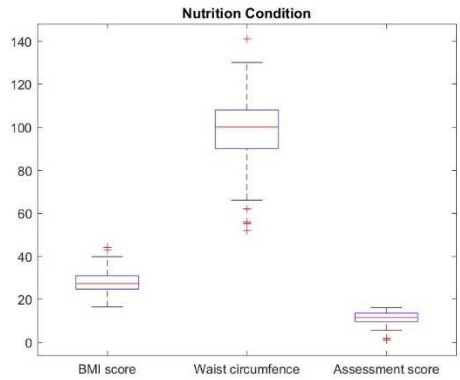
(Sun2008) Sun, Y., Wu, D., 2008. A RELIEF based feature extraction algorithm. In Proceedings of SIAM International Conference on Data Mining.

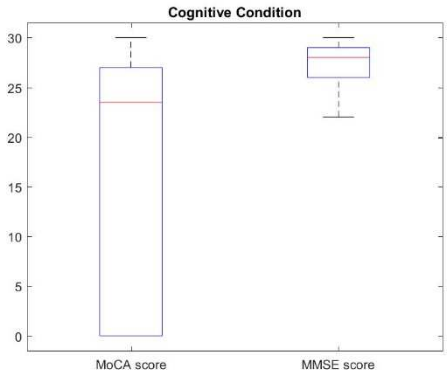
(Xue2011) Xue, Qian-Li. "The frailty syndrome: definition and natural history." Clinics in geriatric medicine 27.1 (2011): 1-15.

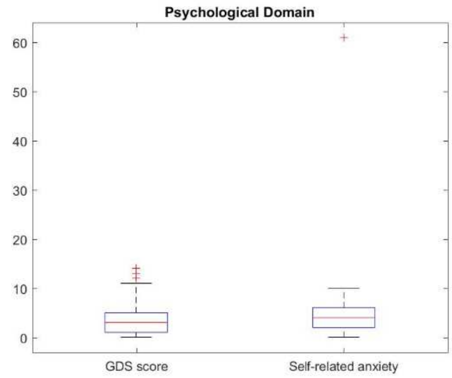
Appendix

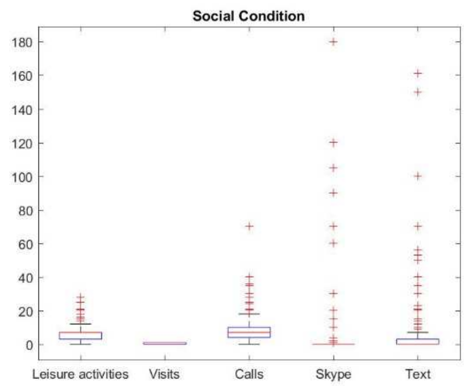
1. Box plots from univariate analysis

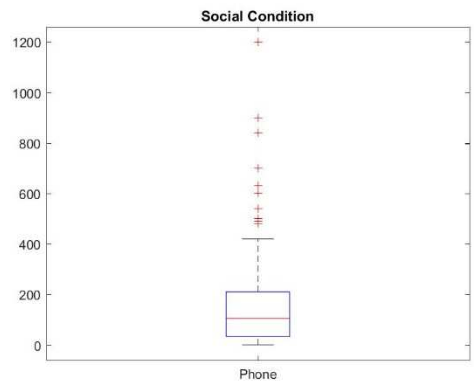


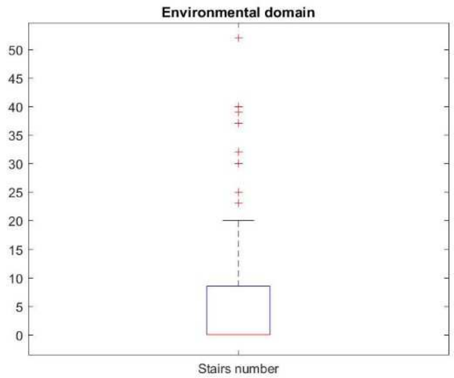


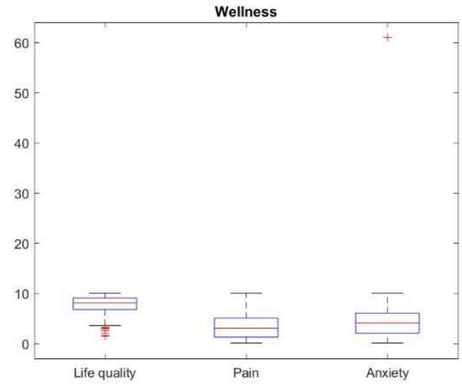


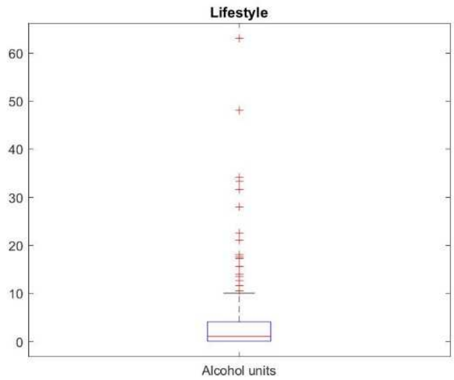












2. Tables with values of participants' answers from eCRF

Cognitive Domain

	Range	NonFrail	PreFrail	Frail
MoCA	<26	22	42	47
	26--30	89	80	27
Subjective Memory Complaint	NO	80	92	48
	YES	28	26	22
MMSE	24--26	14	42	47
	27--30	106	116	53

Environmental Domain

	Range	NonFrail	PreFrail	Frail
Suitable housing environment (participant's evaluation)	NO	6	4	4
	YES	92	123	90
Suitable housing environment (professional's evaluation)	NO	4	6	7
	YES	94	122	85
Stairs Number	0--9	65	105	84
	10--19	24	16	6

	20--30	5	5	5
	31--40	3	3	1
	41--52	1	0	0

Functional Domain

Katz Index	Range	Non Frail	PreFrail	Frail
	0 -- 4.5	0	2	12
	5 -- 5.5	27	27	13
	6	93	129	75
Telephone	Answers telephone; but does not dial	1	1	4
	Dials a few well-known numbers	3	5	8
	Operates telephone on own initiative; looks up and dials numbers	116	150	83
	I dont know	0	1	0
	Does not use telephone at all	0	0	5
Shopping	I dont know	1	0	0
	Completely unable to shop	0	0	16
	Needs to be accompanied on any shopping trip	0	6	16
	Shops independently for small purchases	5	15	9
	Takes care of all shopping needs independently	114	136	59
Food	Heats and serves prepared meals or prepares meals but does not maintain adequate diet	2	12	6
	Needs to have meals prepared and served	1	6	19
	Non applicable-never used to do this	8	13	3
	Plans; prepares; and serves adequate meals independently	102	124	64
	Prepares adequate meals if supplied with ingredients	7	2	8
Housekeeping	Does not participate in any housekeeping tasks	1	7	12
	Maintains house alone with occasion assistance (heavy work)	96	110	46
	Needs help with all home maintenance tasks	0	2	10
	Non applicable-never used to do this	4	11	7
	Performs light daily tasks such as dishwashing; bed making	14	22	17
	Performs light daily tasks; but cannot maintain acceptable level of cleanliness	5	5	8
Laundry	All laundry must be done by others	15	16	27
	Does personal laundry completely	94	120	59
	I dont know	0	1	0
	Launders small items; rinses socks; stockings; etc	3	6	5
	Non applicable-never used to do this	8	14	9
Transportation	Arranges own travel via taxi; but does not otherwise use public transportation	4	12	12
	Does not travel at all	2	1	8

	I dont know	0	1	0
	Travel limited to taxi or automobile with assistance of another	1	4	26
	Travels independently on public transportation or drives own car	111	131	51
	Travels on public transportation when assisted or accompanied by another	2	8	3
Medications	Is not capable of dispensing own medication	1	3	16
	Is responsible for taking medication in correct dosages at correct time	113	144	72
	Not applicable; does not take any medication	3	6	0
	Takes responsibility if medication is prepared in advance in separate dosages	3	4	12
Finances	I don't know	1	0	0
	Incapable of handling money	1	1	16
	Manages day-to-day purchases; but needs help with banking; major purchases; etc	2	7	17
	Manages financial matters independently (budgets; writes checks; pays rent and bills; goes to bank); collects and keeps track of income	109	138	60
	Non applicable-never used to do this	7	11	7
IADL	0 --5	0	1	0
	6 --11	0	1	7
	12 --17	1	9	19
	18 --23	8	16	15
	24 --31	111	131	59

General Domain

Unintentional weight loss	**Range**	**NonFrail**	**PreFrail**	**Frail**
	NO	118	144	67
	YES	0	12	31
Self-reported exhaustion	No	119	135	23
	Yes	0	23	76

Lifestyle domain

Physical activity	**Range**	**NonFrail**	**PreFrail**	**Frail**
	< 2 h per week	13	51	51
	> 2 h and < 5 h per week	34	57	17
	> 5 h per week	67	39	7
	No	4	11	24
Smoking	Current smoker	7	11	8
	Never smoked	71	98	59
	Past smoker (stopped at least 6 months)	40	49	33
Alcohol units (females)	<=14	74	93	60

	14.1 -- 21	1	1	2
	>21	3	1	0
Alcohol units (males)	<=21	40	60	37
	21.1 -- 28	0	2	0
	>28	1	0	0

Medical domain

Orthostatic hypotension	**Range**	**NonFrail**	**PreFrail**	**Frail**
	NO	104	134	87
	YES	15	20	10
Visual impairment	Sees moderately	23	46	25
	Sees poorly	1	9	12
	Sees well	96	103	63
Hearing impairment	Hears moderately	23	40	24
	Hears poorly	1	6	12
	Hears well	96	112	64
Hospitalizations (one year)	0	105	121	76
	1	13	27	13
	>=2	1	10	8
Hospitalizations (three years)	0	90	95	57
	1	26	44	21
	>=2	4	19	20
Comorbidities’ number	0	5	14	9
	1 -- 2	35	44	33
	3 -- 4	32	37	9
	5 -- 9	45	51	29
	>=10	3	12	20
Significant comorbidities’ number	0	97	119	69
	1	20	28	21
	>=2	3	11	10
Medication number	0--3	71	79	55
	4--7	37	54	28
	8--10	8	19	11
	>10	4	6	6

Nutritional domain

Body Mass Index	Range	NonFrail	PreFrail	Frail
	<=18	0	2	2
	18.2--21	6	6	8
	21.1--25	39	25	17
	25.1--29.9	55	69	31
	>=30	18	53	39
Waist (females)	<88	25	16	11
	>=88	52	77	51
Waist (males)	<102	25	24	14
	>=102	15	39	23
Mini Nutritional Assessment total score	0--16.5	0	0	3
	17 - 23.5	4	8	24
	24--30	7	5	2
Mini Nutritional Assessment screening score	0--7	0	2	6
	8--11	11	11	23
	12--14	109	145	71
Lean body mass (females)	19.5 --32.01	20	19	6
	32.02 --41.06	28	18	6
	41.07 --76.2	20	19	13
Lean body mass (males)	27.2 --44.5	11	8	6
	44.51 --56.27	13	12	3
	56.28 --74.1	11	13	5

Physical domain

Lower limb strength	Range	NonFrail	PreFrail	Frail
	0--10	35	27	14
	10.1--15	65	77	27
	>15	20	48	28
Balance	<5 sec	17	51	37
	>5 sec	102	98	27
	test non realizable	1	9	36
Gait related task speed	0--10	94	105	39
	10.1--12	12	12	5
	12.1--20	14	32	27
	>20	0	8	23
Gait speed 4 meters	0--0.8	44	103	80
	0.81--1	17	19	10

	1.1--1.2	8	4	0
	>1.2	51	32	5
Qualitative evaluation of mobility	NO	113	140	80
	YES	7	18	20
Grip strength	NO	119	45	5
	YES	0	113	94
Low physical activity	NO	119	143	43
	YES	0	15	56
Falls	0	96	114	60
	1	15	30	16
	>=2	9	13	21
Fractures	0	100	134	77
	1	19	16	18
	>=2	0	8	5

Psychological domain

Geriatric Depression Scale	Range	NonFrail	PreFrail	Frail
	0 -- 4	107	118	53
	5 -- 6	5	18	18
	7 -- 10	6	19	17
	10 -- 15	0	1	10
Anxiety	0--2.4	51	46	24
	2.5--4.9	30	48	25
	5--7.4	29	39	23
	7.5--10	10	25	28

Social domain

Living conditions	Range	NonFrail	PreFrail	Frail
	NO	79	85	71
	YES	41	72	29
Leisure activities	0--3	15	36	49
	4--6	31	34	14
	7--28	74	87	37
Membership of a club	NO	28	53	64
	YES	92	104	36
Number of visits	0--1	22	27	24
	2	30	20	12

	3--6	47	57	30
	7--20	20	51	32
Number of calls	0--3	16	36	33
	4--6	20	17	9
	7--9	42	57	36
	10--70	42	45	19
Time spent on phone	0--30	19	29	27
	31--104	22	35	16
	105--209	31	25	15
	210--1200	39	28	13
Time spent on skype	0	101	134	88
	2--17	6	2	1
	18--59	3	2	1
	60--180	5	7	1
Number of written messages	0	49	95	73
	1--2	10	14	4
	3--9	16	13	9
	10--161	40	19	2

Wellness domain

Quality of life self-rating	**Range**	**NonFrail**	**PreFrail**	**Frail**
	0--2.4	0	0	3
	2.5--4.9	3	9	8
	5--7.4	37	56	40
	7.5--10	80	93	49
Self-rated health status	1 - Very bad	0	1	2
	2 - Bad	2	6	10
	3 - Medium	26	47	53
	4 - Good	77	84	31
	5 - Excellent	15	20	4
Self-assessed change since last year	1 - A lot worse	0	2	8
	2 - A little worse	26	38	31
	3 - About the same	77	97	45
	4 - A little better	14	14	15
	5 - A lot better	3	7	1
Self-rated pain	0--2.4	72	70	27
	2.5--4.9	34	45	28
	5--7.4	12	29	32

	7.5--10	2	14	13
Self-rated anxiety	0--2.4	51	46	24
	2.5--4.9	30	48	25
	5--7.4	29	39	23
	7.5--10	10	25	28